SecureTransport

Version 5.5

29 August 2024

Administrator's Guide

This documentation describes the following Axway software:

Axway SecureTransport 5.5 Modernized Standard Cluster (Beta)

Revision history

The following changes are added to the SecureTransport 5.5 Administrator Guide.

SecureTransport 5.5 Updates	Topics changed
August 2024	• New topic added: *Secret vault integration* on page 66 • *ZDU: Detailed process description* on page 396updated for clarity • *Configure the database properties file* on page 392 updated with Microsoft SQL Server configuration properties • *Set up usage reporting in SecureTransport* on page 180 updated with a new metric: • *Troubleshooting I/O problems* on page 1062 updated with solution for slow file and folder listings • Added information about bulk resubmission of failed transfers: *Manage file transfers from the File Tracking page* on page 314 and *Resubmitted and retried transfers* on page 306 • *Duplicate an account* on page 679 updated with details on certificate uniqueness • *Manage subscriptions* on page 664 updated
July 2024	• *Transfer profile: Example configurations* on page 651 updated • *Publish To Account* on page 943 updated with a new option to control the visibility of files in the process of being published • *Axway and third-party software support* on page 35 updated • *Beta features* on page 33 updated • *View file transfer information* on page 309 updated • *Set up a cluster* on page 375 updated

SecureTransport 5.5 Updates	Topics changed
June 2024	• Updated, reorganized, and improved documentation on *Zero Downtime Update* on page 389 • *Transfer profile: Example configurations* on page 651 updated • *Beta features* on page 33 updated • *Axway and third-party software support* on page 35 updated • *Change external PostgreSQL configuration and manage partitioning* on page 104 updated • *Send To Partner* on page 948 updated • *Transfer Log Maintenance application* on page 856 updated • *Log Entry Maintenance application* on page 839 updated • *Connection pools for SecureTransport components* on page 110 updated • *Performance tuning for increased transfer load* on page 377 updated • *Advanced service configuration and memory allocation* on page 287 updated
May 2024	• *Utility files* on page 1066 updated • *Axway and third-party software support* on page 35 updated • Updates across the whole guide to reflect the new *PeSIT message transfers* on page 321 • *Graceful shutdown* on page 290 updated • *Transfer Log Maintenance application* on page 856 updated • *Log Entry Maintenance application* on page 839 updated • *Send To Partner* on page 948 updated for clarity • Added more details about *CFT PeSIT extensions* on page 318
April 2024	• New topic added: *Health checks for services and cluster nodes* on page 295 • New topic added: *Troubleshoot "CMS parsing has failed"* on page 1064 • *Zero Downtime Update* on page 389 process updated with a readiness check • *Set up usage reporting in SecureTransport* on page 180 updated for clarity • *Graceful shutdown* on page 290updated • *Manage filesystem restrictions* on page 786 updated • *Upload restrictions* on page 789 updated • *Transfer Log Maintenance application* on page 856 updated • *Log Entry Maintenance application* on page 839 updated • *Axway and third-party software support* on page 35 updated

SecureTransport 5.5 Updates	Topics changed
March 2024	• New topic added: *Zero Downtime Update* on page 389 • *Axway and third-party software support* on page 35 updated • *User certificates* on page 526 updated • *Duplicate an account* on page 679 updated • *Beta features* on page 33 updated • *Set up usage reporting in SecureTransport* on page 180 updated • *Configure SecureTransport Server to Edge streaming communication* on page 239 updated • *Sentinel Graphs explained with examples* on page 166 updated • *Server log* on page 322 updated • *Repository encryption* on page 46 updated • *Miscellaneous options* on page 199 updated • *Create a user account* on page 503 updated
February 2024	• *Create a user account* on page 503 updated • *Server log* on page 322 updated • *Axway and third-party software support* on page 35 updated • *Applications* on page 817 updated • *Set up usage reporting in SecureTransport* on page 180 updated • *Event Queue* on page 344 updated • *Define LDAP search criteria for a domain* on page 482 updated • *SSO configuration file for administrators* on page 435 updated • *SSO configuration file for end-users* on page 445 updated

SecureTransport 5.5 Updates	Topics changed
January 2024	• *Add a PeSIT server* on page 280 updated with new options for configuring retry parameters for the check of open pTCP connections • *Routing steps* on page 943 updated for clarity • *Log rotation and filtering* on page 1083updated • *Graceful shutdown* on page 290 updated • *Log Entry Maintenance application* on page 839 updated • *Transfer Log Maintenance application* on page 856 • *Server-initiated transfer authentication* on page 763 updated • *Client-initiated transfer authentication* on page 762 updated • *Add an SSH server* on page 277 updated • *Axway and third-party software support* on page 35 updated
December 2023	• *Axway and third-party software support* on page 35 updated • New topic added: *Manage the Dashboard page* on page 83 • New advanced use case example added: *Transmitting specific files to designated destinations as specified in a PeSIT acknowledgment* on page 976 • *Graceful shutdown* on page 290 updated
November 2023	• *Audit Log Maintenance application* on page 828 updated with instructions for configuring the chunk size • Added information on how to configure logging for *Unlicensed Accounts Maintenance application* on page 861 • *Set up usage reporting in SecureTransport* on page 180 updated with an option to configure reporting period • Updated documentation on how to configure log rotation • New topic added: *Event Queue* on page 344 • *General log files* on page 1077 updated • *Repository encryption* on page 46 updated

SecureTransport 5.5 Updates	Topics changed
October 2023	• *Pull From Partner* on page 956 updated • *Manage Routes* on page 881 updated • *Event states* on page 157 updated • *XFBTransfer Tracked Object* on page 128 updated • *PeSIT states and roles explained with examples* on page 171 updated • *General log files* on page 1077 updated • *File Tracking* on page 302 updated • *SSH transfer sites* on page 613 updated • *Axway and third-party software support* on page 35 updated
September 2023	• *Routing steps* on page 943 updated for clarity • *Trigger Route Execution on PeSIT Acknowledgment* on page 893 updated • New advanced use case examples added: ◦ *Configure AR flow with payload based on received acknowledgment* on page 980 ◦ *Send a file specified via PeSIT acknowledgment to a partner* on page 978 • *Additional attributes* on page 759 updated • *Configure Sentinel reporting* on page 123 updated • *General log files* on page 1077 updated • *Manage Route Package Templates* on page 875 updated
August 2023	• *Subscribe to Advanced Routing application* on page 888 updated with a new option to trigger routing upon receiving PeSIT acknowledgment • *Manage Routes* on page 881 updated • *Transfer related EL for AR* on page 1022 updated • *PeSIT related EL for AR* on page 1010 updated • *File Maintenance application* on page 834 and *Business units* on page 746 updated with details on policy inheritance • *Set up email notifications via SMTP* on page 202 updated • *Customize the email notification templates* on page 1125 updated • *Connection pools for SecureTransport components* on page 110 updated

Contents

Preface

This guide provides information about common administration and management tasks performed by Axway SecureTransport administrators.

Who should read this document

This guide is intended for system administrators who configure and maintain SecureTransport. As SecureTransport system administrator, you must be able to work effectively with the operating system platform and network used by SecureTransport. As a system administrator who deploys and configures SecureTransport and maintains its configuration, you must know or learn the file transfer requirements of your organization including the systems, protocols, and other standards used within your corporate network and by your partners. You must also be familiar with the configuration of systems that you integrate with SecureTransport, including LDAP, CA SiteMinder, Axway Sentinel, Axway Transfer CFT, and other file transfer clients and servers. As a system administrator who manages user and service accounts and maintains day-to-day operation of SecureTransport, you must understand the roles of your users and partners so you can assign them correct account properties and access permissions and restrictions. You must also be familiar with application logs and reports so you can interpret those provided by SecureTransport.

Available documentation

The following documentation is available for SecureTransport 5.5:

- *SecureTransport Administrator's Guide* – Describes how to use the SecureTransport Administration Tool to configure and administer your SecureTransport Server. The content of this guide is also available in the Administration Tool online help.
- *SecureTransport Appliance Guide* - provides the SecureTransport Appliance installation, configuration, and operation instructions. It also provides SecureTransport installation and upgrade instructions on Axway Appliances.
- *SecureTransport Capacity Planning Guide* – provides useful information when planning your production environment for SecureTransport.
- *SecureTransport Developer's Guide* – provides descriptions and usage instructions for implementing custom pluggable components in SecureTransport.
- *SecureTransport Getting Started Guide* – explains the initial setup and configuration of SecureTransport using the SecureTransport Administrator setup interface.
- *SecureTransport Installation Guide* – provides instructions for installing and uninstalling SecureTransport on UNIX-based platforms and Microsoft Windows.

- *SecureTransport on AWS Setup Guide* – provides a detailed overview and detailed instructions for setting up SecureTransport in the Amazon Web Services (AWS) Virtual Private Cloud (VPC).
- *SecureTransport on Azure Setup Guide* – provides a detailed overview and detailed instructions for setting up SecureTransport in the Microsoft Azure portal.
- *SecureTransport Upgrade Guide* – provides instructions for upgrading SecureTransport on UNIX-based platforms and Microsoft Windows.
- *SecureTransport Security Guide* – provides security information necessary for the secure operation of the SecureTransport product.
- *ST Web Client Configuration Guide* - describes how to configure and customize the ST Web Client user interface.
- *ST Web Client User Guide* – describes how to use the ST Web Client for end users.
- *SecureTransport Release Notes* – contains information about new features and enhancements in the current version of SecureTransport, as well as a comprehensive list of fixes and known issues.
- *SecureTransport Software Development Kit (SDK)* – a set of software development tools and examples that allow extending SecureTransport by consuming and implementing available APIs.
- *SecureTransport REST API documentation* – the portal published API documentation derived from the API swagger documents. To access the administrator and the end-user API documentation, go to docs.axway.com/category/api.

Accessibility Conformance Reports and statement

- Accessibility Conformance Report for SecureTransport Administration Tool
- Accessibility statement for SecureTransport Administration Tool
- Accessibility Conformance Report for ST Web Client
- Accessibility statement for ST Web Client

Visit docs.axway.com to view or download documentation.

Get more help

Go to Axway Support at support.axway.com to get technical support, download software, documentation and knowledgbase articles. The website requires login credentials and is for customers with active support contracts.

The following support services are available:

- Official documentation
- Product downloads, service packs, and patches
- Information about supported platforms
- Knowledgebase articles
- Access to your cases

When you contact Axway Support with a problem, be prepared to provide the following information for more efficient service:

- Product version and build number
- Database type and version
- Operating system type and version
- Service packs and patches applied
- Description of the sequence of actions and events that led to the problem
- Symptoms of the problem
- Text of any error or warning messages
- Description of any attempts you have made to fix the problem and the results

Training

Axway offers training across the globe, including on-site instructor-led classes and self-paced online learning. For details, go to training.axway.com

SecureTransport overview 1

SecureTransport is part of the Axway family of managed file transfer (MFT) products. SecureTransport allows organizations to adeptly control and manage the transfer of files inside and outside of the corporate firewall in support of mission-critical business processes and ad hoc human transactions, while satisfying policy and regulatory compliance requirements. SecureTransport serves as a hub and router for moving files between humans, systems, and more. SecureTransport also completes tasks related to moving files (push or pull), hosting files in mailboxes or "FTP-like" folders, and provides portal access with configurable workflow for file handling and routing. SecureTransport delivers user-friendly governance and configuration capabilities, including delegated administration and pre-defined and configurable workflows, while providing the highest possible level of security.

Designed to handle everything from high-volume automated file transfers between systems, sites, lines of business, and external partners, to user-driven ad hoc communications, to portal-based file exchange, SecureTransport supports the full range of file transfer scenarios while satisfying stringent security, policy, and regulatory compliance requirements. Serving as an MFT gateway, SecureTransport can perform the following key MFT functions:

- Accelerate and manage movement of files (push or pull) and host files in secure mailboxes or folders
- Push data securely to trading partners in real time
- Support ultra-high-end shared service bureaus to meet the demands of multiple business units and organizations in one scalable infrastructure
- Provide a configurable workflow for flexible and dynamic file handling and routing

Also, user-friendly governance and configuration capabilities, including delegated administration and pre-defined and configurable workflows, make SecureTransport a secure, easy-to-implement, and easy-to-use alternative to high-maintenance proprietary file transfer software, simple MFT gateways, and costly VANs and VPNs.

SecureTransport is compatible with FTP, FTPS, HTTP, HTTPS, SSH, FIPS 140-2 Level 1, AS2, and PeSIT standards. SecureTransport includes features that support business processes that are mission-critical to the enterprise and the documentation, auditing, and accountability required by government regulations such as HIPAA, GLBA, and Sarbanes-Oxley.

SecureTransport has many enterprise-class features, including the following:

- Standard Clustering and Enterprise Clustering
- Comprehensive authentication and access control
- A user-friendly HTML5-based end user client for file transfers, transfer status, and full email-style message compose, inbox, and outbox views for ad hoc transactions
- Comprehensive multiple LDAP system integration and mapping
- Complete representational state transfer (REST) web services APIs for administration, for file transfers management, and for custom end user interactions

- Interactive and automated transfers
- Guaranteed delivery
- Flexible support for deployments using one or more peripheral networks (DMZs) that host SecureTransport Edge servers configured to provide specific protocol and proxy services
- Data integrity
- Comprehensive logging and auditing
- Event-driven agents
- Java APIs and protocol support for application integration
- Scheduled transfers
- Advanced Routing
- PGP encryption and decryption
- Compression and decompression
- Line ending and transcoding
- Embedded Axway File Bus support

The cryptographic libraries used by SecureTransport for the AS2 (SSL), FTPS, HTTPS, PeSIT (SSL and legacy SSL), and SSH (SFTP and SCP) protocols have been certified Federal Information Protection Standard (FIPS) 140-2 Level 1 compliant by the US National Institute of Standards and Technology (NIST), Computer Security Division, and the Communications Security Establishment of the Government of Canada Information Protection Group.

SecureTransport offers two REST APIs: the SecureTransport Administrator API and the SecureTransport End-User API. The Administrator API provides configuration services for business units, accounts, routes and associated entities that are part of managed file transfer data flows. With the End-user API, you can monitor file transfers and manage user mailboxes and shared folders. To access the API documentation, go to docs.axway.com/category/api.

SecureTransport Server

SecureTransport Server provides a centrally managed system for monitoring and managing secure file transfer activity across multiple file transfer sites or applications. Key capabilities of the SecureTransport Server include:

- **Guaranteed delivery** – Guarantees secure, reliable, and scalable file transfer service even over unstable network connections or dial-up lines using integrity checking implemented with a cryptographic hash algorithm and provides powerful automation to integrate with back-end systems
- **Checkpoint restart** – With Axway clients, allows restarting stopped or failed HTTP, PeSIT, and some FTP file transfers from the point of failure
- **Secure connectivity** – Accepts, validates, and secures incoming connections and file transfers
- **Multi-protocol support** – Executes file transfers using widely adopted open standard FTP, secure FTP, HTTP, HTTP(S), AS2, PeSIT, SSH-based (SFTP and SCP), and Folder Monitor protocols

- **Proxy support** – Can use the SOCKS5 proxy provided by SecureTransport Edge or an HTTP proxy
- **Native Axway File Bus support** – Supports PeSIT for connectivity with the Axway File Bus and implements the Axway File Bus using metadata, routing, and automation capabilities aligned with Axway Transfer CFT
- **Ad hoc file transfer support** – Manages human-to-human (H2H) and human-to-system (H2S) file transfers sent using ST Web Client and system-to-human (S2H) file transfers delivered through email notifications
- **Repository encryption** – Encrypts all files on disk at the server transparently
- **PGP encryption** – Handles PGP encryption and decryption, the generation and storing of PGP keys, and the management of the stored keys
- **Application integration** – Includes REST and Java APIs, protocols, a file services interface, and Axway File Bus support
- **Transfer scheduling** – Allows administrators to plan and configure scheduled file transfers and AdHoc tasks
- **Monitoring and reporting** – Monitors and analyzes file transfer activity, providing real-time reports and alerts
- **Signed messaging disposition notification (MDN) receipts** – Can generate receipts for all transfers, regardless of protocol
- **Flexible clustering models** - Includes Standard Clustering for simple deployment with no external dependencies and optional Enterprise Clustering to increase the capacity of a SecureTransport deployment to handle large workloads
- **Database support** – Uses an embedded database or, with optional Enterprise Clustering, an external database to store and retrieve configuration parameters and data pertaining to objects and events
- **Web administration and configuration** – Provides a web-based user interface (the Administration Tool) for centralized administration, configuration, and monitoring of file transfer activity and applications
- **Fully Embeddable** – Includes a REST API with resources for administration, configuration, and file transfer request management and for creating custom end user access

SecureTransport Server is available as software on Windows and UNIX platforms and as a Cloud service. You can deploy it as part of an Enterprise Cluster or as a stand-alone server.

SecureTransport Edge

SecureTransport Edge is the gateway required in the perimeter network (also called demilitarized zone or DMZ) in a typical multilayer security architecture deployment. You can use SecureTransport Edge to implement secure interactions between client systems in a public or other external network and SecureTransport Servers in your internal secure network.

SecureTransport Edge serves as a protocol converter in such a deployment. It treats a wide range of file transfer protocols as presentation layer services and each protocol server translates its protocol onto the streaming protocol used to communicate with the Transaction Manager (TM) server on the SecureTransport Server. The TM Server connects to the protocol servers on the configured SecureTransport Edge servers to establish the connections for the streaming protocol, so no process on a SecureTransport Edge ever makes a connection from the DMZ into the internal secure network. A flexible network zone configuration supports connection to the protocol servers on specific SecureTransport Edge servers for different protocols and file transfers. For more information see *Communication across Transaction Manager, protocol, and proxy servers* on page 227.

SecureTransport Edge serves all the protocols supported by SecureTransport. When an external partner client program or file transfer server initiates a connection to one of the protocol servers hosted on SecureTransport Edge, it terminates the inbound connection from the client, collects the client's credentials, and establishes an authenticated encrypted connection to the TM. SecureTransport Edge sends the credentials to the TM as a service request. The TM attempts to authenticate the account using the configured method and returns the result to SecureTransport Edge. If the account is authenticated, SecureTransport Edge establishes the connection.

For a file transfer, SecureTransport Edge uses the streaming protocol to check the access control rules on SecureTransport Server to authorize the transfer. SecureTransport Edge converts the network messages between the client protocol and the SecureTransport streaming protocol, decrypting and encrypting the data as needed. The data is *streamed* between the external-facing protocol server and the Transaction Manager, the streaming protocol server, running on the SecureTransport Server. No transferred file data is stored in the SecureTransport Edge file system in the perimeter network.

When SecureTransport Server connects to a partner server in the external network to check for files or to transfer a file, it can use the SOCKS5 circuit-level proxy component of SecureTransport Edge to broker the connection through the perimeter network to the external network. Thus, the authentication credentials exist only in the internal secure network and are encrypted until they are presented to the external server. (SecureTransport Server can also use an HTTP proxy.)

SecureTransport Edge is available as software on Windows and UNIX platforms. You can deploy it with stand-alone or clustered SecureTransport Servers. You can deploy two or more SecureTransport Edge systems in support of a SecureTransport Server cluster and synchronize configuration changes dynamically. Each SecureTransport Edge stores its configuration in a local embedded database. For more information see *SecureTransport Edge synchronization* on page 407.

Deployment models

SecureTransport can be deployed as part of a Standard Cluster (SC), Enterprise Cluster (EC), or in standalone mode.

Cluster models

To provide flexibility for both ease in managing clustering and scale to meet the most demanding of loads, SecureTransport Server offers two cluster models. These are the Standard Cluster and Enterprise Cluster.

Standard Clustering uses an embedded database, which minimizes external dependencies and overhead and reduces the cost of clustering. A Standard Cluster can have from two to three nodes (servers). For more information, see *Standard Cluster* on page 351.

For a situation that exceeds the capacity of a Standard Cluster or requires a shared database, SecureTransport 5.5 offers an Enterprise Cluster option. An Enterprise Cluster using an external database and a high-performance cache-management layer significantly improves efficiency, provides near-linear scaling, and enables very large scale configurations. With the Enterprise Cluster option, an active/active cluster can have up to 20 nodes. The Enterprise Cluster option requires your organization to provide and maintain an external database. You must also provide a high-performance shared file system for the user files. For more information, see *Enterprise Cluster* on page 368.

Standalone deployment

When you do not need a cluster for additional capacity or improved availability, you can deploy SecureTransport Server as a single server. A stand-alone SecureTransport Server can use the embedded database or an external database. All stand-alone deployments can use a local file system for user files.

Note Multiple standalone SecureTransport instances sharing the file system where the user home directories are located is not a supported configuration.

Beta features

Following our "One Version" release model, some of the features introduced with SecureTransport 5.5 Updates are flagged as Beta ((previously called Technical Preview). Their purpose is to allow you to preview upcoming features and changes to SecureTransport before they are released as General Availability. By releasing Beta features, we aim to shorten the feedback cycle, ease the adoption, and lower the risk when releasing new functionality.

Beta features are still evolving and can be incomplete. During the preview period, we may change the behavior of a Beta feature based on received feedback. Those changes should not be considered regression defects.

Please note that the REST API endpoint definitions of a Beta feature may change, and these endpoints do not comply with the REST API versioning policy for breaking changes while in preview mode.

Beta features:

- Are labeled as 'Beta' in the documentation, UI and/or REST API. This label will be removed when the feature is released in GA.
- Are disabled by default and can be enabled via feature toggles or optional parameters.
- Have undergone extensive regression testing and do not break backward compatibility when turned off. Existing integration points such as xml import and REST API calls will continue working as expected.

- Have undergone the same security testing and adhere to the same security standards as the GA features in the release.
- May support only a subset of use cases, have limitations in the scope of the supported use cases, or may not behave as intended in all use cases. Information about the known limitations will be provided on a per-feature basis.
- Will likely lose their configuration when a new Update is installed.
- May cause performance degradation when enabled.
- Beta features must not be used in Production environments. Any use of a Beta feature is at the customer's own risk and will not be covered by the Axway Support policy.

Questions and feedback on Beta features should be sent to st-techpreview@axway.com.

Features currently in Beta

You can explore these Beta features before they're generally available.

Modernized Standard Cluster

The Modernized Standard Cluster is the successor of the Legacy Standard Cluster, with notable differences outlined below:

- The embedded database for SecureTransport Server and Edge is changed from MariaDB to PostgreSQL.
- In MSC, each SecureTransport node runs its own instance of the PostgreSQL server. Data replication among these instances is achieved through PostgreSQL-native logical replication. For more information, see PostegreSQL Documentation. The replication rules between the databases of the cluster nodes can be managed as needed.
- Automatic application and database failover capabilities
- MSC utilizes Coherence for cluster communication, replacing the previous reliance on JGroups.

For comprehensive information on Modernized Standard Cluster, please refer to the dedicated guide covering installation, setup, and administration details. The document is available on the Axway Documentation portal only after login.

Dynamic Node IP Address Discovery

This feature automates the process of adding and updating IP addresses for a Network Zone node. When enabled, SecureTransport performs regular DNS queries against a configured FQDN to find out and dynamically update the Network Zone node's IP addresses. *Learn more*

New interface of the Route Package Templates

A modernized version of the *Route Package Templates* page allows master administrators to create, edit, and delete route package templates and their respective routes. New features, like breadcrumb navigation and reording of routes and steps, are also part of the enhancement.
Currently only the following route steps can be added through the new interface: Publish To Account, Compress, Decompress, Encoding Conversion, External Script, Line Ending, Line Truncating, PGP Decryption, PGP Encryption, and Rename. The rest of the steps can be added via the old interface.

Axway and third-party software support

This section lists the Axway and third-party software supported for the various protocols and integrations.

Operating Systems

Minor releases of supported major versions are considered safe to upgrade and are supported by SecureTransport. If a release is deprecated by the vendor, it automatically gets not supported by SecureTransport. We consider a release deprecated when it no longer receives security updates by the vendor. Deprecation notice will not be issued.

Operating systems are supported both on hardware or Type-1 hypervisors. If Axway suspects that the virtualization layer is the root cause of an incident, the customer may be required to contact their virtualization support provider to resolve it.

While SecureTransport is expected to function properly in a virtual environment, there may be performance implications which can invalidate typical deployment recommendations.

Only 64-bit operating systems are supported.

Linux

- SUSE Linux Enterprise Server 12.x **(EOS January 2025)**
- SUSE Linux Enterprise Server 15.x
- Red Hat Enterprise Linux 8.x
- Red Hat Enterprise Linux 9.x
- Rocky Linux 9.x
- Oracle Linux 8.x
- Oracle Linux 7.x **(EOS December 2024)**

Windows

- Windows Server 2019 **(EOS March 2025)**
- Windows Server 2022

Cloud Environments

Any cloud environments are supported using a VM running a supported operating system.

Customers should consider vendor specific limitations.

For recommended deployment instructions, refer to the deployment guides for the following cloud environments:

- Amazon AWS
- Microsoft Azure

Databases for Enterprise Cluster

Minor releases of supported major releases are considered safe to upgrade and are supported by SecureTransport. It is the customer's responsibility to keep their external databases up to date with the latest updates.

Microsoft SQL Server

The following major releases of Microsoft SQL Server (Standard, Enterprise and Developer Editions) are supported:

- Microsoft SQL Server 2017
- Microsoft SQL Server 2019

The following features are supported for all supported releases:

- SQL Server Always On Failover Cluster Instances

Oracle Database

The following major releases of Oracle Database are supported:

- Oracle Database 12.2 Family (19c LTS) - Only Enterprise Edition with Partitioning License option is supported

The following features are supported for all supported releases:

- Oracle RAC (with SCAN)
- Kerberos authentication

- Kerberos authentication is currently not supported for containerized deployments and Amazon RDS.
- The Oracle database server may incorrectly interpret requests from SecureTransport as a replay attack ("Request is a replay" error message). This issue is resolved in Oracle 19.8 Patch 31716873. On previous Oracle database versions, the available workaround is to disable the Replay Cache mechanism.

PostgreSQL

The following major releases of PostgreSQL are supported:

- PostgreSQL 12.x **(EOS November 2024)**
- PostgreSQL 14.x
- PostgreSQL 15.x

Azure Database for PostgreSQL

The following major releases of Azure Database for PostgreSQL (Flexible Server) are supported:

- PostgreSQL 12.x **(EOS November 2024)**
- PostgreSQL 14.x

Amazon RDS

SecureTransport supports the Amazon RDS cloud service for the following database engines and their above-mentioned versions:

- Oracle (Kerberos authentication is not supported)
- PostgreSQL

Browsers

Both the Administration Tool and the ST Web Client are supported on the latest version of the following browsers:

- Apple Safari (not supported for Admin UI)
- Google Chrome
- Microsoft Edge
- Mozilla Firefox

Note Browser must have JavaScript enabled.

Software for File Exchange

SecureTransport is expected to work properly with any client or server software which complies with:

Protocol	CIT / SIT	Standard/Details
FTP(S)	CIT, SIT	RFC 959 RFC 2228 RFC 2389 RFC 2428 RFC 2640 RFC 4217
SFTP / SCP	CIT, SIT	RFC 4251 RFC 4252 RFC 4253 RFC 4254 Draft RFC - Secure Shell File Transfer Protocol Draft RFC - SSH File Transfer Protocol (draft-ietf-secsh-filexfer-04.txt)
AS2	CIT, SIT	RFC 4130 (AS2 1.0 and 1.1 protocol versions) List of certified products: https://www.drummondgroup.com/certified-products-2/applicability-standards/ S/MIME version v3.1 (RFC 3851) is supported.
PeSIT	CIT, SIT	PSIT_HS_E (PeSIT rev.E) pTCP protocol v2 as an extension to PeSIT rev.E
HTTP	CIT, SIT	
HTTP - JMS	SIT	Supported by JMS Connector
HTTP - Google Cloud Storage	SIT	Supported by Google Cloud Storage Connector
HTTP - Google Drive	SIT	Supported by Google Drive Connector
HTTP - OneDrive	SIT	Supported by OneDrive Connector
HTTP - Azure File Storage	SIT	Supported by Azure File Storage Connector

HTTP - Azure Blob Storage	SIT	Supported by Azure Blob Storage Connector
HTTP - Microsoft SharePoint	SIT	Supported by SharePoint Connector
SMB 2.x, SMB 3.x	SIT	Supported by SMB Connector
HTTP - Amazon S3	SIT	Supported by Amazon S3 Connector
HTTP - Axway Syncplicity	SIT	Supported by Syncplicity Connector
HTTP - Apache Hadoop	SIT	Supported by Hadoop Connector

Authentication Providers and Directory Services

SecureTransport is expected to work properly with any LDAP server implementations that are compliant with:

- RFC 4510 - LDAP v3

Single Sign-On (SSO)

SecureTransport is expected to work properly with any software which complies with:

- SAML 2.0 for administrators and end-users
- Kerberos 5 for end-users

SecureTransport only supports SAML-based Identity providers for SSO for administrators.

The client name has to be the same on all Identity Providers, SecureTransport only supports one service provider per component (Administrator and End-user).

ICAP antivirus and DLP

SecureTransport supports integration with ICAP servers as part of its process flows, allowing antivirus (AV) and Data Loss Prevention (DLP) scanning for the processed data. Although not officially certified, SecureTransport is known for its ability to operate alongside other antivirus, Intrusion Detection System (IDS), or Endpoint Detection and Response (EDR) agents installed on the same machine, provided they do not perform file locking on files during SecureTransport processes, including installation and updates. If during a troubleshooting session with a customer Axway Global Support suspects that the antivirus software installed on a machine where SecureTransport is running is the underlying cause of an incident, it may be necessary to disable this software and/or reach out to its vendor for further assistance.

SecureTransport implements the ICAP functionality adhering to the published ICAP standard, therefore it is expected that it will work with any server complying to the finalized ICAP standard (RFC 3507).

However, based on previous experience from validating ICAP servers, connecting each additional ICAP server to SecureTransport is associated often with particularities in requests/responses for ICAP scan, which could result in behavior change and/or failures in the processing. Axway will evaluate such incidents (if any) and, whenever possible, will address them in the SecureTransport ICAP implementation to ensure operation continuity.

Incidents requiring major changes will be evaluated individually and addressed as part of the SecureTransport roadmap for new development.

Software	Type	Version	Details
Symantec DLP	DLP	15.8	
Trellix DLP	DLP	11.5	
Symantec Protection Engine for NAS	AV	8.2.2	AVSCAN and AVSCANREQ are supported
Secure Web Gateway	AV	10.2	

Proxy Software

Supported Proxy Types:

- HTTP(S) - supported only with HTTP(S) and AS2 transfer sites
- HTTP(S) reverse proxy for HTTP transfers
- SOCKS5
- SecureTransport Edge

PGP

SecureTransport is expected to work properly with any PGP keys which have been generated with compliance with RFC 5581.

File Systems for User Files

- Amazon EFS
- Amazon FSx for OpenZFS over NFS v3.0 and v4.0
- Azure NetApp Files (ANF) over NFS v3.0
- CIFS

- GFS 2
- GlusterFS
- IBM Spectrum Scale (GPFS) 5.x
- NFS 3.0 - RFC 1813
- NFS 4.0 - RFC 7530
- NTFS
- OCFS 2

Hardware Security Modules (HSM)

SecureTransport supports storing local certificates in a Thales Luna 7 HSM via Client 10.x. For details, see *Store certificates in a Luna hardware security module* on page 63.

Compatibility with Axway Products

SecureTransport is compatible with the following Axway products for all of their supported releases.

Reporting

- Axway Sentinel (QLTv1)
- Decision Insight (QLTv1)
- Embedded Analytics (QLTv1)

Flow management

- Flow Manager (supported as of SecureTransport 5.5-20210729)

  We recommend that you update Flow Manager and your SecureTransport instances to most recent versions to benefit from the latest functionalities and corrections. Flow Manager version required for flow deployment: 2.0 BN20220505 or higher.

Disclaimer

SecureTransport supports only third-party software or OS releases that are supported by their vendor. End of support versions are automatically not supported by SecureTransport.

Deprecation of supported operating systems and databases will be announced in advance.

Setup 2

The following set of topics provides detailed SecureTransport configuration and setup information:

- *Certificates* on page 44: types, import and export, generating certificates and certificate signing requests.
- *PGP key encryption and signing* on page 68
- *Configure FTP server messages and modes* on page 72
- *Configure HTTP server messages* on page 79
- *Configure the AS2 server settings* on page 80
- *Configure Administration Tool server settings* on page 82
- *PeSIT server configuration settings* on page 84
- *Configure AdHoc file transfers* on page 87
- *Configure your database* on page 93
- *Sentinel* on page 121
- *Server licenses* on page 178
- *Configure FTP command log* on page 188
- *Configure FTP command log* on page 188
- *Configure transfer log* on page 192
- *Configure holiday schedule* on page 194
- *Mail templates* on page 195
- *Configure miscellaneous settings* on page 199
- *ICAP settings* on page 207
- *Transaction Manager Settings* on page 217: enable, disable, export rule packages
- *File archiving* on page 224
- *Communication across Transaction Manager, protocol, and proxy servers* on page 227
- *Configure SecureTransport Server to Edge streaming communication* on page 239
- *Address Book* on page 242

Log in to the SecureTransport Administration Tool

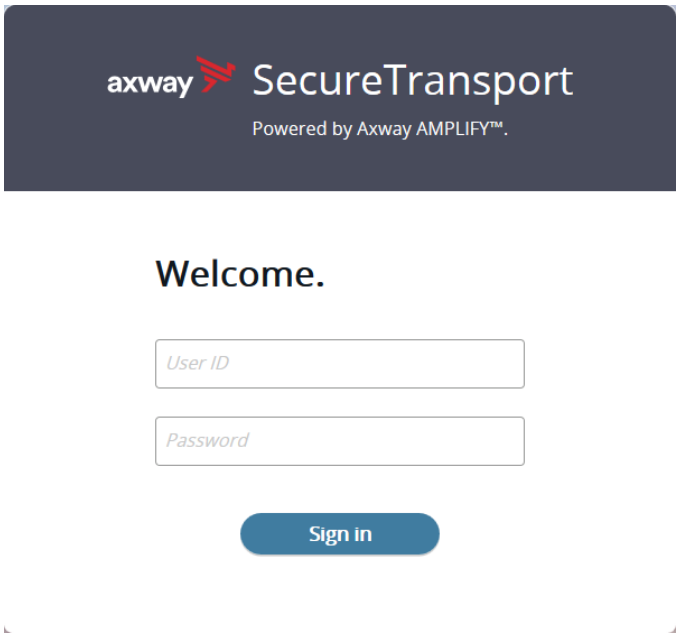
To log in to the SecureTransport Administration Tool through a web browser, take the following steps. For a list of supported web browsers, see *Axway and third-party software support* on page 35.

1. Open the web browser.
2. Type the URL for the Administration Tool as follows:

   `https://<host>:port`

   where `<host>` is the host name, FQDN or IP of the computer running SecureTransport and `port` is the administration port number entered during installation. The default port number is 444 for root installations. When SecureTransport is installed as non-root, the default port is 8444.
3. Following the instructions for your browser, add a certificate exception for the SecureTransport instance if needed.

   The login page is displayed.



4. Type the administrator name and password. The default user name is `admin` and the default password is `admin`.
5. Click **Log In**.

If an error occurs when logging into the Administration Tool, SecureTransport prompts you to type the name and password again.

Note When using the Administration Tool, make sure that your browser does not block pop-up windows for SecureTransport.

Note You cannot open Administration tool in multiple browser tabs/windows that share the same session.

For more information about administrator login, see *Administrator login options* on page 468 and *Manage administrator accounts* on page 700.

Certificates

This topic describes the different types of certificates used in SecureTransport and how to import, export, and generate certificates and certificate signing requests.

SecureTransport uses digital certificates for many security functions. These certificates can either be signed by a self-signed Internal Certificate Authority (CA), that is, issued by the SecureTransport Server; signed by an imported internal CA; or signed by a third party, such as an external company like Verisign or a corporate CA. During the installation process, SecureTransport installs a default self-signed CA (valid for one month) that you should replace during the initial setup procedures. For details about initial setup procedures for certificates, refer to the *SecureTransport Getting Started Guide*. You can also import an external CA to serve as the SecureTransport internal CA so that certificates signed by SecureTransport are trusted by clients that trust that CA.

Note As of Update 5.5-20201029, it is required that the certificates with the *keyUsage* extension contain the *digitalSignature* bit.

The following topics describe the certificate types and provide how-to instructions for managing certificates.

Certificate types

SecureTransport uses the following types of certificates overall:

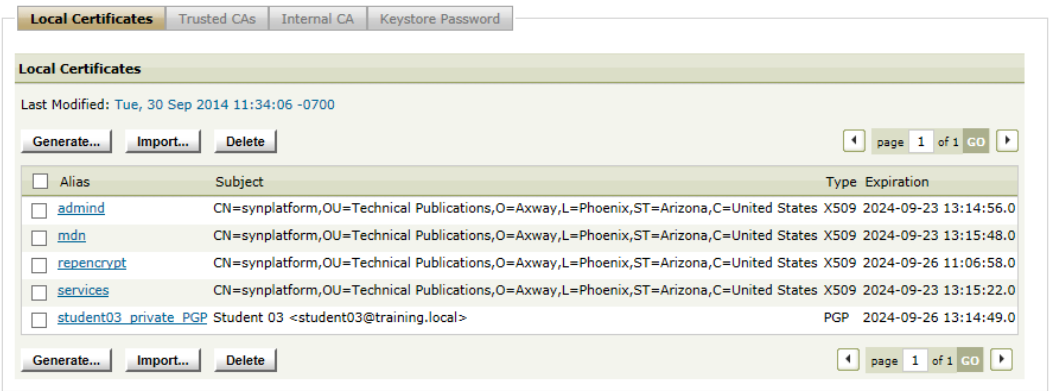
- **Local Certificates** – Local certificates are used for server authentication. They are also used to decrypt incoming documents and sign documents and receipts. The SecureTransport administrator can generate self-issued local certificates, import local certificates signed by a third-party certificate authority, and export certificate public keys and private keys to a file. For example, if you have a certificate signed by a widely-trusted CA such as Verisign, you can import it as a local certificate and use it for the HTTPS SSL key alias so browsers do not require the user to accept a certificate which they cannot validate. You manage server-scoped local certificates from the Setup menu. For more information, see *Manage local certificates and certificate signing requests* on page 48.

- **Trusted Certificate Authority (CA) Certificates** – CA certificates are used for indirect trust of SSL client or server certificates. SecureTransport includes a set of commonly used CA certificates. For SecureTransport to use imported local certificates, the trusted CA certificates must include certificates for all CAs in their certification paths (certificate chains). CA certificates are only server-scoped, and there are no account-specific CA certificates.
- **Internal CA** – A CA is the authority that issues, manages, revokes, and renews certificates, and assists people with performing certificate life cycle tasks such as retrieving, renewing, revoking, and so on. CAs are represented by a certificate that contains the name of the issuer and they are used to sign end-user certificates. An internal CA represents the company that you work for or a company that you pay to issue you a certificate.
- **Login Certificates** – Client authentication certificates are assigned to a specific user and are required for logging in to SecureTransport. You can use SecureTransport to generate X509 login certificates and to import X509 login certificates and SSH keys. You can manage user certificates from the **Accounts** menu. For more information, see *Manage login certificates* on page 527*Manage login certificates* on page 527.
- **Partner Certificates** – Account-specific public certificates are used for encrypting PGP and AS2 data before sending to the respective account and for verifying the signature of data from the account. Account-specific certificates can be managed using the **Accounts** menu. For details, see *Manage partner certificates* on page 530.
- **Private Certificates** – Account-specific certificates used to decrypt incoming documents and sign documents and receipts. The Certificate Manager can generate self-issued private certificates, import private certificates signed by a third-party certificate authority, and export certificate public keys and private keys to a file. You manage account-specific private certificates from the **Accounts** menu. For more information, see *Manage private certificates* on page 534.

Certificate Management page

To access the *Certificate Management* page, select **Setup > Certificates**.

Note In SecureTransport 5.5, regardless of installation type (either fresh install or upgrade from previous version), trusted certificates come with a `jdk` label in the certificate alias. This indicates that they are imported from JRE and does not affect the SecureTransport operations.



Use the *Certificate Management* page to perform the following certificate management functions:

- Import certificates
- Delete certificates
- Generate self-issued local certificates and certificate signing requests (CSRs)
- View certificates
- Export certificates to a file
- View, delete, or finish pending CSRs

Note Because SecureTransport stores the trusted and local certificates and the protocol server configuration in the database and recreates them in the file system when a server starts, you must import certificates into the database using the Administration Tool. You cannot install a certificate by copying it into a directory as you could in previous releases.

Repository encryption

Repository encryption increases SecureTransport security by avoiding storing unencrypted files. It can be enabled on different levels (for example, per account). When you enable repository encryption, SecureTransport encrypts (according to the activated repository encryption level) each file that it pulls from a partner site or that a client pushes to it. When SecureTransport pushes a file to a partner site or a client pulls a file from it, SecureTransport decrypts the file. SecureTransport encrypts and decrypts each file dynamically in memory as it receives and sends it, so the files never exist unencrypted in the storage of the host system.

By default, SecureTransport utilizes the Bouncy Castle provider for cryptographic operations, such as repository encryption, but it also supports SunJSSE. Bouncy Castle is FIPS-certified and offers more cipher suites, while SunJSSE makes use of AES-NI when available, which can greatly accelerate AES operations. The difference in performance is particularly noticeable in the SIT transfers and CIT uploads to shared storage locations. If FIPS compliance is not required, you can boost system performance by setting the server configuration option `TM.preferBouncyCastleProvider` to `false`. That way the cryptographic operations going through the Transaction Manager will be handled by SunJSSE.

Enable repository encryption

Follow the procedure:

1. Import an X.509 certificate in PKCS#12 format or generate a self-issued local certificate. See *Import a local certificate* on page 53 and *Generate a self-issued server certificate* on page 49.
2. Set the value of the `Stfs.Encryption.CertAlias` server configuration parameter to the alias of the certificate. SecureTransport uses this certificate to encrypt and decrypt files. See *View and change server configuration parameters* on page 334.

   SecureTransport prevents you from deleting the certificate referenced by `Stfs.Encryption.CertAlias`.

Note If `Stfs.Encryption.CertAlias` is not set, Repository Encryption will not be enabled.

3. To choose the encryption algorithm, set the value of the `Stfs.Encryption.Algorithm` server configuration option to one of the following values:
   - `AES128` (default)
   - `AES256`
   - `3DES`

   See *View and change server configuration parameters* on page 334.
4. To configure SecureTransport to compute the MD5 checksum for an uploaded file dynamically as the file is uploaded, set the value of the `Stfs.Hash.HashOnUpload` server configuration parameter to `true`. When the value is `false`, the default value, SecureTransport computes the MD5 checksum after the file transfer is complete.
5. Create a user class named `EncryptClass`. Files transferred by users in this class are encrypted. See *Add a user class* on page 776.

   Note For server-initiated transfers, the user class is defined by the UID and GID only. If you define the `EncryptClass` using user name or other attributes, there are limitation on server-initiated transfers. See *Repository encryption and server-initiated transfers* on page 765.

   Note Repository encryption can also be enabled on an individual account basis. When the setting is specified in the account, the user class is ignored.
6. Restart the TM Server. See Start and stop servers.

Note If you enable repository encryption, the following SecureTransport functions are not supported: resume PeSIT transfers and pause and resume transfers when SecureTransport is the server.

Change the repository encryption certificate

Follow the procedure:

1. Import a new X.509 certificate in PKCS#12 format or generate a new self-issued local certificate.

   Do not delete the old certificate as it will still be used to decrypt files that were encrypted with it.
2. Change the value of the `Stfs.Encryption.CertAlias` server configuration option to the alias of the new certificate.
3. Restart the Transaction Manager.

Note The `repconv` tool located in `<FILEDRIVEHOME>/bin` can be used to decrypt and re-encrypt files with a different certificate.

Manage local certificates and certificate signing requests

Use the *Certificate Management* page to generate local, self-issued server certificates or to generate certificate signing requests.

A certificate signing request is an unsigned copy of a certificate that you submit to a CA when requesting a signed certificate. Based on the information in the CSR, the CA creates a new signed certificate for your use. After receiving the signed certificate from the CA, you must import it into SecureTransport. For details, see *Import a new certificate for a pending certificate signing request* on page 51.

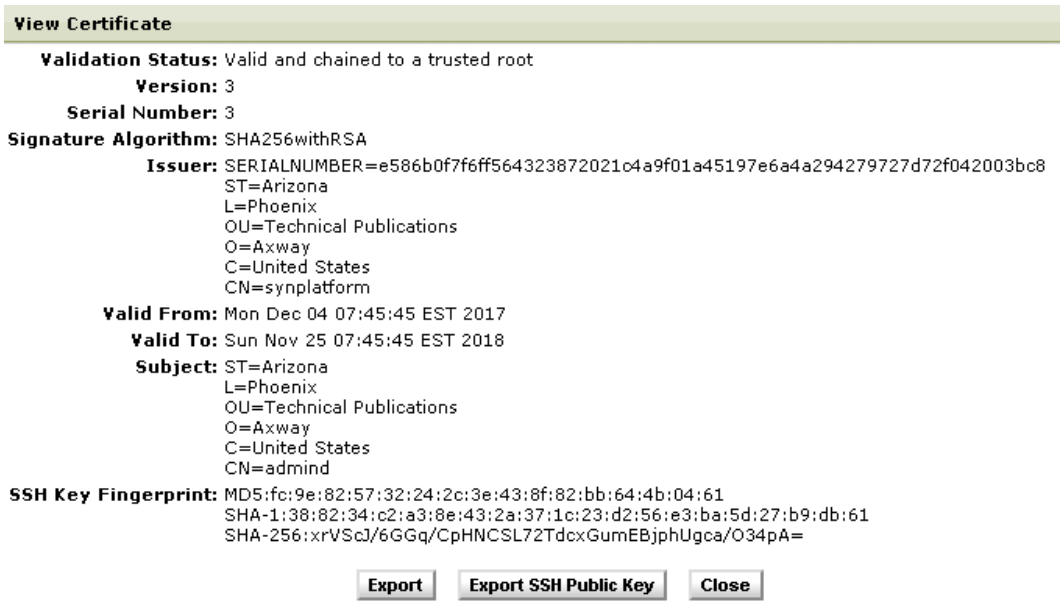
The following topics describe and provide how-to instructions managing local certificates and certificate signing requests:

View local certificates

You can view a list of all the local certificates or view the details of a specific certificate.

1. Select **Setup > Certificates**.
2. Click the **Local Certificates** tab.
3. Select a certificate alias from the list.

The *View Certificate* page displays the detailed information about the certificate.



Generate a self-issued server certificate

Use the following procedure to generate a self-issued server certificate.

1. Select **Setup > Certificates**.
2. Click the **Local Certificates** tab.
3. Click **Generate**.
4. Select **Self-issued Certificate**.

5. Type the required information for the self-issued certificate, including the fields that are displayed below the **Certificate Signing Request** option.

   The alias name should contain only lower case letters, digits, and hyphens (-). It must be at most 80 characters long. If the alias name you type is already assigned to another certificate, you are prompted to overwrite the existing certificate or cancel the operation.

   You can overwrite any existing certificate including the default SecureTransport `admind` server certificate.

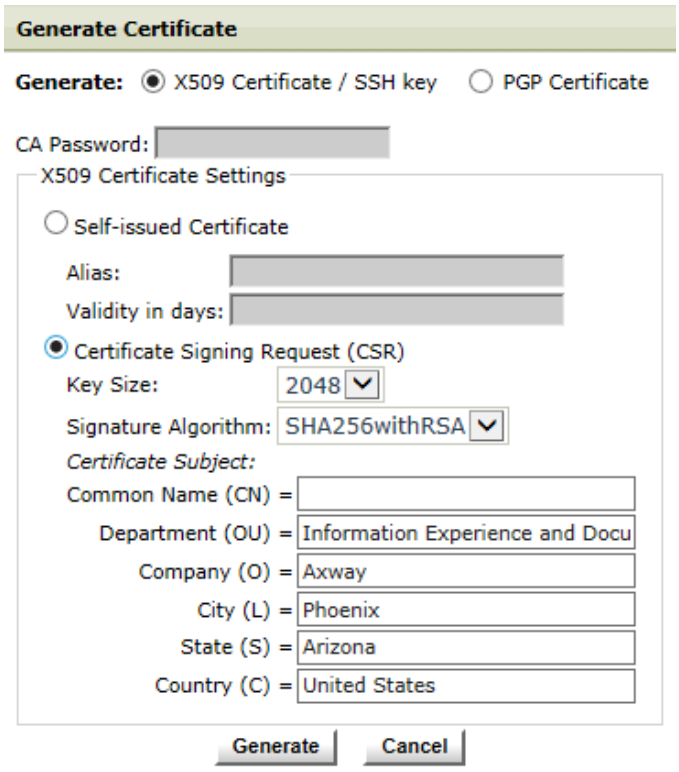
6. Click **Generate**.

Note When regenerating or overwriting the `admind` certificate or any certificate used in the daemons configuration, all services must be restarted.

Generate a certificate signing request

A certificate signing request (CSR) is a request to an external CA to sign a certificate.

1. Select **Setup > Certificates**.
2. Click the **Local Certificates** tab.
3. Click **Generate**.



4. Select **Certificate Signing Request (CSR)**.
5. Type the required information for the certificate signing request.

   The **Common Name (CN)** field must be the FQDN to be secured by the CA-signed certificate.
6. Click **Generate**.

   A message is displayed that explains how to download the CSR and send it to your CA.

The CSR is displayed in the list of Pending Local Certificates where it remains until you delete it or import the signed certificate you receive from the CA. You cannot use the new CA-signed certificate until you import it into SecureTransport.

Import a new certificate for a pending certificate signing request

Use the following procedure to import a new certificate for pending certificate signing request.

1. Select **Setup > Certificates**.
2. Click the **Local Certificates** tab.
3. Click **Finish** beside the appropriate CSR in the Pending Local Certificates list.
4. In the **Certificate Alias** box, type an alias.

5. Identify the certificate to import using one of the following methods:
    - Click **Import CA signed certificate from file** and specify the file name.
    - Click **Paste CA signed certificate in space below** and paste the certificate text in the box.
6. Click **Finish**.

    The certificate is displayed in the list of local certificates.

After you have imported the certificate, to use the certificate for a protocol server, set the Key Alias to the certificate alias and restart the protocol server. You can also use the certificate for other purposes, for example, as a login certificate for a transfer site or as an encryption or signing certificate for an AS2 transfer site.

Delete a pending certificate signing request

Use the following procedure to delete a pending certificate signing request.

1. Select **Setup > Certificates**.
2. Click the **Local Certificates** tab.
3. Click **Delete** beside the appropriate CSR in the Pending Local Certificates list.

Export a local certificate

Use the following procedure to export a local certificate.

1. Select **Setup > Certificates**.
2. Click the **Local Certificates** tab.
3. Click the alias of the certificate to export.

    The *View Certificate* window displays the certificate details.
4. To export a PGP certificate:
    a. Click **Export**.
    b. In the *Export Certificate* window, click **Export** to save the certificate as an ASCII-Armored (`.asc`) file, or select **Export private key**, type and confirm a password, and click **Export** to save the certificate as an `.asc` file with the private key.
5. To export a PGP certificate:
    a. Click **Export**.
    b. In the *Export Certificate* window, click **Export** to save the certificate as an ASCII-Armored (`.asc`) file, or select **Export private key**, type and confirm a password, and click **Export** to save the certificate as an `.asc` file with the private key.

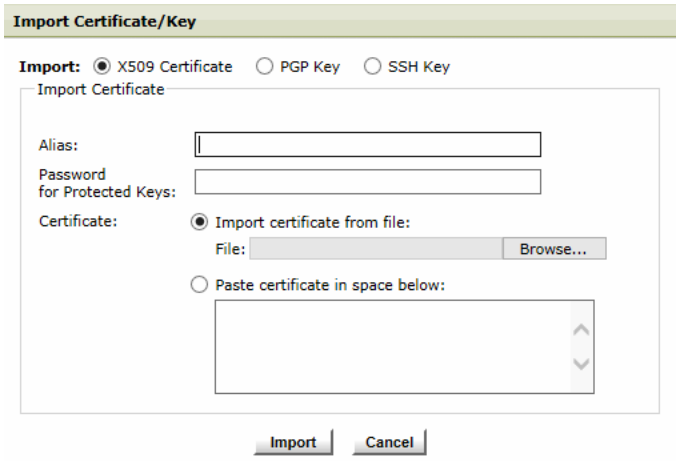
6. To export an X509 certificate:
    a. Click **Export**.
    b. In the *Export Certificate* window, click **Export** to save the certificate as a PEM-encoded certificate (`.crt`) file, or select **Export private key**, type and confirm a password, and click **Export** to save the certificate as a PKCS#12 (`.p12`) file.
7. To export an X509 certificate public key:
    a. Click **Export**.
    b. In the *Export Certificate* window, click **Export SSH Public Key** to save the public key as a `.pub` file.

Import a local certificate

You can import a private key and the corresponding certificate in **Local Certificates** as a PKCS#12 (`.p12`) file. You can then use it where you need a certificate that a remote client or system must trust, for example as the HTTPS SSL key alias.

1. Select **Setup > Certificates**.
2. Click the **Local Certificates** tab.
3. Click **Import**.

    The *Import Certificate/Key* window is displayed.



4. Select **X509 Certificate** or **PGP key**.
5. In the **Alias** field, type an alias name for the certificate.

    If the alias name is already assigned to another certificate, you are prompted to either overwrite the existing certificate or cancel the operation.
6. If the PKCS#12 (`.p12`) file being imported is password protected, type the password. You can also import PGP certificate files (`.asc` files).
7. Select **Import certificate from file** and type or browse to the file name or select **Paste**

certificate in space below and paste the certificate text into the text field.

8. Click **Import**.

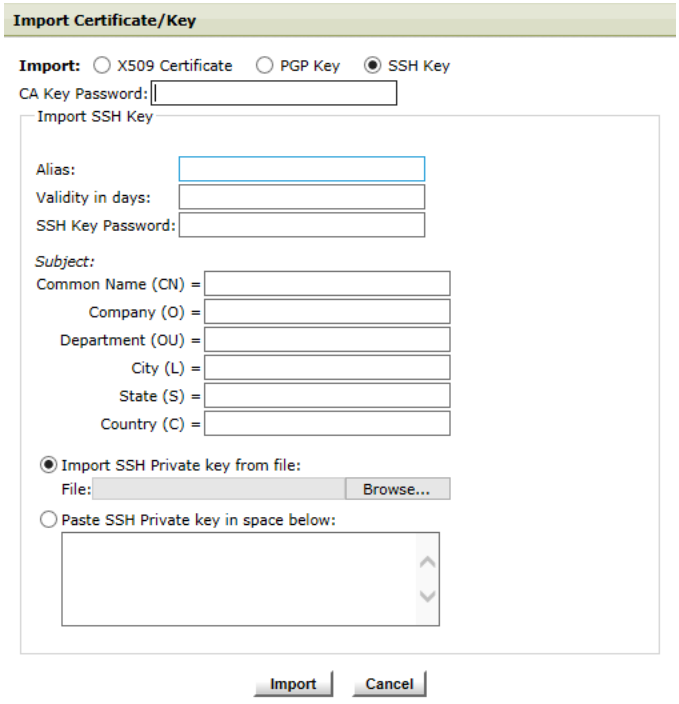
Note For specific requirements for certificates used to secure the SSL communication between the TM Server and the other servers, see *Secure the communication between the TM server and the protocol servers* on page 237.

Import an SSH key

Use the following procedure to import an SSH key.

1. Select **Setup > Certificates**.
2. Click the **Local Certificates** tab.
3. Click **Import**.

   The *Import Certificate/Key* window is displayed.
4. Select **SSH key**.



5. Type the **CA Key Password** specified during the certificate generation.

   Note Imported SSH Keys are stored as X509 certificates.

   Note **CA Key Password** is not a required field. When a SSH key is imported (without providing the internal CA key password), the key will be stored as X.509 certificate and signed with temporarily generated certificate. As a result, the SSH key will be stored as X.509 self-signed certificate.
6. Type the information necessary to import the key. **Alias** and **Validity in days** are required fields.

7. Paste the certificate content directly in the provided space, or import the certificate from a file. To import from file, type the file path or click **Browse** to browse for the file.
8. Click **Import**.

Delete a local certificate

Use the following procedure to delete a local certificate.

1. Select **Setup > Certificates**.
2. Click the **Local Certificates** tab.
3. Select the checkboxes for the certificates to delete or select the checkbox in the table header to select all certificates.
4. Click **Delete**.
5. Confirm the deletion.

Do not delete the certificate with the `admind` alias which is implicitly assigned to the Administration Tool server. SecureTransport prevents you from deleting a certificate that is in use, for example, a server certificate configured as the SSL key alias the AS2, FTP, HTTP, or SSH server or configured as the repository encryption certificate.

Manage trusted CAs

Trusted CAs represent the list of root and intermediate CAs used to build the certificate chain for client and server certificates.

The following topics provide how-to instructions for managing trusted CAs:

- *View a trusted CA certificate* on page 55
- *Export a trusted CA certificate* on page 56
- *Import a trusted CA certificate* on page 56
- *Delete a trusted CA certificate* on page 57

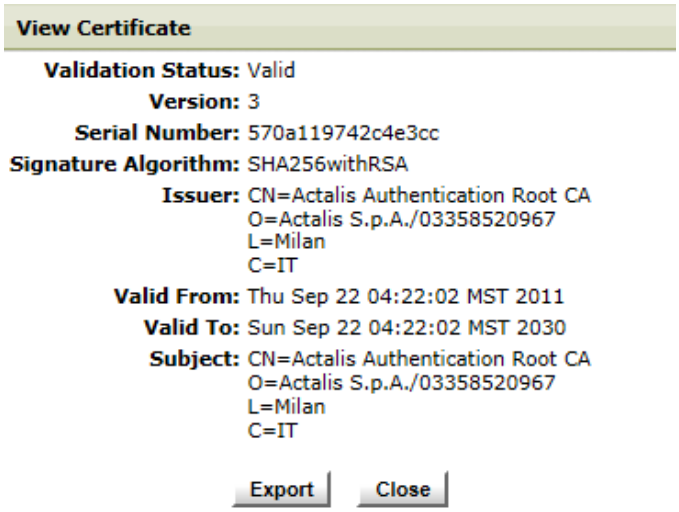
View a trusted CA certificate

Use the following procedure to view a trusted CA certificate.

1. Select **Setup > Certificates**.
2. Click the **Trusted CAs** tab.

   The list of trusted certificates is displayed.
3. Navigate to the page that lists the certificate to view.
4. Click the alias from the list.

The *View Certificate* page displays the detailed information about the certificate.



Export a trusted CA certificate

Use the following procedure to export a trusted CA certificate.

1. Select **Setup > Certificates**.
2. Click the **Trusted CAs** tab.
3. Navigate to the page that lists the certificate to export.
4. Click the alias of the certificate to export.
5. On the *View Certificate* page, click **Export** and save the certificate file in the desired location.

Import a trusted CA certificate

A X509 certificate can be imported as a trusted CA in the form of a X509 DER or PEM encoded file. SecureTransport checks the certificate fingerprint before importing. If a certificate with the same fingerprint already exists on the server, the new certificate won't be imported and a failure message will be logged in the Server Log.

Note SecureTransport protocol servers and services does not require restart after importing, overwriting, or deleting a trusted certificate.

1. Select **Setup > Certificates**.
2. Click the **Trusted CAs** tab.
3. Click **Import**.
4. Type an alias for the certificate in the **Alias** box.

If you use an alias that is already assigned to another certificate, the imported certificate overwrites the original one. Be sure that you are entering the appropriate alias for the new certificate.

5. Identify the certificate to import using one of the following methods:
    - Click **Import certificate from file** and type the file name.
    - Click **Paste certificate in space below** and paste the certificate text in the box.
6. Click **Import**.

Delete a trusted CA certificate

Use the following procedure to delete a trusted CA certificate.

1. Select **Setup > Certificates**.
2. Click the **Trusted CAs** tab.
3. Navigate to the page that lists the certificates to delete.
4. Select the checkboxes for the certificates to delete.
5. Click **Delete**.

Note If an end user has a certificate issued by a trusted CA that was deleted, the user can no longer authenticate using that certificate.

Manage the internal CA

Each SecureTransport installation maintains its own copy of the internal CA. During the initial post-installation setup procedure, an internal CA is generated. All SecureTransport Servers in a cluster share the same internal CA. If you have a disaster recovery site, the DR cluster should use the same internal CA, replicated from the primary production site.

Note For security reasons, client certificates should only be stored inside the SecureTransport home folder or within a sub-folder of the home folder.

The internal CA can be used to generate server or client certificates. It provides a convenient alternative to using a third-party CA. It is used implicitly in the following cases:

- Generating a local certificate. For detail, see *Manage local certificates and certificate signing requests* on page 48.
- Generating an account certificate. For detail, see *Manage login certificates* on page 527.
- Signing imported SSH Keys. For detail, see *Manage login certificates* on page 527.

The internal CA key is protected with a password. Any operation that involves use of the internal CA require that password. This password cannot be retrieved if it is lost. Contact Axway Global Support for more information. For contact information, see *Get more help* on page 27.

In addition to issuing certificates signed by internal CA, SecureTransport supports a few management operations for the internal CA. These operations include:

- Viewing the internal CA certificate
- Generating a new internal CA
- Importing an internal CA
- Exporting the internal CA

Note If you attempt to delete the internal CA, an alert dialog box is displayed.
If the certificate of the internal CA is deleted the Validation Status of all X509 local certificates, as well as imported SSH keys (because they are converted to X509 certificates during the import,) become `Not chained to a trusted root`. This has critical impact on the generated Login Certificates because users are no longer able to log into the SecureTransport Server.

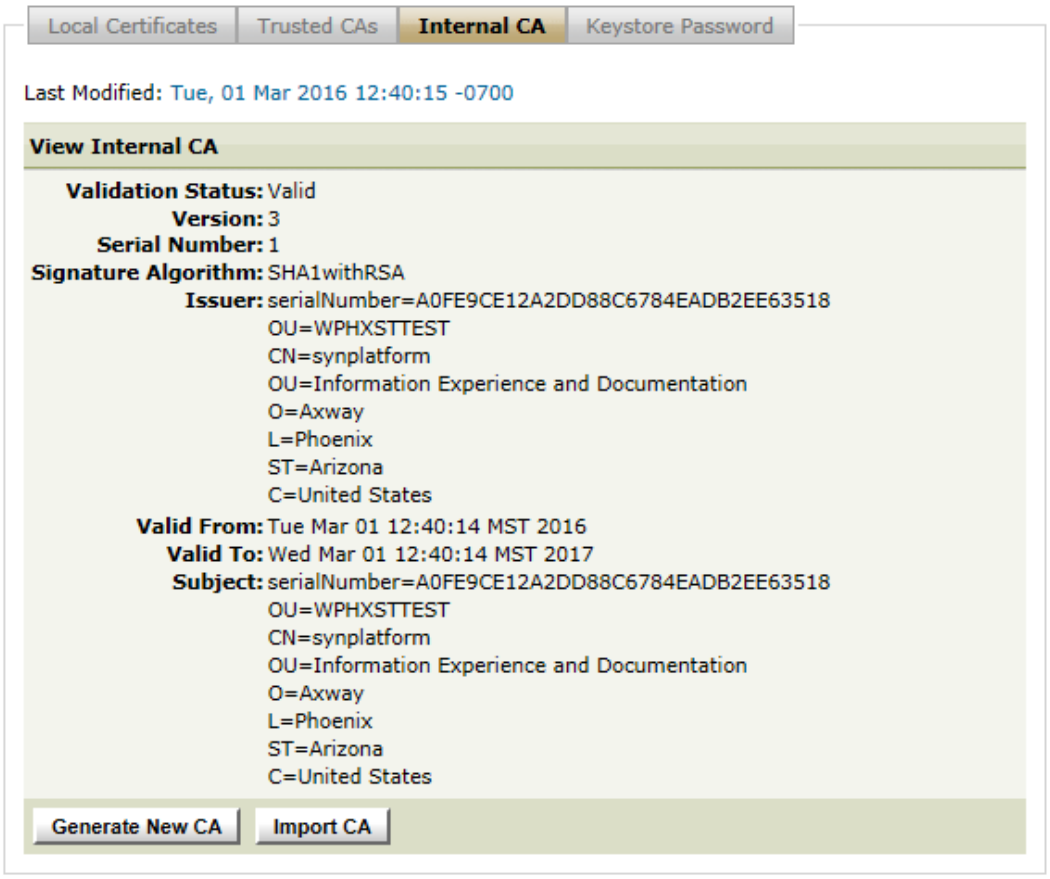
The following topics provide how-to instructions for managing the internal CA:

- *View the internal CA* on page 58
- *Generate an internal CA* on page 59
- *Import an external CA* on page 61
- *Export the internal CA* on page 62

View the internal CA

1. Select **Setup > Certificates**.
2. Click the **Internal CA** tab.

Basic certificate information is displayed including the validation status.

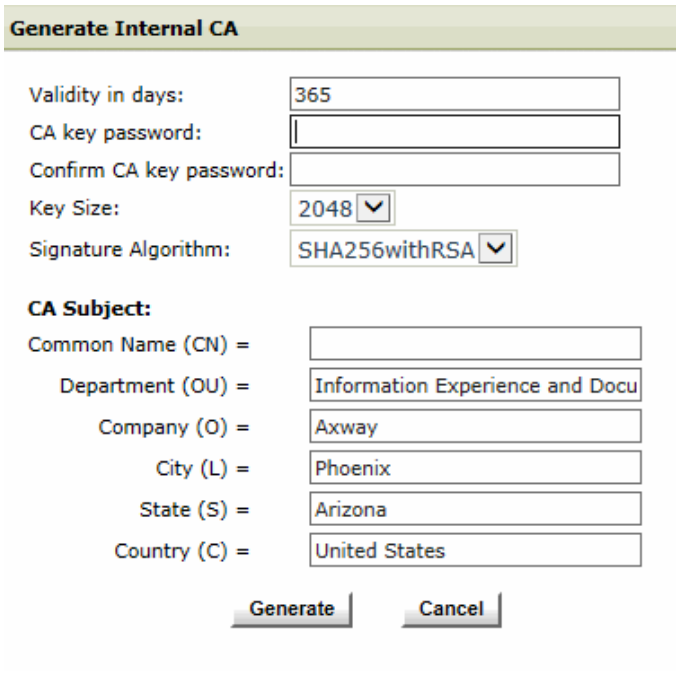


Generate an internal CA

There are a number of reasons why you might want to generate an internal CA. The most common reason is that the current CA is nearing its expiration date.

1. Select **Setup > Certificates**.
2. Click the **Internal CA** tab.
3. Click **Generate New CA**.

   The *Generate Internal CA* dialog box is displayed.



4. Provide the required information for the internal certificate.

   Internal certificates require the Certificate Subject information. For internal certificates, provide following information:

   - **Validity in days** – the number of days the certificate is valid.
   - **CA key password** – the password to protect internal CA. This password is requested when generating certificate signed by the internal CA.
   - **Confirm CA key password** – the key password must be entered again for confirmation.
   - **Key Size** – a number representing the size of the generated key, expressed in bits. Possible values are 1024, 2048 (default), 3072, or 4096 bits.
   - **Signature Algorithm** – the selection of the signature signing hashing algorithm. Possible values are SHA1withRSA, SHA256withRSA (default), SHA384withRSA, and SHA512withRSA.

     Note SHA1withRSA is available for backwards compatibility, but its usage is not recommended.

   - **Common Name** – the name that identifies the certificate.
   - **Department** – the name of department that the certificate is issued.
   - **Company** – the name of the company that the certificate is issued.
   - **City** – the name of the city where the location of the certification is located.
   - **State** – the name of the state where the location of the certification is located.
   - **Country** – the name of the country where the location of the certification is located.

5. Click **Generate**.

Generating a new internal CA does not automatically invalidate the certificate issued by the previous CA. When you generate an internal CA, SecureTransport adds the certificate to the Trusted CAs list under alias `ca`. The previous internal CA certificate is still in the Trusted CAs list under an alias of the form `ca-old-<serialNumber>`. When you do not want to accept certificates issued by the old internal CA, you can delete the `ca-old-<serialNumber>` aliases from the Trusted CAs list. For detail, see *Delete a trusted CA certificate* on page 57.

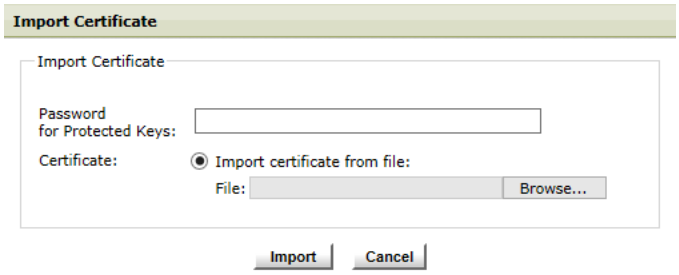
Once the new internal CA is generated, all certificates generated from that point on are signed by the new internal CA. Unless the new internal CA is added to the list of trusted certificates on the remote host, the host might reject the new certificates. An internal CA can be exported into a file that in turn can be used to add the CA to the list of trusted CAs.

Import an external CA

Optionally, you can also import an external certificate. Make sure the certificate is valid and configured to validate certificates before you import it. The CA attribute in the X509v3 extension section of the certificate must be true.

1. On the *Generate CA* page, click **Import CA**.

   SecureTransport displays *Import Certificate* page.



2. Type a password in the field provided. The password is required.

   If the CA certificate requires a pass phrase, SecureTransport uses this password. If the certificate does not require a pass phrase, the password is ignored. SecureTransport also uses this password to encrypt the CA private key in the keystore stored in the database and file system.

3. Specify the certificate by typing the path to the PKCS#12 (`.p12`) file in the field or by browsing to the file.

4. Click **Import**.

   SecureTransport reports if the import was successful.

Now, SecureTransport uses the imported certificate as the internal CA and signs all certificates generated using that CA. To make sure that a remote host accepts those certificates, add the certificate for this CA to the list of trusted certificates on that host. To export the certificate for import on another system, see *Export the internal CA* on page 62.

Generating or importing an internal CA does not automatically invalidate the certificate issued by the previous CA. When you generate or import an internal CA, SecureTransport adds the certificate to the Trusted CAs list under alias `ca`. The previous internal CA certificate is still in the Trusted CAs list under an alias of the form `ca-old-<serialNumber>`, where `<serialNumber>` is the value of the

certificate serialNumber field. When you do not want to accept certificates issued by the old internal CA, you can delete the `ca-old-<serialNumber>` aliases from the Trusted CAs list. For detail, see *Delete a trusted CA certificate* on page 57.

Once the internal CA is imported, all certificates generated from that point on are signed by the new internal CA. Unless the new internal CA is added to the list of trusted certificates on the remote host, the host might reject the new certificates. An internal CA can be exported into a file that in turn can be used to add the CA to the list of trusted CAs.

After you import a certificate authority, you might need to update the `CertificateStores.CertificateAuthority.serialNo` server configuration parameter. SecureTransport uses this parameter as a serial number and then increments it every time it generates a certificate. If the imported CA has been used by another SecureTransport server, set the parameter to the value from that server for consistency. To propagate the change, restart all SecureTransport servers in the cluster.

Export the internal CA

You can export the internal CA public certificate using the Administration Tool. You can also copy the certificate files from the server file system.

1. Select **Setup > Certificates**.
2. Click the **Trusted CAs** tab.
3. In the Alias column, click **ca**.

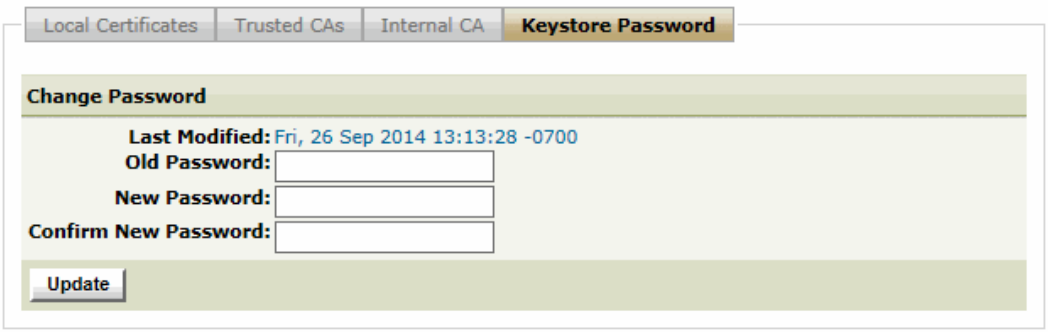
   The *View Certificate* page for the internal CA is displayed.
4. Click Export to export the public certificate for the internal CA as a `.crt` file.

Note The internal CA files are located in the `<FILEDRIVEHOME>/lib/certs/db` directory as `.pem` files. The public certificate is `ca-crt.pem`. The private key is `ca-key.pem`.

Change the certificate keystore password

SecureTransport stores all the available certificates for the system in the Certificate Keystore and the database.

1. Select **Setup > Certificates**.
2. Click the **Keystore Password** tab.



3. Type the old and new passwords, confirming the new one a second time. Leave the **Old Password** field empty if this is the first time you are changing the keystore password and SecureTransport uses the default.
4. Click **Update**.

Note Before you change the password the first time, you do not need to type the old password, because SecureTransport supplies the default value.

Certificates to generate during initial setup

For SecureTransport Edge and SecureTransport Server installations, generate an `admind` SSL server certificate for users connecting to the Administration Tool. This certificate may be signed by the internal SecureTransport CA.

Note If this certificate or any of the CA certificates in its certification paths (certificate chains) are expired or otherwise not valid, the Administration Tool server does not start.

To use repository encryption or MDN receipts, generate a repository encryption certificate or an `mdn` certificate, respectively. For information about the repository encryption certificate, see *Repository encryption* on page 46.

To be able to enable FTPS, HTTPS, AS2 using SSL, SFTP, SCP, or PeSIT over a secured socket, generate the required certificates.

When you set up FTPS, HTTPS, AS2 (SSL), SSH, PeSIT, or SecureTransport Edge communication with the Transaction Manager on the SecureTransport Server, you select a key alias to specify the certificate to use to secure the communications. You created the alias when you generated the certificate. For a list of certificates commonly used with SecureTransport, refer to the *SecureTransport Getting Started Guide*.

Note For more information about the post-installation setup process, refer to the *SecureTransport Getting Started Guide*.

Store certificates in a Luna hardware security module

You can store certificates in a Thales Luna hardware security module (HSM). As of SecureTransport 5.5-20230223, Luna is the default HSM. The following protocols are supported in SecureTransport: FTPS, HTTPS, and SSH.

The following sections provide detailed configuration information:

- *Install and configure the HSM* on page 64
- *Generate, sign and import an HSM certificate* on page 65
- *Use an HSM certificate for FTPS, HTTPS or SSH* on page 66

This procedure uses the following placeholders:

- <alias> – the SSL key alias for FTPS, HTTPS or SSH, for example ftpd, httpd or sshd
- <cert_file> is the file name of the PEM-format certificate file, for example, ftpd.pem, httpd.pem or sshd.pem
- <CSR_file> – the file name of the CSR request file, for example, ftpd.req, httpd.req or sshd.req
- <FILEDRIVEHOME> – SecureTransport installation directory, for example, /opt/TMWD/SecureTransport
- <key_size> – the key size, for example, 1024, 2048, 3072, or 4096
- <keystore_passphrase> – the passphrase for the HSM keystore
- <keystore_path> – the path to the SecureTransport HSM keystore
- <validity> – the validity of the certificate in days
- <luna_provider_path> - the path to the LunaProvider.jar file

Install and configure the HSM

1. Install the Luna HSM Client. See Add and Configure Client and Service Description. Note that the slot number is needed later in step 5.
2. Copy the Luna client content to SecureTransport:
   - On Unix, copy the LunaProvider.jar file from <HSM_client_folder>/jsp/LunaProvider.jar to <FILEDRIVEHOME>/lib/jars/external.
   - On Windows, copy the LunaProvider.jar file from <HSM_client_folder>\LunaProvider.jar to <FILEDRIVEHOME>\lib\jars\external.
3. Copy the Luna API file to SecureTransport:
   - On Unix, copy the libLunaAPI.so file from <HSM_client_folder>/jsp/64/libLunaAPI.so to /usr/lib.
   - On Windows, copy the LunaAPI.dll file from <HSM_client_folder>\LunaAPI.dll to C:\Windows\System32.
4. Load the HSM client environment on Unix:

   Run the source ./setenv command from the <HSM_client_folder>.
5. Create an hsm.keystore file with the content slot:<LunaSlot> (e.g., slot:3) in <FILEDRIVEHOME>/lib/certs.
6. Verify that the clocks on the SecureTransport Server and the HSM client machine are synchronized.
7. To test your setup, navigate to <FILEDRIVEHOME> and enter the following command which should list all available certificates:

```
jre/bin/keytool -list -v \
    -keystore <keystore_path> -storetype Luna \
    -providerclass com.safenetinc.luna.provider.LunaProvider \
    -providerpath <luna_provider_path> \
    -storepass <keystore_passphrase>
```

Generate, sign and import an HSM certificate

1. Make the SecureTransport installation directory the current working directory using the following command:

```
cd <FILEDRIVEHOME>
```

2. Generate an RSA key using the following command:

   Note Currently, only RSA keys are supported.

```
jre/bin/keytool -genkey -keyalg RSA -keysize <key_size> \
    -keystore <keystore_path> -storetype Luna \
    -providerclass com.safenetinc.luna.provider.LunaProvider \
    -providerpath <luna_provider_path> \
    -alias <alias> -storepass <keystore_passphrase>
```

3. Generate a certificate signing request (CSR) using the following command:

```
jre/bin/keytool -certreq \
    -keystore <keystore_path> -storetype Luna \
    -providerclass com.safenetinc.luna.provider.LunaProvider \
    -providerpath <luna_provider_path> \
    -alias <alias> -storepass <keystore_passphrase> \
    -file <CSR_file>
```

4. Sign the certificate and create the PEM-format certificate file using the following command:

```
openssl x509 -req -in <CSR_file> -days <validity> \
    -CA lib/certs/db/ca-crt.pem -CAkey lib/certs/db/ca-key.pem \
    -CAserial lib/certs/db/serial -out <cert_file>
```

5. Append the public part of the internal CA to the certificate file using the following command. This is required so that SecureTransport can build the certificate chain.

```
cat lib/certs/db/ca-crt.pem >> <cert_file>
```

6. Import the signed certificate into the HSM using the following command:

```
jre/bin/keytool -importcert -file <cert_file> \
    -keystore <keystore_path> -storetype Luna \
    -providerclass com.safenetinc.luna.provider.LunaProvider \
    -providerpath <luna_provider_path> \
    -alias <alias> -storepass <keystore_passphrase>
```

Use an HSM certificate for FTPS, HTTPS or SSH

1. Specify the HSM for SecureTransport by setting the following server configuration parameters:
    - Set `Hsm.keystorePath` to the location of the SecureTransport HSM keystore relative to `<FILEDRIVEHOME>`.
    - Set `Hsm.keystorePassword` to the keystore passphrase.
2. Enable HSM for the protocol servers by setting the following server configuration parameters:
    - Set `Ftp.Hsm.enabled` to `true` to enable HSM for the FTP Server
    - Set `Http.Hsm.enabled` to `true` to enable HSM for the HTTP Server
    - Set `Ssh.Hsm.enabled` to `true` to enable HSM for the SSH Server
    - Set `Hsm.Type` to `Luna`
3. Create a local certificate with the same alias as the HSM certificate you created in the previous section, for example, `ftpd`, `httpd` or `sshd`. See *Generate a self-issued server certificate* on page 49. SecureTransport will take that alias and look in the HSM keystore instead of its local one, as long as any of the options `<PROTOCOL>.Hsm.enabled` is set to `true`.
4. Set the SSL key aliases for the protocol servers. See *Add an FTP Server* on page 264, *Add an HTTP server* on page 267 and *Add an SSH server* on page 277.
5. Restart the protocol servers. In a cluster environment, they must be restarted on every node. See Start and stop servers.

Secret vault integration

On startup, all SecureTransport services use the SecureTransport secret to access the database. By default, the secret is stored in the local `<FILEDRIVEHOME>/bin/taeh` file. As of Update 5.5-20240829, it can be stored in a *HashiCorp Vault*.

Vault integration is supported on:

- existing deployments - see the procedure below.
- fresh installations - see *Secret file and vault integration*.

Move the secret to an external vault

Perform the following steps to migrate to an external secret. Repeat the procedure on all SecureTransport nodes in your deployment.

1. Make sure your HashiCorp Vault is set up and ready to use.
2. Import the content of the existing `taeh` file in the vault by following the *procedure*.
3. Obtain the *secret_id* and *role_id* as described in *Set up application authentication*.

4. Make a copy in a temporary location of one of the two template files in the `<FILEDRIVEHOME>/conf/vault` directory:
   - `hashicorp-vault-conf-template.properties` - for the default HashiCorp Vault setup.
   - `vault-conf-template.properties` - for a more advanced or non-standard vault setup.
5. Edit your custom `.properties` file as described in *Configure the vault client connection*.
6. Save the content of the `taeh` file as it can be deleted irreversibly in the next step.
7. Run the *update_taehlocation* script to validate the edited file:

   ```
   <FILEDRIVEHOME>/bin/update_taehlocation -f
   <FILEDRIVEHOME>/<your.properties.file> --backup
   ```

   The `--backup` parameter is optional and renames the `taeh` file to `taeh.backup`.

   Caution If you skip `--backup`, the content of the `taeh` file will be lost.

   The script checks if the secret can be retrieved from the vault, and compares it to the content of the binary `taeh` file. If the retrieved secret matches the current `taeh` content, the script creates a new *<FILEDRIVEHOME>/conf/taeh.properties* file based on the one you provided, and deletes the original `taeh` file. If the secret cannot be retrieved or it does not match, the vault integration errors out with either of the following:
   - `Cannot read secret from the specified vault. Check your vault configuration.`
   - `Secret does not match.`
8. Restart all SecureTransport services.

Vault configuration changes

To make changes to an already operational vault configuration:

1. Edit a copy of the *<FILEDRIVEHOME>/conf/taeh.properties* file.
2. Run the *update_taehlocation* script to validate your changes. Restart is not required.

Alternatively, for minor changes, you can edit the *<FILEDRIVEHOME>/conf/taeh.properties* file directly and restart all SecureTransport services.

Switch back to an internal secret

Perform the following steps to switch from storing the secret in an external vault to storing it in a local `taeh` file. Repeat the procedure on all nodes in your cluster, and make sure to use the same `taeh` file every time.

1. Locate your saved copy or backup of the previously used `taeh` file.

   If your SecureTransport installation was initially configured to store the secret in an external vault, no previous `taeh` file will be available. In that case, you must obtain and decode the secret from the vault, and save it in a binary (`taeh`) file in a temporary location. You can do so by running a command like the following:

   ```
   curl -k
   https://<vaulthost>:<port>/v1/st/data/taeh
   -H "X-Vault-Token:<token>" | grep -o '"secret":"[^"]*' | grep -o '[^"]*$' |
   base64 --decode > taeh.bin
   ```

2. Run the *update_taehlocation* script and specify the path to your `taeh` file:

   ```
   <FILEDRIVEHOME>/bin/update_taehlocation -f <FILEDRIVEHOME>/<your.taeh.file>
   ```

   The script compares the content of the binary `taeh` file to the secret in the vault. If it matches, the script deletes the vault configuration *taeh.properties* file and creates *<FILEDRIVEHOME>/bin/taeh* in its place, which SecureTransport starts using right away. A "*Secret does not match.*" error is displayed if if the `taeh` content does not match the retrieved secret.

3. (Optional) Remove the secret from the vault.

PGP key encryption and signing

SecureTransport supports PGP encryption. The system handles the generation and storing of PGP keys and the management of the stored keys. The PGP keys are managed from the Administration Tool.

PGP signature verification, encryption, and signing do not work properly when the PGP key has expired. However, SecureTransport continues to decrypt files successfully even when the PGP key has expired.

PGP encryption and signing only support ZIP, BZIP2, and ZLIB compression.

PGP verification is performed using the current time on the computer, not the time when signing occurred.

The following topics describe how to manage the PGP keys and provide the PGP key transfer dependencies:

- *Manage PGP keys* on page 68
- *PGP transfer settings dependencies* on page 70

Manage PGP keys

PGP keys are managed from the Administration Tool according to their scope: *local* or *partner*.

- *Local PGP keys* can be either server-scoped or account-based. To manage server-scoped certificates select **Setup > Certificates** and the *Local Certificates* tab. Account-based certificates are managed from the *Private Certificates* pane of the *Certificates* pane of the *User Account* page. You can use the appropriate page to get detailed information for a particular key, generate a key, delete a key, or export public and private keys.
- *Partner PGP keys* are also managed by account. You can generate, import, delete, and export account-specific PGP keys. For partner PGP keys, the private key can be exported on certificate creation only. For more information on managing account-based PGP keys, see *Use the following procedure to import a partner certificate.* on page 533

The following topics provide how-to instructions for managing PGP keys:

- *Generate PGP keys* on page 69
- *Export PGP keys* on page 69
- *Import PGP keys* on page 69
- *Local (server) PGP keys management* on page 70

Generate PGP keys

The two scenarios for generation of PGP keys are:

- Generate a local key pair and export it into a text file that can be used by the respective partners for encryption of the incoming data. You can export the private key for a local or account private certificate at any time.
- Generate a partner key pair and export the private key in a file that can be used by the partner for decryption and signing. In this case only the public key is saved and used by SecureTransport. You can save the private key immediately after you generate the key pair. SecureTransport deletes the private key after you save it or decline to save it.

Note Partner and local PGP keys are stored in different key rings. Partner keys are account-specific and have a relation to the account, while local keys are not connected to a particular account. To this end, all partner keys are stored in a single key ring and all partner private keys are stored in a corresponding single secret key ring.

Export PGP keys

PGP keys are exported in the format of ASCII-armored key data in compliance with RFC 2440. The data of the exported PGP key is stored directly in a file. The private key of a partner key pair can only be exported when the certificate is first generated by SecureTransport.

Import PGP keys

There are three scenarios for importing PGP keys into SecureTransport:

- Import a partner PGP public key
- Import a PGP key pair, generated by a third party, in the set of local keys that applies to all of the SecureTransport system
- Import a local PGP key pair as a private certificate from the *Private Certificates* pane of the *Certificates* pane of the *User Account* page

Note The supported formats of the imported keys are as specified in the RFC 2440 standard: a binary or armored PGP public/private key message.

Local (server) PGP keys management

Local (server) and partner PGP keys are generated, exported, imported, and deleted for a particular account. See *User certificates* on page 526

User certificates on page 526

PGP transfer settings dependencies

The following PGP transfer settings dependency matrix shows different configurations and scenarios for PGP-encrypted transfers depending on the transfer settings, accessible from the *Subscription* page.

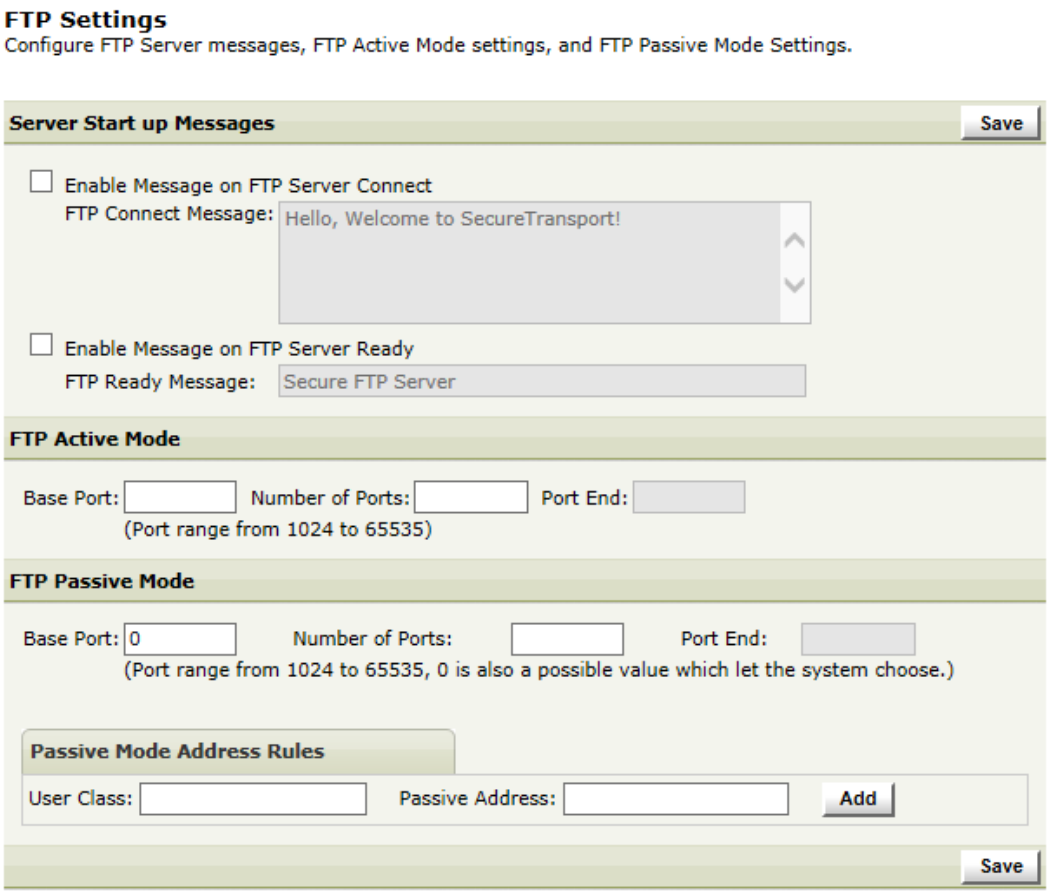
Incoming file	Encryption required	Signature required	Server action	Result	Transfer status
Signed only	ON	ON	SecureTransport attempts decryption and verification.	Decryption fails, because the file is not encrypted, and therefore no verification is performed.	Unsuccessful
Encrypted only	ON	ON	SecureTransport attempts decryption and verification.	Decryption is successful, but verification fails, because the file is not signed.	Unsuccessful

Incoming file	Encryption required	Signature required	Server action	Result	Transfer status
Signed and Encrypted	ON	ON	SecureTransport attempts decryption and verification.	Decryption and verification are successful, and the signature is removed.	Successful
Plain text	ON	ON	SecureTransport attempts decryption and verification.	Decryption fails because the file is not encrypted, and no verification is performed.	Unsuccessful
Signed only	ON	OFF	SecureTransport attempts decryption without verification.	Decryption fails because the file is not encrypted, and no verification is performed.	Unsuccessful
Encrypted only	ON	OFF	SecureTransport attempts decryption without verification.	Decryption is successful, and no verification is performed.	Successful
Signed and Encrypted	ON	OFF	SecureTransport attempts decryption without verification.	Decryption is successful, and no verification is performed, but the signature is removed.	Successful

Incoming file	Encryption required	Signature required	Server action	Result	Transfer status
Plain text	ON	OFF	SecureTransport attempts decryption without verification.	Decryption fails because the file is not encrypted, and no verification is performed.	Unsuccessful
Signed only	OFF	ON	SecureTransport attempts verification without decryption.	No decryption is performed. Verification is successful and the signature is removed.	Successful
Encrypted only	OFF	ON	SecureTransport attempts decryption (to get to the signature) and verification.	Decryption is successful, but verification fails, because the file is not signed.	Unsuccessful
Signed and Encrypted	OFF	ON	SecureTransport attempts decryption (to get to the signature) and verification.	Decryption and verification are successful.	Successful
Plain text	OFF	ON	SecureTransport attempts verification without decryption.	No decryption is performed. Verification fails because the file is not signed.	Unsuccessful

Configure FTP server messages and modes

Use the *FTP Settings* page to enable and disable FTP server messages and to configure active mode and passive mode for the FTP server.



The following topics describe FTP server messages, provide how-to instructions for setting FTP active and passive modes, provide FTP server limitations, and describes how to improve FTP performance and increase the FTP timeout for large files:

- *FTP server messages* on page 73
- *Set up FTP active mode* on page 75
- *Set up FTP passive mode* on page 75
- *FTP server limitations* on page 77
- *Improve FTP performance on a multi-homed system* on page 78
- *Increase the timeout for large files using server-initiated transfer* on page 79

FTP server messages

FTP server messages are messages displayed to users at the startup of an FTP session. Two types of startup FTP server messages are available:

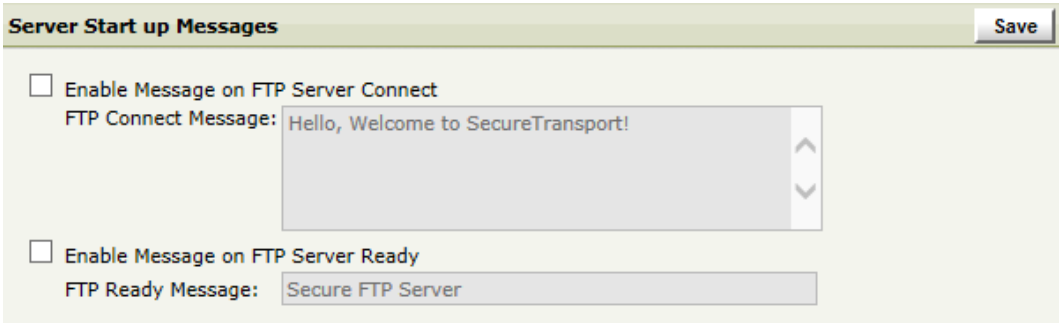
- *Create or edit the server connect message* on page 74
- *Create or edit the server ready message* on page 74

Users might or might not see FTP server messages, depending on the software they use to connect to SecureTransport. Most FTP client applications display these messages, but they are not displayed on browser clients. Users downloading a single file by URL from a browser might not see these messages.

Startup messages are displayed while a connection to the server is being established, prior to any login prompt. A message is displayed to the user when the connection is established. A server ready message also be displayed when SecureTransport is ready to accept a login. The default message displays the host name of the server and the server version, but this can be overridden.

Create or edit the server connect message

1. Select **Setup > FTP Settings** to open the *FTP Settings* page.
2. Under *Server Start up Messages:*



   a. Select **Enable Message on FTP Server Connect** to display a message to users logging onto SecureTransport through the FTP Server. By default, this option is not selected.

   b. In the **FTP Connect Message** field, type the message you want to be displayed when a user connects to the SecureTransport Server

3. Click **Save** to apply the changes.

Create or edit the server ready message

1. Select **Setup > FTP Settings** to open the *FTP Settings* page.
2. Under *Server Start up Messages:*

   a. Select **Enable Message on FTP Server ready** to display a message indicating that the server is ready to accept logins. By default, this option is not selected.

   b. In the **FTP Ready Message** field, type the message you want to be displayed when the server is ready.

3. Click **Save** to apply the changes.

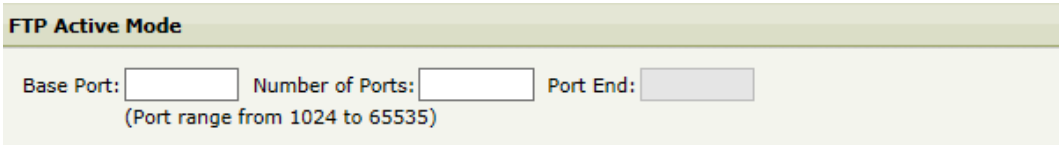
Set up FTP active mode

You can configure SecureTransport to use active mode connections for server-initiated transfers over FTP. Use the active mode options to configure the server-initiated transfer agents to initiate these requests. You can specify an active mode base port and a range of ports that can be used for active mode. Make sure the ports are accessible on the firewall.

The base port represents the first port number you want to assign. The number of ports specifies how many open ports you want to allow. For example, the base port is set to 10000 and the number of ports is set to 1024, active mode connections can use only the ports from 10000 to 11023. The end port should display 11023. Acceptable values are between 1024 and 65535. If the range is not specified, the server randomly selects an unused high port number greater than 1023 for active mode.

Active FTP mode can be used only for internal connections where there is no proxy between SecureTransport and the remote server. You must select **Enable Active Connection Mode** to use Active FTP in an FTP transfer site.

1. Select **Setup > FTP Settings** to open the *FTP Settings* page.
2. Under *FTP Active Mode*:



   a. Type the appropriate port number for the **Base Port**. This is the first port used.
   b. Type the **Number Of Ports**. This value sets the range of ports available for use.

      The **Port End** field should display the appropriate port number for the last port in the range you want to use.
3. Click **Save** to apply the changes.

Note To use Active FTP with an FTP(S) transfer site, by pass the proxy by setting the **Network Zone** field to **none**. See *FTP(S) transfer sites* on page 562.

Set up FTP passive mode

If an FTP client is behind a firewall that does not permit SecureTransport to open a data port as required by active mode FTP, you use the passive mode options to configure the FTP server to accept passive mode FTP connections. You specify a passive mode base port and a number of ports that can be used for passive mode. Make sure the ports are accessible through the firewall. Do not use these ports for any other service.

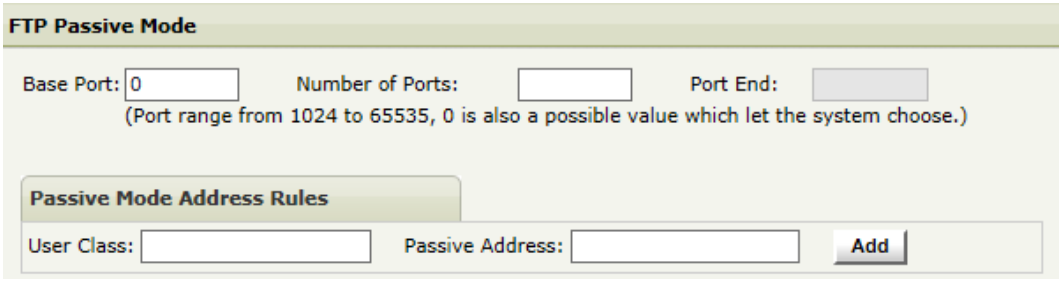
The base port represents the number of the first port available for the FTP server to use for a passive mode data connection. The number of ports specifies how many open ports to allocate. For example, the base port is set to 10000 and the number of ports is set to 1024, the FTP server can use only the ports from 10000 to 11023 for passive mode data connections.

Specify ports for five times the expected number of concurrent connections based on the following considerations:

- If the port range is too small or equal to the number of expected number of connections, a port to connect might not be available when needed. The operating system does not release a port immediately after a user disconnects, so the free port does not become available for another connection immediately. This can result in a failure when a new connection is attempted.
- When most of the ports are used up and a new connection is attempted, the server scans for the next available port. This is resource intensive and can affect the performance of the server. With a large pool of ports, finding next available port is quicker and is less likely to affect the performance.

You can also set up passive address rules, which allow the SecureTransport server to respond with the external address of the firewall instead of the internal address of the server when a FTP client issues the `PASV` (passive mode) command.

1. Select **Setup > FTP Settings** to open the *FTP Settings* page.
2. Under *FTP Passive Mode*:



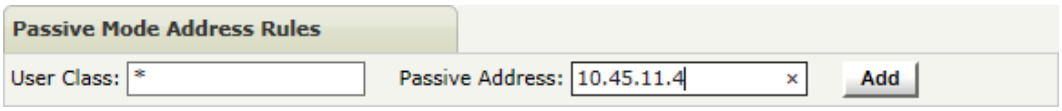
   a. Type the appropriate port number for the **Base Port**. This is the first port used.
   b. Type the **Number Of Ports**. This value sets the range of ports available for use.
3. The **Port End** field displays the number for the last port to use.
4. Click **Save** to apply the changes.

The following sections provide how-to instructions for adding and editing a passive mode address rules and insuring that passive mode sessions are initiated on the correct system:

- *Add passive mode address rules* on page 76
- *Edit a passive mode address rule* on page 77
- *Assure that passive mode sessions are initiated on the correct system* on page 77

Add passive mode address rules

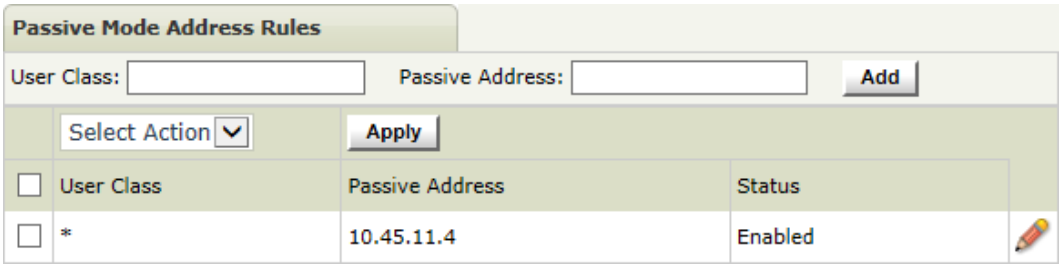
1. Under *Passive Mode Address Rules*, enter in the **User Class** field a pattern that matches the user classes for which this rule will be applied. Use question mark (`?`) to match one character and asterisk (`*`) to match any sequences of characters. For example, enter `*` to apply the rule to all user classes.
2. Type the external address with which the FTP server should respond in the **Passive Address** field.



3. Click **Add** to add the rule. To make the change permanent, click **Save**.

Edit a passive mode address rule

1. To change the **User Class**, **Passive Address** or **Status** of a rule, click the Edit icon ( ) at the end of the rule.



2. Make the required changes in the fields and click the Save icon ( ).
3. To change the status of one or more rules or to remove a rule entirely, select in the left column each rule to modify.
4. From the **Select Action** list, choose **Delete**, **Enable**, or **Disable** and click **Apply** to make the change.
5. To make your updates on this page permanent, click **Save**.

Assure that passive mode sessions are initiated on the correct system

When clients open passive mode sessions though a load balancer, a client communicating on the server assigned port might create a new session which opens a port on a different system than the originally contacted one. You can prevent this issue by configuring the load balancers and servers.

- Set the load balancer to use "session sticky" mode. This ensures that the data-channel connection goes to the same system that serviced the control-channel connection.
- Make sure to set a passive-mode address rule on all the SecureTransport Edge gateways or servers to return the external IP address of the load balancer in the response to the PASV command instead of the internal address of the SecureTransport Edge.

FTP server limitations

The following sections describe the FTP server limitations:

- *FTP APPE command* on page 78
- *FTP COMB command* on page 78

- *FTP client configuration* on page 78

FTP APPE command

SecureTransport handles the FTP `APPE` command differently than it is typically handled. Typically, the `APPE` command appends uploaded content to an existing file of the same name. However, when SecureTransport receives this command, it overwrites the server's copy of the file with the client's copy.

FTP COMB command

SecureTransport supports the `COMB` command only for the CuteFTP client. The CuteFTP client separates the file into chunks and uploads them as temporary files on the FTP server. Once CuteFTP uploads all the file chunks are uploaded on the FTP server, it sends the `COMB` command to combine them. By default, file names are generated by the CuteFTP client to match the pattern: `\d.tmp` where `\d` is a decimal number. However, the first file chunk name is the original file name.

Note The `COMB` command cannot be combined with post-transmission actions, repository encryption, or PGP data transformations.
Because SecureTransport releases the FTP command channel after successfully uploading the multiple file chunks, using the `COMB` command does not provide a guaranteed file transfer. If there is no space left in the upload directory or if errors exist during the process of combining the file chunks, the transfer fails, but no errors are sent to the client.

FTP client configuration

Some settings for LIST arguments need to be configured manually. The `Resolve Links [-L]` argument must be removed for both clients, but leave the show all file `[-a]` argument.

Improve FTP performance on a multi-homed system

You can improve FTP performance by assigning an IP address to the server if you are using a multi-homed system.

1. Select **Operations > Server Configuration**.
2. In the *Server Configuration* page, search for the `Ftp.Host` parameter.
3. Set the value of the parameter to the IP address for the server.
4. Restart the FTP Server on all servers in your cluster.

Increase the timeout for large files using server-initiated transfer

When sending 1.5 GB or larger files, the default timeout settings might be too low and SecureTransport might throw an exception error. You can increase the timeout settings for the FTP(S) and HTTP(S).

1. Select **Operations > Server Configuration**.

   The *Server Configuration* page is displayed.

2. Search for the `OutboundConnections.connectTimeout` and `OutboundConnections.receiveTimeout` parameters.

   The default value for each parameter is 25 seconds.

3. Set the value of each parameter to the amount of time it takes to upload the file in seconds. For example, if the file takes 20 minutes to upload, then increase the value of each parameter to 1200 seconds or higher.

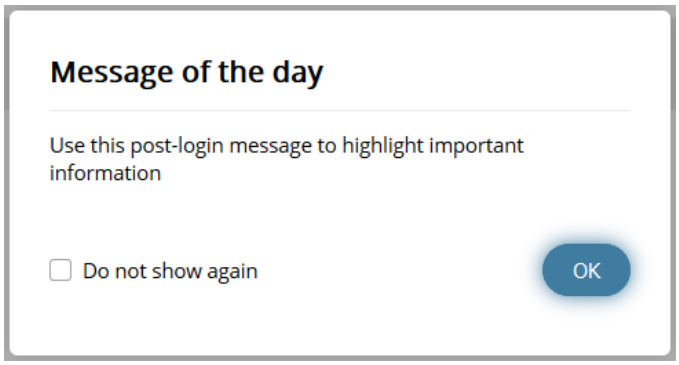
4. Restart the TM Server on all servers in your cluster.

Configure HTTP server messages

Use the *HTTP Settings* page to enable and disable HTTP server messages and to configure the Message of the Day functionality. HTTP server messages are displayed to both licensed and unlicensed users after login using the ST Web Client. You can use these messages as notifications, for example, announce dates for planned maintenance. The message text is presented in a pop-up box with a **Do not show again** checkbox and an **OK** button.

The Message of the Day pop-up behavior depends on the current selection of the **Do not show again** checkbox:

- If selected, the message will not appear again unless you modify it or disable and then re-enable the functionality.
- If not selected, the message will be displayed every time the user logs in or refreshes the page after login.



Set up Message Of The Day

1. Select **Setup > HTTP Settings** to open the *HTTP Settings* page.

2. Under *Message Of The Day:*
   a. Select **Enable Message of the day functionality** to display a message after the user logs in. By default, this option is not selected.
   b. In the **Message of the day content** field, type the message to display to your users.

Note The message consists of both plain text and HTML content, including links and images. The allowed HTML tags are: `<a>`, `<img>`, `<p>`, `<div>`, `<span>`, `<br>`, `<b>`, `<strong>`, `<i>`, `<u>`.

3. Click **Save** to apply the changes.

Note Message Of The Day can be disabled from the ST Web Client even if it is enabled on the server. For additional information, refer to *ST Web Client Configuration Guide*.

Configure the AS2 server settings

In order to manage the AS2 settings, an administrator must have access to the following pages: *Server Configuration* and *AS2 Settings*. The access rights are defined through *Administrative roles* on page 711.

The following AS2 settings apply to all partnerships:

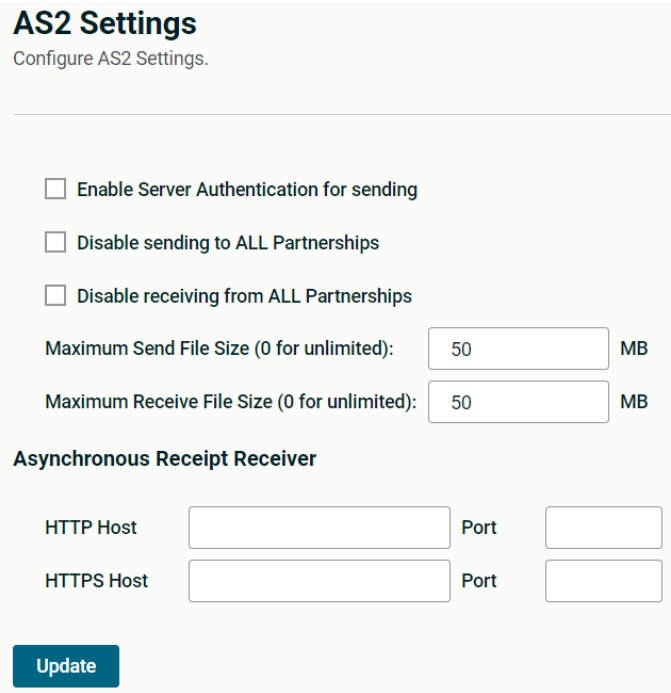
- Server authentication (enable or disable). By default, server authentication is disabled.
- Sending to and receiving from all partnerships (enable or disable). By default, sending and receiving is enabled for all partnerships.
- Maximum file sizes for sending and receiving. The default maximum file sizes are 50 megabytes.
- Asynchronous Receipt Receiver for HTTP and HTTPS.

For detailed information, see *AS2 transfers on page 1035*.

Configure AS2 transfer settings

Use the following procedure to configure the AS2 transfer settings.

1. Navigate to **Setup > AS2 Settings**.



2. Select the **Enable Server Authentication for sending** checkbox to request server authentication for outbound transfers.

   Note If server authentication is enabled, the local SecureTransport AS2 server authenticates the remote (partner) server during the SSL connection. This is an additional layer of security provided by AS2. To be authenticated, the remote server must present a certificate signed by a certificate authority that is trusted by SecureTransport. For information about importing trusted CA certificates for indirect trust of partner server certificates, see *Import a trusted CA certificate* on page 56.

3. Select the **Disable sending to ALL Partnerships** checkbox to turn off all outbound transfers to AS2 partner sites.
4. Select the **Disable receiving from ALL Partnerships** checkbox to turn off all inbound transfers from AS2 partner sites.
5. Set the **Maximum Send File Size** for outbound traffic. The file size is measured in MB.

   To allow unlimited file sizes, enter 0 (zero). However, to transfer files larger than 2 GB, the remote partner site must support chunking and have it enabled in its Send Options.
6. Set the **Maximum Receive File Size** for inbound traffic. The file size is measured in MB.

   To allow unlimited file sizes, enter 0 (zero).

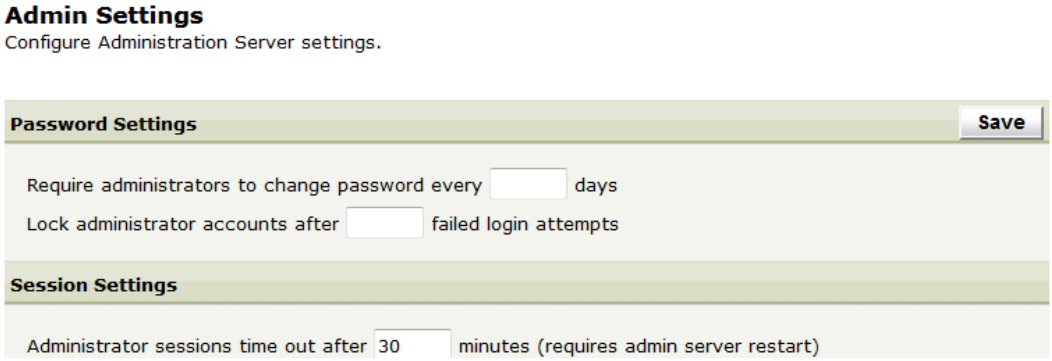
7. Enter host names and port numbers for the HTTP and HTTPS protocols for the **Asynchronous Receipt Receiver**.

   Outgoing AS2 messages are sent by the SecureTransport Server. If an asynchronous receipt is requested from a partner, the partner server tries to reach the AS2 port on the SecureTransport Server. In a two-layer architecture, asynchronous receipts should be delivered to SecureTransport Edge. In that case, host and port numbers in these fields should be set to AS2 server on SecureTransport Edge.

8. When done, click **Update**.

Configure Administration Tool server settings

Use the *Admin Setting* page to configure administrator login options. You can configure the password expiration, the number of consecutive failed login attempts allowed and the length of a login session.



The following sections provide how-to instructions for changing password settings, session settings, and certificate settings:

- Change password settings
- Change session settings

Change password settings

Use the password settings to control how often an administrator needs to change the login password and to control how many consecutive failed login attempts are allowed before the administrator is prevented from logging in.

1. Select **Setup > Admin Settings**.
2. Type the number of days the administrator password is valid without changing it.

   You can leave this field blank, which is the default setting. When left blank, the password never expires.

3. Type the number of consecutive failed login attempts to allow before preventing an administrator from logging in to the server.

You can leave this field blank, which is the default setting. When left blank, there is no limit to the number of login attempts permitted.

4. Click **Save** to accept the changes.

Note If the master administrator is locked out (or the wrong password is used too many times), contact Axway Global Support for the procedure to reset the required parameters.

Change session settings

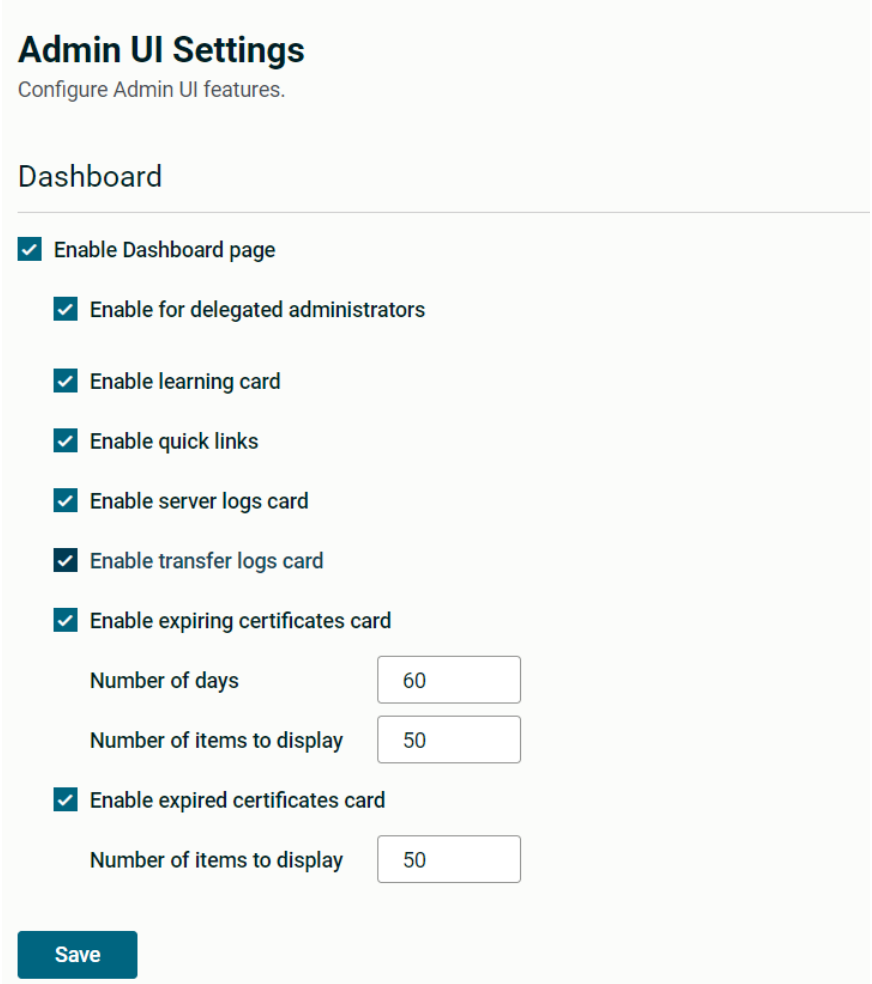
Use the session settings to control how long an administrator can be logged into SecureTransport before the session expires and the administrator must log in again.

1. Select **Setup > Admin Settings**.
2. Type the number of minutes the session stays active. The default setting is 30 minutes.
3. Click **Save** to accept the changes.
4. Restart the Administration Tool server using the `stop_admin` and `start_admin` commands.

Manage the Dashboard page

The *Dashboard* page is the entry point for SecureTransport master administrators upon login. It is available on both SecureTransport Server and Edge, and contains quick links to all SecureTransport configuration menus, a search functionality, and a display of alerts and warnings.

Master administrators can manage the *Dashboard* content from **Setup > Admin UI Settings**.



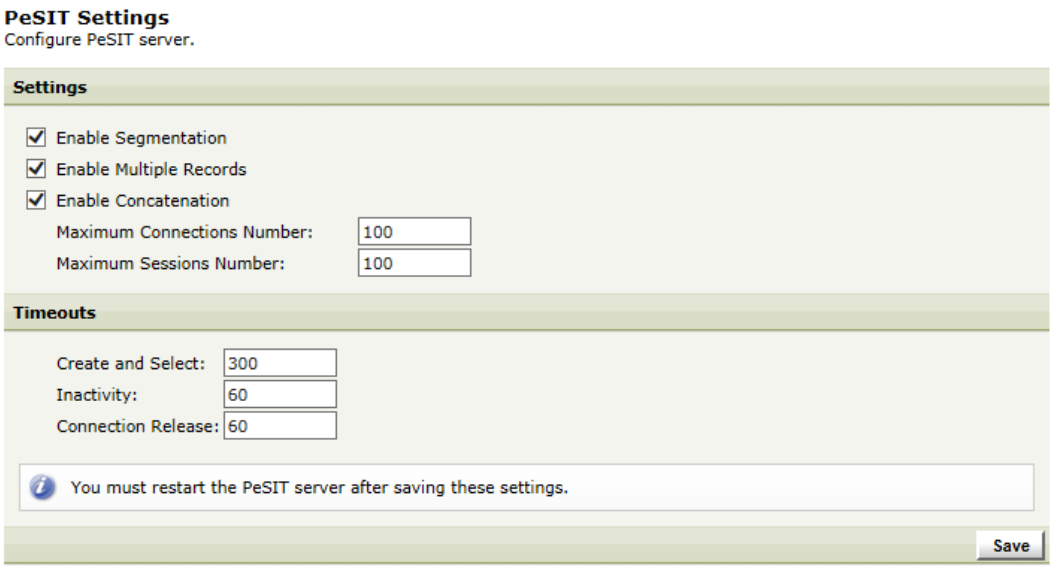
Here you can enable or disable the *Dashboard* page, control whether it should be accessible to delegated administrators, and customize its content. Note that the maximum number of certificates to display is `500`, and the maximum number of days is `180`.

PeSIT server configuration settings

Use the *PeSIT Settings* page to configure the PeSIT protocol server.

1. Select **Setup > PeSIT Settings**.

   The *PeSIT Settings* page is displayed.



2. Provide the information as described in the following table:

Name	Description
Settings	
Enable Segmentation	If a data article (record) is larger than the maximum size, it is sent in multiple FPDUs (messages).
Enable multiple records	If more than one data article fits in a FPDU, they are sent in one FPDU.
Enable Concatenation	Allow several FPDUs to be sent in one TCP message unit.
Maximum Connections Number	The maximum number of concurrent TCP connections remote PeSIT servers can make.
Maximum Sessions Number	The maximum number of concurrent sessions remote PeSIT servers can make.
Timeouts	
Create and Select	For connections initiated by a remote PeSIT server, the time in seconds that SecureTransport waits for a PeSIT F.CREATE or a F.SELECT command before SecureTransport closes the connection.

Name	Description
Inactivity	For connections initiated by a remote PeSIT server, the time in seconds that a connection may be inactive before SecureTransport closes it.
Connection Release	For connections initiated by a remote PeSIT server, the time in seconds that SecureTransport waits for a response to a PeSIT F.RELEASE command before SecureTransport closes it.

3. Click **Update**.
4. On the *Server Control* page, restart the PeSIT server.

AdHoc file transfers

In addition to scheduled and event-driven server-initiated file transfers, SecureTransport supports ad hoc file transfers. Using ST Web Client, a SecureTransport user with the required rights can compose an email, attach one or more files, and send it to any email address. Because the files are uploaded to the SecureTransport Server instead of sent in the email, SecureTransport ad hoc file transfers provide the following features not available with the standard email protocol:

- Large file size (for ST Web Client in a 32-bit browser, a maximum of 2 GB per message)
- Choice of methods of file delivery to the mail recipient, including limits on allowed recipients and secure delivery options
- Human-to-system (H2S) ad hoc file transfers where the SecureTransport server can process the file and forward it to another system

A message from ST Web Client can transfer at most 2 GB data.

The route the email takes to its recipient depends on the client the sender uses:

- **ST Web Client** – The SecureTransport Server sends the email using its SMTP configuration.

The recipient has different requirements to access the files, depending on the delivery mode.

Delivery modes are:

- **Anonymous** – The recipient clicks a link to access the files.
- **Challenge** – The recipient must answer a question correctly to access the files.
- **Account** – The recipient must log on to SecureTransport using ST Web Client with an existing user account.
- **Auto-enroll** – SecureTransport creates a user account for the recipient before the recipient can access the files. The recipient must log in to SecureTransport using a temporary password and set a new password before retrieving the files.

The following features of SecureTransport are useful for ad hoc file transfer recipients:

- **Login by email** – The user can use an email address as the user name in a web client log in page.
- **Unlicensed user accounts** – The user can only log in using ST Web Client to access the files and reply once to an ad hoc file transfer email.

For information about configuring ad hoc file transfers, see the following topics:

- *Configure AdHoc file transfers* on page 87
- *Create a user account* on page 503
- *Create a site template* on page 738
- Create or edit a business unit

For information about configuring H2S file transfers, see *Human to System type application* on page 673.

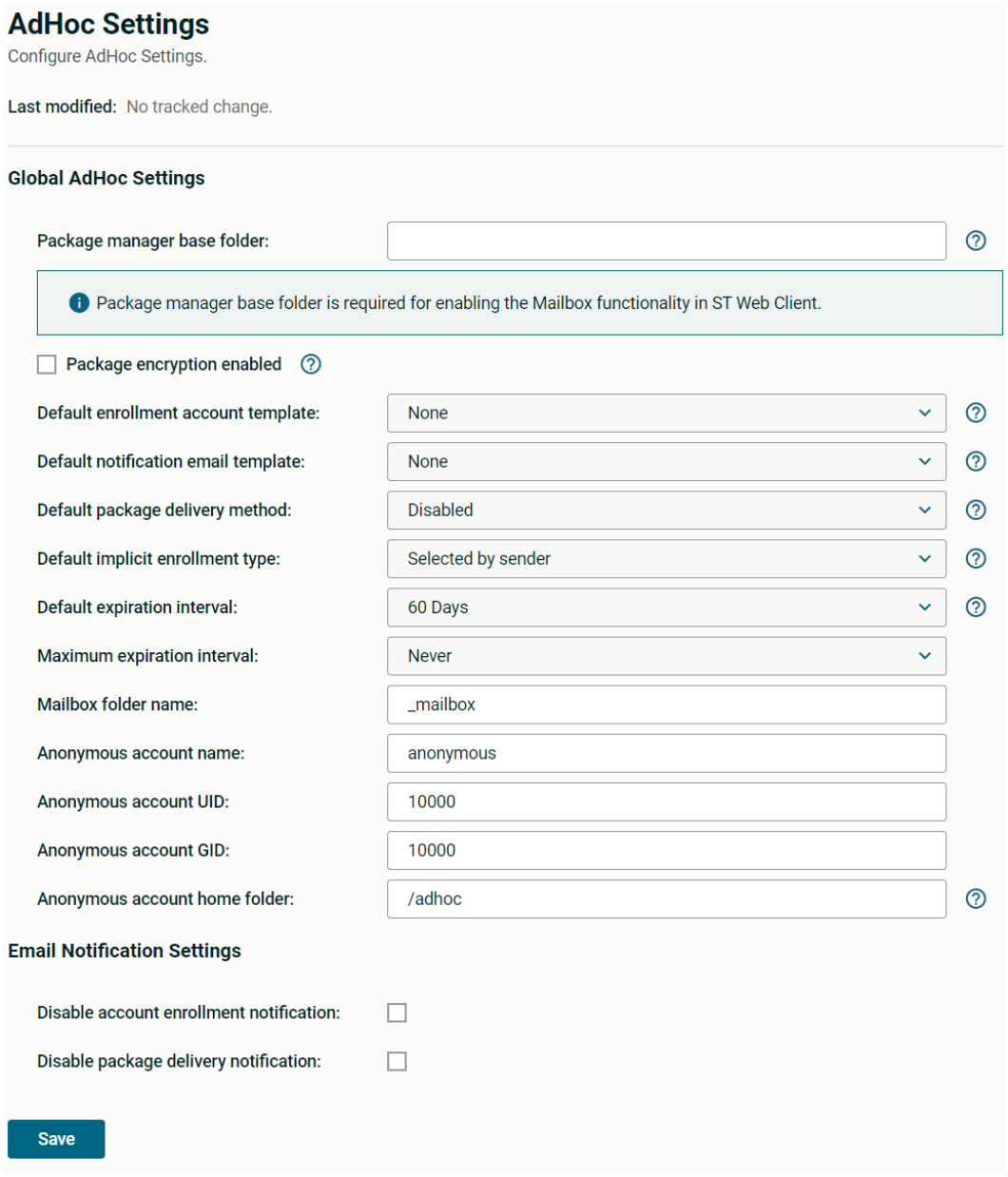
SecureTransport also supports system-to-human (S2H) file transfers. S2H file transfers are like other server-initiated file transfers, except the destination is a human, not a system. For example, the SecureTransport server can upload a large file from another server and send it to an email recipient using an S2H transfer site. The recipient of an S2H file transfer uses the same procedures to retrieve the file as the recipient of an ad hoc H2H file transfer. For more information, see *System to Human transfer sites* on page 624.

Configure AdHoc file transfers

In order to manage the AdHoc settings, an administrator must have access to the following pages: *Server Configuration*, *AdHoc Settings*, *Mail Templates*, and *Account Templates*. The access rights are defined through *Administrative roles* on page 711.

Use the following procedure to configure the settings for ad hoc file transfers using ST Web Client.

1. Navigate to **Setup > AdHoc Settings**.



2. Provide the necessary information as described in the following table:

Name	Description
Global AdHoc Settings	

Name	Description
Package manager base folder	SecureTransport uses this working folder to process ad hoc file transfers. This field is also required for using the Mailbox feature of ST Web Client. In a cluster, all servers use the same folder, so you must specify a location in the shared storage. Note that SecureTransport does not verify that the specified path exists. For a list of folders that are not allowed, see *Protected folders and accounts* on page 525.
Package manager system username (Windows only)	If the package manager base folder is in a shared storage, enter the username of the Windows system user that SecureTransport uses to access the package manager base folder. The user must be in the SecureTransport password vault. If you change this field, you must not remove the previous user from the password vault.
Package encryption enabled	If checked, SecureTransport enforces repository encryption on ad hoc packages. The global encryption settings must also be configured (`Stfs.Encryption.*` server configuration options).
Default enrollment account template	SecureTransport uses this account template when enrolling an ad hoc file transfer recipient. For more information, see *Account templates* on page 717.
Default notification email template	SecureTransport uses this mail template to create all notification emails to ad hoc file transfer recipients and senders. For more information, see *Mail templates* on page 195.

Name	Description
Default package delivery method	SecureTransport uses this delivery method when the **Delivery Method** field in *Account Setting* is set to **Default** for a user account, in an account template, or in a business unit, . • **Disabled** – Ad hoc file transfers are disabled. • **Anonymous** – The ad hoc file transfer recipient receives a link to retrieve the files and is not enrolled as a user. • **Account Without Enrollment** – Does not enroll ad hoc file recipients. Only existing users can receive files. • **Account With Enrollment** – The ad hoc file recipient must enroll as a SecureTransport user before retrieving the files. • **Custom** – Select the allowed enrollment types in the **Default enrollment type** field. Depending on the value of the **Default implicit enrollment type** field, the sender chooses one of the selected enrollment types when composing the mail in ST Web Client.
Default enrollment type	**Note** This setting becomes editable only when the **Default Package Delivery Method** is set to *Custom*. • **Anonymous Link** – The ad hoc file transfer recipient receives a link to retrieve the files, and is not enrolled as a user. • **Challenged Link** – The ad hoc file recipient receives a link and must answer a challenge question correctly to retrieve the files. The recipient is not enrolled as a user. • **Existing Account** – Does not enroll ad hoc file recipients. Only existing users can receive files. • **Enroll Unlicensed** – The ad hoc file recipient must enroll as a SecureTransport unlicensed user before retrieving the files. An unlicensed user can only reply once to the email and retrieve the files. Other user attributes are defined by the enrollment template. • **Enroll Licensed** – The ad hoc file recipient must enroll as a SecureTransport user with all the attributes specified in the default enrollment template before retrieving the files.
Default implicit enrollment type	The displayed values depend on the settings of the **Default Package Delivery Method** and **Default enrollment type**. ST Web Client sets this value as the initial value selected in the *User Access* window when a user with the **Delivery Method** field set to **Default** composes an email.

Name	Description
Default expiration interval	ST Web Client displays this value as the initial value of the **Expiration** dropdown list in the *Compose Mail* tab. Possible choices: `1 Day`, `7 Days`, `30 Days`, `60 Days`, and `Never`.
Maximum expiration interval	ST Web Client displays this value as the largest value of the **Expiration** dropdown list in the *Compose Mail* tab. Possible choices: `1 Day`, `7 Days`, `30 Days`, `60 Days`, and `Never`.
Mailbox folder name	SecureTransport creates a folder with this name in the user home folder to store the user's mail folders, which are used only for ad hoc file transfers. They are not visible to end users, but administrators can view and access them on the operating system. **Note** Adding or modifying files in the Mailbox folder or the user's mail folders breaks the Mailbox functionality in ST Web Client.
Anonymous account name Anonymous account UID Anonymous account GID Anonymous account home folder	SecureTransport uses this account name, UID, GID, and home folder when an ad hoc file transfer recipient logs in anonymously to retrieve a file. Creating this account is obligatory.
Email Notification Settings	
Disable account enrollment notification	SecureTransport does not send out account enrollment notifications when this option is selected. If selected, you must manage any account enrollment notifications externally using REST API from your email system. By default, this option is not selected.
Disable package delivery notification	SecureTransport does not send out package delivery notifications when this option is selected. If selected, you must manage any package delivery notifications externally using REST API from your email system. By default, this option is not selected. **Note** Package delivery notifications can also be disabled by the end user. If you do not select this option, the end user can disable package delivery notifications. If you select this option, the end user cannot override your selection and package delivery notifications will be disabled even if they are enabled by the end user.

3. Click **Save**.

Two server configuration parameters control handling of human-to-system (H2S) transfers:

- By default, the value of the `PackageManager.DualDeliveryDisabled` server configuration option is **true**. SecureTransport stores the files attached to an H2S email in the target folder and processes them, but does not send the email to the system user. When **true**, SecureTransport also sends the email to the system account. If the email recipients also include H2H addresses, SecureTransport sends the email to the those recipients regardless of the value of `PackageManager.DualDeliveryDisabled`.
- By default, the value of the `PackageManager.BodyRoutingDisabled` server configuration option is **true** and SecureTransport ignores the body of an H2S email. If the value is set to **false**, SecureTransport puts the body of the email in a text file the pattern *`uniqueID`*`_body_wap.txt` and stores it with the files attached to an H2S email in the target folder.

The following topics provide how-to instructions for changing the package manager base folder and describe the Package Retention Maintenance application.

- *Change the package manager base folder* on page 92
- *Package Retention Maintenance application* on page 846

Change the package manager base folder

To change the package manager base folder on a root installation:

1. Log on to the SecureTransport Server as root.
2. Stop all SecureTransport protocol servers and services.
3. If the new folder is on a network share, so that LDAP users can access the package manager base folder, mount the new network share on which the new package manager base folder is located using the default UID and GID specified in LDAP domain page for those users.
4. Copy the package manager base folder to its new location.
5. Delete the old package manager base folder.
6. So that references to the old package manager base folder still work, create a symbolic link that points to the new package manager base folder at the location of the old package manager base folder with the same names as the old package manager base folder.
7. Start the database and the Administration Tool service.
8. In the Administration Tool, edit the **Package Manager Base Folder** field to reference the new package manager base folder.
9. Start the remaining SecureTransport services and protocol servers.

Note You can make a package manager base folder stored on a network share accessible for users of only one LDAP domain.

Configure your database

SecureTransport uses a database to store configuration parameters and data, including log data. With Standard Clustering, SecureTransport Server uses an embedded database server. With Enterprise Clustering (EC), SecureTransport Server uses a shared external database. With the external Oracle database, you can direct log data to separate databases. SecureTransport Edge always uses an embedded database server.

Note MariaDB, MySQL, and Microsoft SQL Server database are configured to use a case-insensitive collation. For example, searching in the File Tracking and Server Log page or executing requests in the REST API will ignore the case sensitivity of the parameters and will return all matching results. Oracle and PostgreSQL, on the other hand, use a case-sensitive collation.

During installation, you specified the database to use and its parameters. Use the *Database Settings* page to perform the following tasks:

- Change database parameters:
  - *Change the embedded database configuration* on page 93
  - *Change the external Microsoft SQL Server database* on page 103
  - *Change Oracle database configuration* on page 99
  - *Change external PostgreSQL configuration and manage partitioning* on page 104
- *Migrate from embedded database to external Oracle database* on page 96
- *Direct log data to separate Oracle databases* on page 97
- *Connect to an Oracle database using Kerberos authentication* on page 100
- *Improve server resiliency in case of Oracle RAC node failure* on page 102
- *Set up PostgreSQL migration* on page 106

Change the embedded database configuration

SecureTransport administrators with database reconfiguration permissions can change the embedded database settings and restart the database service using the Administration Tool or Admin REST API.

In the Administration tool, go to **Setup > Database Settings**. From this page you can:

- change the database settings
- secure the connection to the database
- check and extend the validity of a certificate for connection to the embedded MariaDB database

Make sure you change the settings separately, as described in the following subtopics.

Note The MySQL/MariaDB configuration is stored in the `<FILEDRIVEHOME>/conf/internaldb.conf` file.

Make sure you change the settings separately, as described in the following subtopics.

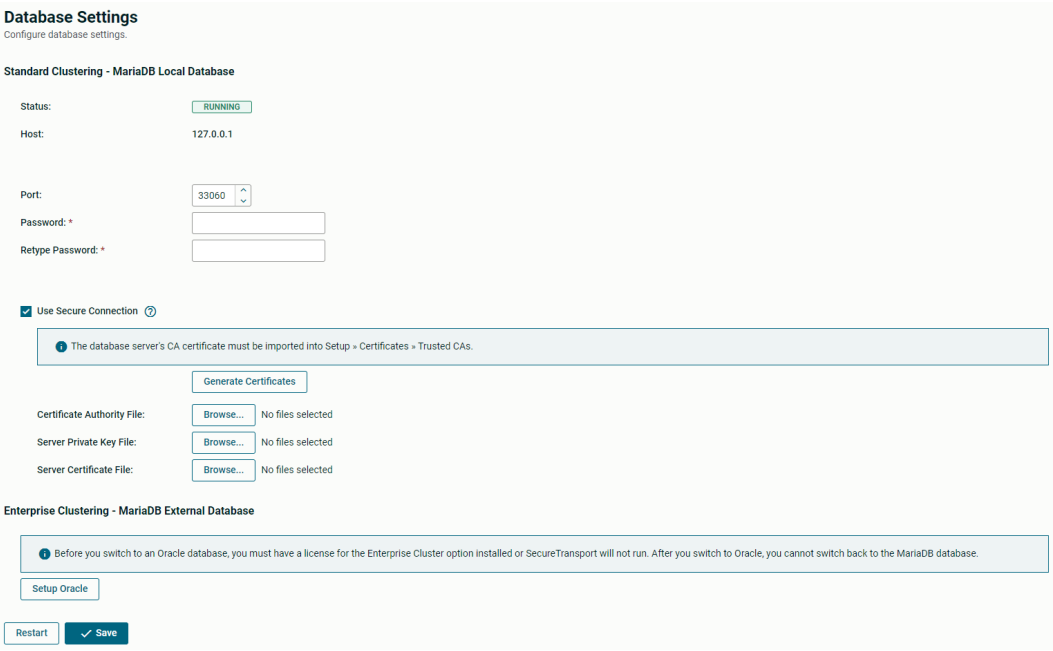
Change the embedded database port

1. Under *Standard Clustering - MariaDB / MySQL Local Database*, type the new port number in the **Port** field.
2. Click **Save**.
3. Click **Restart Database Now**.
4. After the database restart, restart all SecureTransport services.

Change the embedded database password

After installation, the embedded database password is `tumbleweed`. To improve security, change it following the procedure.

1. Under *Standard Clustering - MariaDB / MySQL Local Database*, type the new password in both the **Password** and **Retype Password** fields.
2. Click **Save**.
3. Restart all SecureTransport services.



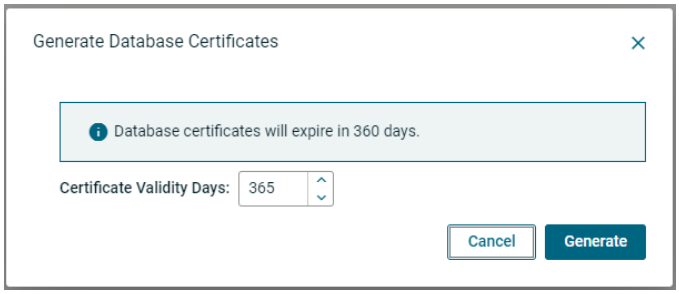
Secure the connection to the embedded MariaDB database

Proceed as follows to encrypt the connection between SecureTransport and the database server. Note that

1. On the *Database settings* page, check the **Use Secure Connection** checkbox.
2. SecureTransport requests the following certificates: **Certificate Authority File**, **Server Private Key File**, and **Server Certificate File**. You can either import the certificates manually or have SecureTransport provide them.
    - If you have used an external mechanism to generate the certificates, import the required files into SecureTransport using the dedicated **Browse** buttons. All files must be in PEM format.
    - If you want SecureTransport to provide the certificates, leave the three certificate fields empty and click **Save**. SecureTransport checks the available certificates in the `<FDH>/lib/certs/db` directory. If there are suitable certificates and key files, and all of them are valid, SecureTransport will use them and won't generate new ones. Otherwise, SecureTransport will create the required certificates with a default validity period of 365 days and store them in the `<FDH>/lib/certs/db` directory.
    - If you want SecureTransport to generate new certificates and use them to connect to the database, save the database configuration. The **Generate Certificates** button gets enabled, click on it. Use the spin buttons to set the certificate's validity period. When ready, click **Generate**.
3. Click **Save**.
4. Restart all SecureTransport services through their scripts for the changes to take effect.

Check certificate validity

To check the validity of the certificates present in the `<FDH>/lib/certs/db` directory, click the **Generate Database Certificates** button. SecureTransport shows the number of days left until the certificates expire. If your certificates have different expiry dates, then it shows the minimum number of days until one of the certificates expires.



By default, self-signed certificates generated by SecureTransport are valid for 365 days. The application checks the certificates for expiry at midnight, and sends a warning message (in the Server Log under WARN logging level) 7 days before the certificate expiration date. If you do not generate new certificates on time, SecureTransport will automatically regenerate the expiring certificates on the day of expiration at 12 am.

All services need to be manually restarted before the new certificates are applied. There will be log a message under ERROR logging level for the required service restart.

Migrate from embedded database to external Oracle database

If you upgraded a SecureTransport Server that used the embedded database or selected the embedded database when you installed SecureTransport Server, you can switch to an external Oracle database. In order to switch to an external Oracle database, you must have a license for the EC option installed or SecureTransport will not run. After you switch to the external database, you cannot switch back to the embedded database.

When you switch from the embedded database to an external Oracle database, you must set up the database. In this process, you specify the parameters of the Oracle database and SecureTransport migrate the configuration and data from the existing embedded database to the Oracle database. When the migration completes, you can use the SecureTransport Server as a stand-alone server or as the first server in an Enterprise Cluster (EC).

1. Make sure the Oracle database meet the requirements.
2. Select **Operations > Server Control** and stop all servers.
3. Select **Setup > Database Settings > Setup Oracle**.
4. On the *Target Database Settings* page of the Oracle wizard, type the values necessary to connect to the external database.
    a. Enter the connection parameters. For a list of database connection parameters, see *Change Oracle database configuration on page 99*.
    b. To configure SecureTransport to connect to the Oracle database using Kerberos authentication, follow the steps described in *Connect to an Oracle database using Kerberos authentication on page 100*. When Kerberos authentication is enabled, the **Username** and **Passwords** fields are disabled, as SecureTransport uses the ticket-granting (TGT) ticket to connect to the database. This feature requires additional configuration actions to be done on both the SecureTransport and the database server.
    c. Select the **Use Proxy Authentication** checkbox to connect to the Oracle database through another user. In the **Proxied Username** field, specify the user whose identity and privileges will be assumed by the connecting (proxy) user.
    d. Specify whether SecureTransport should use secure connection to the database server. Before you enable the secure connection, the issuer's certificates of the database server certificate, should be imported in the Trusted CA certificates store.
    When **Use secure connection** checkbox is selected, you need to configure the following:
        - **Server Certificate DN** (optional) – If the server is successfully authenticated (meaning its certificate is trusted), its DN can be checked. If a value is entered in this field, it will be compared with the server certificate DN. If they do not match, the connection won't be successful.
        - **Enabled Protocols** - List of enabled protocols. TLSv1 is the default protocol.
        - **Enabled Cipher Suites** - List of the enabled cipher suites.

5. To direct log data to separate external databases, see *Direct log data to separate Oracle databases* on page 97.
6. Click **Test Connection**.

   If SecureTransport displays a failure message, correct the network, Oracle, or other error reported and try again.
7. Type the password again, and click **Next**.
8. On the *Data Migration* page, select the **Migration Type**:
   - If you are upgrading the first server in a cluster, select **Migrate All Existing MariaDB/MySQL Data**. The process creates a new database schema that contains all the data from the existing database and configuration.
   - If you are upgrading the second and subsequent servers in a cluster, select **Migrate Local Setting Only**. The installer adds the local data for this server from the embedded database to the existing database schema it created for the first server.

     Note Set the `Cluster.mode` server configuration parameter to **disabled** prior migrating from a Standard Cluster to an Enterprise Cluster. For more information on changing server configuration parameters, refer to *View and change server configuration parameters on page 334*.
9. Leave **Roll-back to MariaDB/MySQL Database on Error** selected.
10. Click **Next**.
11. On the *Summary* page, review your settings. Click **Back** to return to a previous page and change a setting. Click **Setup Now** to migrate the data from the embedded database to the Oracle database or create the Oracle database.

    SecureTransport reports it progress as it transfers the server configuration and data from the embedded database to the external Oracle database.

    When the data migration is complete, the embedded database is no longer available and the *Database Settings* page includes the external database settings under *Enterprise Clustering - Oracle External Database*.

    The log output for the migration is in `<FILEDRIVEHOME>/var/logs/migration.log`.
12. Once the data migration is completed, shutdown and restart the SecureTransport server instance.

Note You can also migrate configuration data using the `data_migrate` command-line utility in the `<FILEDRIVEHOME>/bin` directory. For usage information, run `data_migrate -h`. Before you migrate the data, stop all servers except the database.

Direct log data to separate Oracle databases

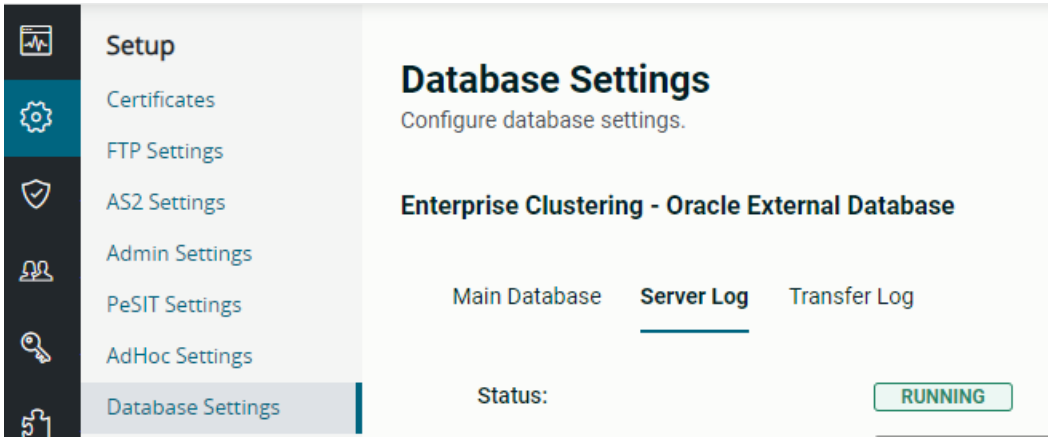
If SecureTransport Server uses an external Oracle database, it can be configured to direct transfer log (file tracking) data and server log data to separate databases.

Specifics:

- Each log database is configured independently.
- If you change a log database, SecureTransport will not copy the existing data to the new location.
- If the Server Log database is unavailable, SecureTransport continues to operate but it directs the server log data to `<FILEDRIVEHOME>/var/logs/serverlog-fallback.log`. The *Server Log* page indicates that the log is unavailable and provides a link to download the file.

Procedure

1. Make sure the Oracle databases have the characteristics listed in the *Database installation prerequisites* topic of the *SecureTransport Installation Guide*. Depending on the log data you want to store in a separate database, you must define the required tablespace: `ST_FILETRACKING` for the Transfer Log, `ST_SERVERLOG` for the Server Log.
2. Select **Setup > Database Settings**.
3. Configure the main database. For description of each property, see *Change Oracle database configuration on page 99*.
4. Click the corresponding tab for the log you want to direct to a separate database: **Server Log** or **Transfer Log**.



5. Configure the log database settings. Transfer Log and Server Log databases are configured independently from each other, and from the main application database.

   Note If using Kerberos authentication, note that the path to the Kerberos configuration file is specified in the main database settings and propagated to log databases. For detailed instructions, refer to *Connect to an Oracle database using Kerberos authentication on page 100*.

6. To validate that everything works properly, click **Test Connection**.

   If SecureTransport displays a failure message, correct the network, Oracle, or other error reported and try again.

7. Type the passwords again and click **Save**.

The first time the new database is referenced, SecureTransport creates the tables for the log data. SecureTransport does not copy the log data from the previous database to the new database and does not display the log entries from the previous database in the *File Tracking* and *Server Log* pages.

8. Select **Operations > Server Control**.
9. On the *Server Control* page, stop and restart all the protocol servers and the TM Server.
10. Log out of the Administration Tool.
11. Log on to the SecureTransport server and stop and restart the Administration Tool service using the `stop_admin` and `start_admin` commands in `<FILEDRIVEHOME>/bin`.
12. In an Enterprise Cluster (EC), repeat steps 2 though 10 for all SecureTransport Servers.

Change Oracle database configuration

If your SecureTransport installation is using an Oracle database, select **Setup > Database Settings** to perform the following tasks:

- Update the database configuration
- Configure separate databases for log data
- Migrate your Oracle database to PostgreSQL

Note You can also update the database connection parameters and test the connection via the Admin REST API 2.0.

Update Oracle database configuration

1. Log on to the SecureTransport Administration Tool as user `dbsetup`.
2. Select **Setup > Database Settings**.

   The *Database Settings* page is displayed. It shows the configuration settings for the main Oracle database.
   To configure separate databases for log data, go to the **Server Log** and **Transfer Log** tab. For more information, see *Direct log data to separate Oracle databases on page 97*.
3. Make the necessary updates in the main database configuration.
   - Connection settings:
     - **Host** – The host name or IP address of the Oracle server
     - **Port** – The port used to access the server, 1521 is the default
     - **User Name** – The name of the user authorized to create the SecureTransport schema.
     - **Password** – The password for the user, not displayed
     - **Service Name** – Used to connect to the Oracle server or cluster
   - **Use Kerberos Authentication**

Select this checkbox if you want SecureTransport to connect to the Oracle database with Kerberos authentication. For detailed instructions, refer to *Connect to an Oracle database using Kerberos authentication on page 100*.

When using separate databases for log data, the Kerberos Configuration file selected in the main database configuration also applies to the log databases.

- **Use Proxy Authentication**

  Select this checkbox to connect to the database through another user. In the **Proxied Username** field, specify the user whose identity and privileges will be assumed by the connecting (proxy) user.

- **Use Secure connection**

  To encrypt the connection between SecureTransport and the database server, do the following: .

  a. Configure Network Encryption for the Oracle database server: in the `$ORACLE_HOME/network/admin/sqlnet.ora` file, add `SQLNET.ENCRYPTION_SERVER=requested`.

  b. Import the CA certificate or the Issuing CA of the database server certificate into the SecureTransport's Trusted CA certificates store (Setup > Certificates > Trusted CAs).

  c. On the *Database Settings page*, select **Use Secure connection** and specify the following:

  - **Server Certificate DN** (optional) – If the server is successfully authenticated (meaning its certificate is trusted), its DN can be checked. If a value is entered in this field, it will be compared against the server certificate DN. If they do not match, the connection will fail.
  - In the **Enabled Protocols** field, specify the allowed SSL/TLS versions.
  - In the **Enabled Cipher Suites** field, enter the allowed cipher suites.
  - Click **Browse** and select the certificate file. Supported formats: PEM, DER, JKS.

4. To validate that everything works properly, click **Test Connection**.

   If SecureTransport displays a failure message, correct the network, database, or other error reported and try again.

5. Click **Save**.

Connect to an Oracle database using Kerberos authentication

This topic describes how to configure SecureTransport to connect to an Oracle database using Kerberos authentication. It assumes that the Oracle database has already been kerberized. To use this functionality, you will need to be running SecureTransport 5.5 October 2020 Update or later. It also requires additional configuration actions to be done on both the SecureTransport and the database server.

Prerequisites

Before you configure the machine where SecureTransport runs (or will be installed) for Kerberos authentication to an Oracle database, make sure the following prerequisites are fulfilled:

- Your Oracle database has been configured for Kerberos Authentication. For step-by-step instructions, refer to the Oracle's Security Guide.

  Note The recommended Kerberos ticket encryption type is AES256, and the generated `keytab` file must support it. As of SecureTransport 5.5-20221124, the DES and RC4 encryption types are deprecated and disabled by default.

- On the database server, verify that the following parameters in the *sqlnet.ora* file are set:

```
SQLNET.KERBEROS5_KEYTAB = /etc/krb5.keytab
SQLNET.AUTHENTICATION_SERVICES= (BEQ, KERBEROS5)
SQLNET.KERBEROS5_CLOCKSKEW = 6000
SQLNET.KERBEROS5_CONF = /etc/krb5.conf
SQLNET.KERBEROS5_CONF_MIT = true
SQLNET.AUTHENTICATION_KERBEROS5_SERVICE = oracle
```

- You have a running Key Distribution Center (KDC), such as Windows Active Directory, and realm setup
- You have created an externally authenticated oracle user that corresponds to the Kerberos user to be used
- The virtual machine, on which you install SecureTransport, has a resolvable FQDN (Fully Qualified Domain Name). If it doesn't, edit the */etc/hostname* file to add the following line: `<hostname>.yourrealm.com`
- All participants in a Kerberos realm have accurate system clocks

SecureTransport configuration

The following are the high-level steps on configuring Kerberos authentication for a connection to an Oracle database. This section points out only the settings that you need to know or change for SecureTransport. The procedure is platform-dependant.

1. Install the Kerberos client packages on the SecureTransport machine.
2. Edit the */etc/krb5.conf* file to reflect your realm setup.
3. Obtain a forwardable TGT ticket for the Kerberos/Oracle user.

Note The Kerberos ticket cache file must have a static name and must be always readable by SecureTransport.

4. Configure SecureTransport to use Kerberos authentication when connecting to the database.
   a. Select the **Use Kerberos Authentication** checkbox.
   b. Specify the location of the Kerberos credential cache file.
   c. Specify the location of the Kerberos configuration file:

- When **Use Kerberos configuration file** is unchecked (default), SecureTransport uses the default `/etc/krb5.conf` file. In this case, SecureTransport keeps the file locally and synchronizes it between nodes. The application monitors the `krb5.conf` for changes but uses the locally stored file after a restart. All changes to the `krb5.conf` file must be done through the *Server Configuration Files* page. To access the page, go to **Operations > Server Configuration** and click the **Configuration Files** button.
- When **Use Kerberos configuration file** is checked, SecureTransport refers to the Kerberos configuration file directly using its file path. In this case, SecureTransport does not copy the file locally or synchronize it between nodes; instead, you specify a configuration file per node. Whenever there is a change in the Kerberos configuration file, SecureTransport automatically reloads the Kerberos configuration.

Caution In an Enterprise Cluster setup, execute all the steps on all of theSecureTransport servers.

5. For more information on configuring fresh installations, refer to the *SecureTransport Installation Guide*. To change the database configuration on existing systems, refer to *Change Oracle database configuration on page 99*.

Caution In an Enterprise Cluster setup, execute all the steps on all of the SecureTransport servers.

Troubleshooting

- When using separate databases for the Server Log and the Transfer log, you need a dedicated Kerberos user for each. The users must use their own Kerberos cache file.
- When a Kerberos credential expires, the TGT is not renewed by SecureTransport. Ensure that a valid Ticket is always present; otherwise, the connection to the database will be lost and an internal server error will be reported.
- The Oracle database server may incorrectly interpret requests from SecureTransport as a replay attack ("Request is a replay" error message). This issue is resolved in Oracle 19.8 Patch 31716873. On previous Oracle database versions, the available workaround is to disable the Replay Cache mechanism.

Improve server resiliency in case of Oracle RAC node failure

Note The following procedure might affect SecureTransport Server performance.

Adjust the SecureTransport connection pool settings per the following procedure.

1. Set the C3P0 connection pools configured in `<FILEDRIVEHOME>/conf/configuration.xml` (set for each component) as follows:

   ```
   hibernate.c3p0.testConnectionOnCheckout=true
   ```

   ```
   hibernate.c3p0.preferredTestQuery=select 1 from DUAL
   ```

2. The DBCP connection pool is used for Quartz scheduling component. Set the DBCP settings configured in `<FILEDRIVEHOME>/conf/scheduler.properties` as follows:

   ```
   org.quartz.dataSource.DS.validationQuery=select 1 from DUAL
   org.quartz.dataSource.DS.testOnBorrow=true
   ```

Note Please note that the name of dataSource can be different than DS, for example, if SecureTransport database was migrated from an embedded database to external Oracle database, the name of dataSource will be DS2-migrate, and configuration options will be:
`org.quartz.dataSource.DS2-migration.validationQuery=select 1 from DUAL`
`org.quartz.dataSource.DS2-migration.testOnBorrow=true`

Change the external Microsoft SQL Server database

If the SecureTransport Server uses an external Microsoft SQL Server database and any of the database settings change, you must change the corresponding settings in the Administration Tool.

1. Log on to the SecureTransport Administration Tool as user `dbsetup`.
2. Select **Setup > Database Settings**.

   The *Database Settings* page is displayed with the external database settings under *Enterprise Clustering - Microsoft SQL Server External Database*.
3. Type the values necessary to connect to the new external database:
   - **Host** – The FQDN or IP address of the Microsoft SQL Server system
   - **Port** – The number of the port used to access the server, 1433 is the default
   - **User Name** – The name of the user authorized to connect to the database
   - **Password** – The password for the user, not displayed
   - **Database Name** – Used to connect to the Microsoft SQL Server.
   - **Use secure connection** – When **Use secure connection** is checked, the connection between SecureTransport and the database server will be encrypted. If **Use secure connection** is checked, the following optionc should be configured:
     - **Server Certificate CN** (optional) – If specified, SecureTransport will match the specified value with the database server certificate CN. If it is not specified, SecureTransport will match the host name with the database server certificate CN. If neither of them is matched, the connection will fail.
     - **Certificate Path** – Browse and select the TrustStore file to import the trusted certificates.
   - **Use Custom JDBC URL** – Unchecked by default. When checked, you can specify a custom connection string for SecureTransport to use to connect to the database. In the URL, specify the host, port, database name, user name and password or use the available placeholders. You can include additional connection properties to define specific

behavior, for example, a connection via SSL. If the custom JDBC URL connects to your database using SSL, make sure the **Use secure connection** checkbox is selected. The exact syntax of a JDBC URL is specified by your DBMS.

```
#Example
To specify an encrypted connection to mirror SQL instances by using a username and password:
jdbc:sqlserver://${host}:${port};databaseName=${databaseName};user=${user};password=${password};encrypt=${encrypt};trustStore=${trustStorePath};trustStorePassword=${trustStorePassword};hostNameInCertificate=${hostNameInCertificate};failoverPartner=${failoverHost};
```

The option to specify a custom JDBC URL is also exposed as a REST API resource.

Note When specified, the custom JDBC configuration is kept on update revert.

4. To validate that everything is working properly, click **Test Connection**.

   If SecureTransport displays a failure message, correct the network, database, or other error reported and try again.

5. Click **Save**.

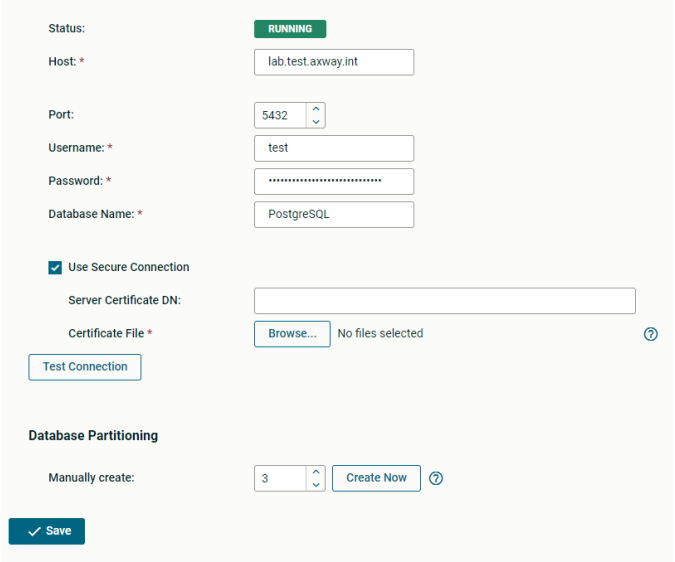
Note You can retrieve and update database configuration, as well as test the connection to the database via the Admin REST API .

Change external PostgreSQL configuration and manage partitioning

If your SecureTransport installation is using a PostgreSQL database, select **Setup > Database Settings** to perform the following tasks:

- Update the database configuration
- Create and manage table partitions

The PostgreSQL configuration is stored in the `<FILEDRIVEHOME>/conf/configuration.xml` file.



Update PostgreSQL database configuration

SecureTransport administrators with database reconfiguration permissions can change the database connection parameters and test the database connection using the Administration tool or the Admin REST API.

To update database configuration using the Administration tool:

1. Log on to the SecureTransport Administration Tool as an administrator with permissions to access *Database Settings* page.
2. Select **Setup > Database Settings**.

   The *Database Settings* page is displayed.
3. Make the necessary updates in the database configuration.
   - **Host** – The host name or IP address of the PostgreSQL server
   - **Port** – The port used to access the server, 5432 is the default.
   - **User Name** – The name of the user authorized to create the SecureTransport schema and populate it
   - **Password** – The password for the user, not displayed. The password cannot contain any of the following symbols: `%`, `&` or `+`.
   - **Database Name** – Used to connect to the PostgreSQL server or cluster
   - **Use secure connection** – When **Use secure connection** is checked, the connection between SecureTransport and the database server will be encrypted. If **Use secure connection** is checked, the following option should be configured:
     - **Certificate File** – Browse and select the public key certificate file. TrustStore files are not supported.

Note SecureTransport does not currently support JDBC connection strings for PostgreSQL.

4. To validate that everything is working properly, click **Test Connection**.

   If SecureTransport displays a failure message, correct the network, database, or other error reported and try again.

5. Click **Save**.

Manage database partitioning

On PostgreSQL, the Log Entry and Transfer Log maintenance applications do not create partitions. Instead, a dedicated partition creator service is executed on startup of the Transaction Manager and protocol daemons. The `Partiotion.DaysToPrebuild` server configuration option can be used to specify the number of days that partitions will be created in advance. If you leave it empty (default), partitions will be created for 3 days ahead. The service will not create new partitions if they have already been created for the specified interval.

By default, SecureTransport is also scheduled to daily partition log tables at midnight, 00:00, pre-creating partitions three days in advance. Partition names have the following pattern:

<table_name>_<start_date>v<end_date_epoch_to_millis>.

Change partition creation time

By default, the daily partitions are created every day at 00:00. To change the partition creation time, update the value of the `PartitionManagement.Create.triggerTime` server configuration option either using the Administration tool or the Admin REST API. The format is HH:MM, with hours in the range 0–23.

Create partitions manually

In situations where you have high volumes of data you can manually create partitions in advance to prevent table locking. SecureTransport allows you to pre-create partitions for minimum 3 days and up to 365 days ahead.

You can pre-create partitions in two ways:

- Using the Admin Rest API
- Using the Administration Tool:

  Select **Setup > Database Settings**. In the **Manually create** field, specify the number of days into the future for which to pre-create partitions. Then, click **Create now**.

Set up PostgreSQL migration

SecureTransport supports the following migration scenarios:

- from an on-premises Oracle instance to an on-premises PostgreSQL instance
- from an on-premises Microsoft SQL Server instance to an on-premises PostgreSQL instance
- from SQL Server on Azure VMs to Azure Database for PostgreSQL

After the migration completes, you cannot switch back to your previous database.

Caution SecureTransport instances running a 2021 Update and migrate to an on-premises PostgreSQL instance with non-superuser privileges, reverting to a 5.5 Update released in 2020 will cause issues as there are some incompatibilities. Refer to the Update Readme file for a workaround and revert instructions.

Prerequisites

Before you start the migration process, ensure that the following prerequisites are met:

- You have a license for Enterprise Cluster with PostgreSQL database
- The target database configuration meets the requirements listed in the Installation Guide and, in the case of Azure PostgreSQL, also in the Installation on Azure Guide.

Procedure

Currently, migration to an external PostgreSQL can be done only through the Administration Tool. Do not use `data_migrate` command-line utility.

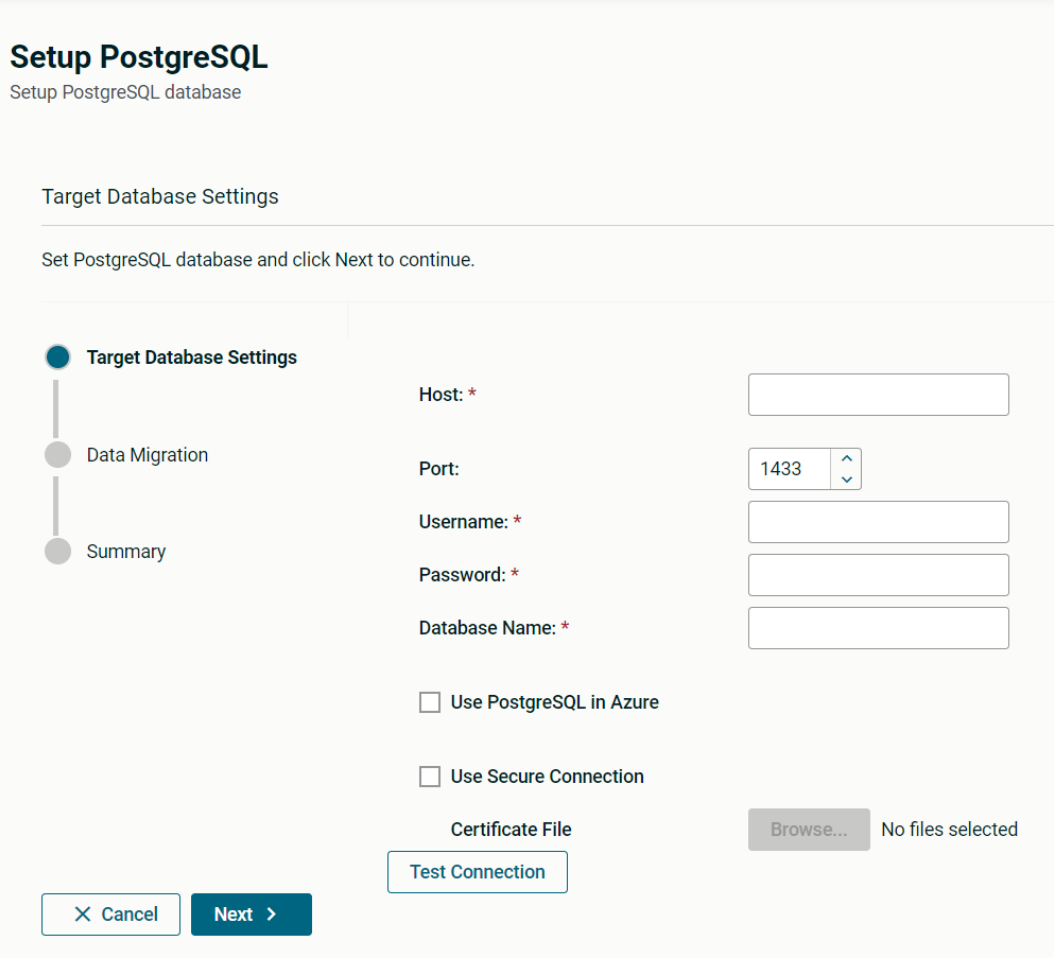
The process is straightforward - you specify the characteristics of the target PostgreSQL instance and SecureTransport migrates the configuration and data from the existing database to the target one. Regardless of the source database, the migration procedure is much the same. The most noticeable difference is the Azure migration option which is available only with SQL Server.

The procedure below covers all migration scenarios. Note that the name of the parameters and checkboxes may slightly vary for different source databases.

To migrate to PostgreSQL on-premises or Azure Database for PostgreSQL, perform the following steps:

1. Go to **Operations > Server Control**.
2. Stop all servers.
3. Go to **Setup > Database Settings**.
4. Click the **Setup PostgreSQL** button.

   The PostgreSQL setup wizard opens.

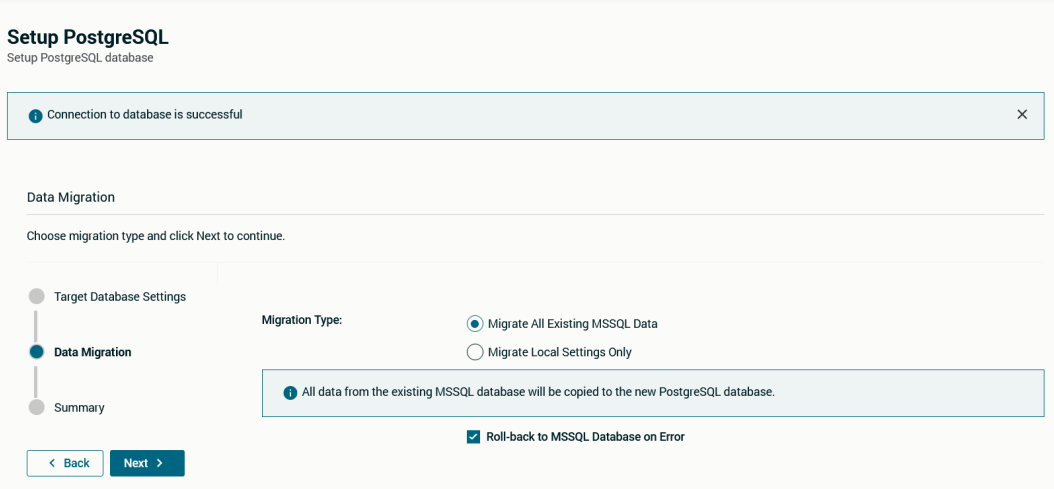


5. On the *Target database settings* screen of the wizard, provide the required information to connect and authenticate to the target PostgreSQL database.
    a. Fill in the required parameters according to the migration scenario:
        - If your target is on-premises PostgreSQL, leave the **Use PostgreSQL in Azure** checkbox cleared and fill in the required parameters. For detailed descriptions of all parameters, see *Change external PostgreSQL configuration and manage partitioning on page 104*.
        - If your target is Azure Database for PostgreSQL, select the **Use PostgreSQL in Azure** checkbox.
          In the **Host** field, enter the Server name of your Azure Database for PostgreSQL server.
          In the **Username** field, enter the server admin login name. Then, enter the password for that admin login name in the **Password** field.
    b. Use the **Use secure connection** checkbox to specify whether SecureTransport should connect to the database server using SSL. When selected, you need to import the public key certificate file.

        Note TrustStore files are not supported.
6. Click **Test Connection** to validate that everything is working properly.

If SecureTransport displays a failure message, correct the network, PostgreSQL, or other error reported and try again.

7. Type the password again, and click **Next**.
8. On the *Data Migration* page, select the **Migration Type**.



   - If you are upgrading the first server in a cluster, select **Migrate All Existing Data**. The process creates a new database schema that contains all the data from the existing database and configuration.
   - If you are upgrading the second and subsequent servers in a cluster, select **Migrate Local Setting Only**. The installer updates the local data for this server to point to the existing database schema it created for the first server.

9. Leave the **Roll-back on Error** checkbox selected.
10. Click **Next**.
11. On the *Summary* page, review your settings. To return to a previous page and change a setting, click **Back**. To run the migration, click **Setup Now**.

    SecureTransport reports its progress as it transfers the server configuration and data to the PostgreSQL database.

    When the data migration completes, the old database is no longer available. The *Database Settings* page shows the new database settings.

    The log output for the migration is in `<FILEDRIVEHOME>/var/logs/migration.log`.
12. Once the data migration is completed, switch to your PostgreSQL license.
13. Shut down and restart the SecureTransport Server instance.

Database connection pool configuration

This chapter provides details for setting database connection pool properties in SecureTransport. The application uses multiple database connection pools, pre-configured with a default setup that

provides for system stability. However, the database connection requirements are individual to deployments. In some cases, you may need to modify the default pooling behavior and tune a connection pool for a particular SecureTransport component in order to achieve optimal performance.

SecureTransport uses the c3p0 connection pooling, configured in the following way:

- the connection pools for all SecureTransport components are described in the *<FDH>/conf/configuration.xml* file. See *Connection pools for SecureTransport components* on page 110.
- the connection pools for the Server log are configured per daemon in dedicated log4j files under the `<FILEDRIVEHOME>/conf` directory. See *Connection pools for the Server log on page 118*.

When configuring the connection pool size, you need consider the number of started services and their corresponding max number of connections, configured in the *configuration.xml* file and the *log4j* files. See *Calculating min and max number of connections on page 119*. In the case of Enterprise cluster, this amount should be multiplied by the number of nodes to get the min and max number of the TCP connections to the database.

Connection pools for SecureTransport components

The c3p0 connection pooling configuration is described in the *<FDH>/conf/configuration.xml* file. Inside the file are a series of dedicated configuration sections that define the database connection and pool settings for each SecureTransport component. The following table lists all SecureTransport components, their usage and availability in SecureTransport Server and Edge.

SecureTransport Components

Component	Description
ToolsComponent	Used by various CLI utilities to prepare configuration on startup, database migration, and script execution. Example utilities: *rmadmin, collect_support_configuration, repconv, log_export, options_overwrite, bounce_tool, update_administrator, system_import, system_export, mkadmin, import_certificate_ authority, xml_import*, and etc.
TransferLogComponent *(only on SecureTransport Server)*	Used for configuring the TransferLog connection pool (File Tracking functionality)

Component	Description
ServerLogComponent	Used for configuring the ServerLog connection pool (Server Log functionality). The configuration defined in the *configuration.xml* is used for operations like export and reading of logs, log entry maintenance. SecureTransport maintains separate connection pools for Log4j to log events to the database. The configuration for those pools is specified in the per-daemon Log4j files. See *Connection pools for the Server log on page 118*.
AdminComponent	Administration server. Extracts, persists and manages server configuration and user data from/to the database.
TransactionManagerComponent *(only on SecureTransport Server)*	Transaction Manager. Extracts server/user/flow configuration from the database. Creates, updates and deletes workload events.
AS2Component FTPDComponent HTTPDComponent PesitComponent SSHDComponent	Every SecureTransport Server (AS2 Server, FTP Server, HTTP Server, PeSIT Server, SSH Server) uses a separate database connection pool to fetch its own configuration from the external database.

Component's connection pool options

Each component's configuration section contains options that define its database connection pool - those are the options with the *hibernate.c3p0* prefix. This extract of the *configuration.xml* file is the default connection pool configuration for the HTTPDComponent.

```
...
<Database_HTTPDComponent ......>
    <Options
hibernate.dialect="com.tumbleweed.st.server.appframework.util.STPostgreSQLDial
ect"
          hibernate.show_sql="false"
          hibernate.c3p0.min_size="2"
          hibernate.c3p0.max_size="8"
          hibernate.c3p0.timeout="1800"
          hibernate.c3p0.checkoutTimeout="300000"
          hibernate.cache.use_minimal_puts="false"
          hibernate.cache.use_second_level_cache="false"
          hibernate.cache.use_query_cache="false"
          hibernate.c3p0.preferredTestQuery="select 1"
          hibernate.c3p0.max_statements="50"
          hibernate.c3p0.acquireRetryAttempts="0"
          hibernate.c3p0.acquireRetryDelay="5000"
          hibernate.c3p0.breakAfterAcquireFailure="false"
          hibernate.c3p0.maxIdleTime="30"
```

```
    />
...
```

The table below lists the pre-configured database connection and pool options available in the *configuration.xml* file for all components. The connection pool size is defined per component via `hibernate.c3p0.min_size` and `hibernate.c3p0.max_size`.

Option	Description
hibernate.c3p0.min_size *(integer)*	Minimum number of connections a pool should maintain at any given time.
hibernate.c3p0.max_size *(integer)*	Maximum number of connections a pool can maintain at any given time.
hibernate.c3p0.timeout *(integer)*	Seconds a connection can remain pooled but unused before being discarded. Zero means idle connections never expire.
hibernate.c3p0.checkoutTimeout *(integer)*	The number of milliseconds a client calling getConnection() will wait for a connection to be checked in or acquired when the pool is exhausted. Zero means "wait indefinitely". Setting any positive value will cause the getConnection() call to time out and break with an SQLException after the specified number of milliseconds.
hibernate.cache.use_minimal_puts *(boolean)*	Optimizes second-level cache operations to minimize writes, at the cost of more frequent reads. Providers typically set this appropriately.
hibernate.cache.use_query_cache *(boolean)*	Enable or disable second-level caching of query results.
hibernate.cache.use_second_level_cache *(boolean)*	Enable or disable second level caching overall.
hibernate.show_sql *(boolean)*	Write all SQL statements to the console.
hibernate.c3p0.preferredTestQuery *(string)*	Defines the query that will be executed for all connection tests. Defining a preferredTestQuery that will execute quickly in your database may dramatically speed up connection tests.

Option	Description
hibernate.c3p0.testConnectionOnCheckout *(boolean)*	If true, an operation will be performed at every connection checkout to verify that the connection is valid.
hibernate.connection.oracle.jdbc.ReadTimeout *(integer)*	Read timeout while reading from the socket. Timeout is in milliseconds. **Note** Available on Oracle database
hibernate.connection.lockTimeout *(integer)*	The number of milliseconds to wait before the database reports a lock time-out. The default behavior is to wait indefinitely. If it's specified, this value is the default for all statements on the connection. **Note** Available on MSSQL database.
hibernate.connection.loginTimeout *(integer)*	The number of seconds the driver should wait before timing out a failed connection. A zero value indicates that the timeout is the default system timeout, which is specified as 15 seconds by default. A non-zero value is the number of seconds the driver should wait before timing out a failed connection. **Note** Available on MSSQL database.
hibernate.c3p0.acquireRetryAttempts *(integer)*	Defines how many times c3p0 will try to acquire a new connection from the database before giving up. If this value is less than or equal to zero, c3p0 will keep trying to fetch a connection indefinitely.
hibernate.c3p0.acquireRetryDelay *(integer)*	The time c3p0 will wait between acquire attempts (in milliseconds).

Option	Description
hibernate.c3p0.breakAfterAcquireFailure *(boolean)*	If true, a pooled DataSource will declare itself broken and be permanently closed if a connection cannot be obtained from the database after making *acquireRetryAttempts* to acquire one. If false, failure to obtain a connection will cause all threads waiting for the pool to acquire a connection to throw an Exception, but the DataSource will remain valid, and will attempt to acquire again following a call to getConnection().
hibernate.c3p0.maxIdleTime *(integer)*	The number of seconds a connection can remain pooled but unused before being discarded. Zero means idle connections never expire.

You can edit the configuration file and change the default values to optimize the pooling behavior. You can also specify additional options in the form of key-value pairs. Those options must be placed inside the `Options` section of the component. For information on the available configuration options, refer to the c3p0 and Hibernate official documentation:

- https://www.mchange.com/projects/c3p0/
- https://docs.jboss.org/hibernate/orm/5.0/userguide/en-US/html/ch11.html
- https://docs.jboss.org/hibernate/orm/4.2/manual/en-US/html/ch03.html

Default pool configuration with MariaDB

Parameter | Component	Tools	Admin	TM	FTPD, HTTPD, Pesit, AS2, SSHD, TransferLog, ServerLog
hibernate.c3p0.min_size	1	2	5	2
hibernate.c3p0.max_size	5	32	32	8
hibernate.cache.use_query_cache	false	false	true	false
hibernate.cache.use_second_level_cache	false	false	N/A	false
hibernate.c3p0.testConnectionOnCheckout	N/A	true	N/A	N/A

hibernate.jdbc.batch_size	N/A	N/A	500	N/A
hibernate.c3p0.acquireRetryAttempts	0	0	0	0
hibernate.c3p0.acquireRetryDelay	5000	5000	5000	5000
hibernate.c3p0.breakAfterAcquireFailure	false	false	false	false
hibernate.c3p0.maxIdleTime	30	30	30	30

Default value for all components:
hibernate.cache.use_minimal_puts = false
hibernate.dialect = org.hibernate.dialect.MySQLInnoDBDialect
hibernate.show_sql = false
hibernate.c3p0.preferredTestQuery = select 1
hibernate.c3p0.timeout = 1800
hibernate.c3p0.acquireRetryAttempts = 0
hibernate.c3p0.acquireRetryDelay = 5000
hibernate.c3p0.breakAfterAcquireFailure = false
hibernate.c3p0.maxIdleTime = 30

Default pool configuration with PostgreSQL

Parameter|Component	**Tools**	**Admin**	**TM**	**FTPD, HTTPD, Pesit, AS2, SSHD, TransferLog, ServerLog**
hibernate.c3p0.min_size	1	2	5	2
hibernate.c3p0.max_size	5	32	32	8
hibernate.cache.use_query_cache	false	true	true	false
hibernate.cache.use_second_level_cache	false	N/A	N/A	false
hibernate.c3p0.testConnectionOnCheckout	N/A	true	N/A	N/A
hibernate.jdbc.batch_size	N/A	N/A	500	N/A
hibernate.c3p0.acquireRetryAttempts	0	0	0	0

hibernate.c3p0.acquireRetryDelay	5000	5000	5000	5000
hibernate.c3p0.breakAfterAcquireFailure	false	false	false	false
hibernate.c3p0.maxIdleTime	30	30	30	30

Default value for all components

hibernate.cache.use_minimal_puts = false
hibernate.dialect = com.tumbleweed.st.server.appframework.util.STPostgreSQLDialect
hibernate.show_sql = false
hibernate.c3p0.preferredTestQuery = SELECT 1
hibernate.c3p0.timeout = 1800
hibernate.c3p0.checkoutTimeout = 300000
hibernate.c3p0.acquireRetryAttempts = 0
hibernate.c3p0.acquireRetryDelay = 5000
hibernate.c3p0.breakAfterAcquireFailure = false
hibernate.c3p0.maxIdleTime = 30

Default pool configuration with Oracle

Parameter|Component	**Tools**	**Admin**	**TM**	**FTPD, HTTPD, Pesit, AS2, SSHD, TransferLog, ServerLog**
hibernate.c3p0.min_size	1	2	5	2
hibernate.c3p0.max_size	5	32	32	8
hibernate.cache.use_query_cache	false	true	true	false
hibernate.cache.use_second_level_cache	false	N/A	N/A	false
hibernate.c3p0.testConnectionOnCheckout	N/A	true	N/A	N/A
hibernate.jdbc.batch_size	N/A	N/A	500	N/A
hibernate.c3p0.acquireRetryAttempts	0	0	0	0
hibernate.c3p0.acquireRetryDelay	5000	5000	5000	5000
hibernate.c3p0.breakAfterAcquireFailure	false	false	false	false

hibernate.c3p0.maxIdleTime	30	30	30	30

Default value for all components

hibernate.dialect = org.hibernate.dialect.Oracle10gDialect
hibernate.c3p0.preferredTestQuery = SELECT * FROM ALL_TABLES WHERE 1=0
hibernate.c3p0.timeout = 1800
hibernate.c3p0.checkoutTimeout = 300000
hibernate.cache.use_minimal_puts = false
hibernate.show_sql = false
hibernate.connection.oracle.jdbc.ReadTimeout = 300000
hibernate.c3p0.acquireRetryAttempts = 0
hibernate.c3p0.acquireRetryDelay = 5000
hibernate.c3p0.breakAfterAcquireFailure = false
hibernate.c3p0.maxIdleTime = 30

Default pool configuration with MSSQL

Parameter|Component	**Tools**	**Admin**	**TM**	**FTPD, HTTPD, Pesit, AS2, SSHD, TransferLog, ServerLog**
hibernate.c3p0.min_size	1	2	5	2
hibernate.c3p0.max_size	5	32	32	8
hibernate.cache.use_query_cache	false	true	true	false
hibernate.cache.use_second_level_cache	false	N/A	N/A	false
hibernate.c3p0.testConnectionOnCheckout	N/A	true	N/A	N/A
hibernate.jdbc.batch_size	N/A	N/A	500	N/A
hibernate.c3p0.acquireRetryAttempts	0	0	0	0
hibernate.c3p0.acquireRetryDelay	5000	5000	5000	5000
hibernate.c3p0.breakAfterAcquireFailure	false	false	false	false
hibernate.c3p0.maxIdleTime	30	30	30	30

Default value for all components

hibernate.dialect = com.tumbleweed.st.server.appframework.util.UnicodeSQLServerDialect
hibernate.c3p0.preferredTestQuery = SELECT 1
hibernate.c3p0.timeout = 1800
hibernate.c3p0.checkoutTimeout = 300000
hibernate.cache.use_minimal_puts = false
hibernate.show_sql = false
hibernate.connection.lockTimeout = 300000
hibernate.connection.loginTimeout = 15
hibernate.c3p0.acquireRetryAttempts = 0
hibernate.c3p0.acquireRetryDelay = 5000
hibernate.c3p0.breakAfterAcquireFailure = false
hibernate.c3p0.maxIdleTime = 30

Caution In addition, SecureTransport maintains separate connection pools for Log4j to write events to the database. The configuration for those pools is specified in per-daemon configuration files under the `<FILEDRIVEHOME>/conf` directory. For more information, see *Connection pools for the Server log* on page 118.

Connection pools for the Server log

SecureTransport maintains separate connection pools intended for Log4j STDBAppender that writes log events to the database. The configuration for those pools is specified in per-daemon configuration files under the `<FILEDRIVEHOME>/conf` directory. For more information, see *Log4j files* on page 1074.

A typical STDBAppender configuration might look like this:

```
<STDBAppender name="ServerLog" locationInfo="true"
maxLoggingEventQueueSize="10000"
      fallbackLogger="ServerLogFallback" databaseStatusCheckupDelay="60"
databaseCheckupTimeout="30"
      idAreaBegin="-20000000000" idAreaEnd="-10000000000"
      driverClass="org.mariadb.jdbc.Driver" initialPoolSize="5"
maxPoolSize="25" minPoolSize="5"
      acquireIncrement="5" maxStatements="4000">
   <STLog4JNDCFilter componentName="ADMIN" onMatch="ACCEPT"
onMismatch="DENY"/>
</STDBAppender>
```

Based on the STDBAppender configuration, SecureTransport creates a c3p0 pooling data source. The following table lists the appender properties for configuring connection pooling.

Property	Description (c3p0 official documentation)
initialPoolSize	Number of connections a pool will try to acquire upon startup. Should be between minPoolSize and maxPoolSize.
minPoolSize	Minimum number of connections a pool should maintain at any given time.
maxPoolSize	Maximum number of connections a pool can maintain at any given time.
acquireIncrement	Determines how many connections a c3p0 pool will attempt to acquire when the pool has run out of connections. (Regardless of `acquireIncrement`, the pool will never allow `maxPoolSize` to be exceeded.)
maxStatements	The size of c3p0's global PreparedStatement cache. For details, see the official C3p0 documentation.

Note that those properties have different default values in the different daemon Log4j files. The following table shows the default values of the options affecting the connection pool size:

Property	Admin, FTPD, HTTPD, SSHD, Pesit, AS2	TM	Tools
initialPoolSize	5	30	30
maxPoolSize	25	100	100
minPoolSize	5	30	30

Caution SecureTransport maintains separate connection pools for SecureTransport Components. See *Connection pools for SecureTransport components* on page 110.

Calculating min and max number of connections

The following example illustrates how to calculate the minimum and the maximum number of connections needed for Oracle Standalone setup with the Admin, TM, HTTP, SSH, AS2, and PeSIT services started. For each running service, we sum the values of the connection pool options set in the *configuration.xml* and the *log4j* files. In the example, we use the default values and get a total minimum of 119/154* connections and a total maximum of 486/611* (* if *ImprovedRoutingAppender* is included), distributed as follows:

Admin Startup: [min 15/20* - max 71/96*]

- *admin-log4j.xml*
    - ServerLog [min 5 - max 25]
    - AuditLogAppender [min 5 - max 25]
    - ImprovedRoutingAppender [min 5 - max 25] - is taken only during the creation of routing objects in the account (example: creating a route with a route step in the account)
- *configuration.xml*
    - AdminComponent [min 2 - max 8]
    - ToolsComponent [min 1 - max 5]
    - ServerLogComponent [min 2 - max 8]

TransactionManager Startup: [min 39/69* - max 150/250*]

- *tm-log4j.xml*
    - ServerLog [min 30 - max 100]
    - ImprovedRoutingAppender [min 30 - max 100] - is taken only when the routing is initialized
- *configuration.xml*
    - TransactionManagerComponent [min 5 - max 32]
    - ToolsComponent [min 1 - max 5]
    - TransferLogComponent [min 2 - max 8]
- *scheduler.properties*
    - org.quartz.dataSource.DS.maxConnections (Default is 5) [min 1 - max 5]

AS2D Startup: [min 33 - max 113]

- *as2d-log4j.xml*
    - ServerLog [min 30 - max 100]
- *configuration.xml*
    - AS2Component [min 2 - max 8]
    - ToolsComponent [min 1 - max 5]

HTTPD Startup: [min 8 - max 38]

- *httpd-log4j.xml*
    - ServerLog [min 5 - max 25]
- *configuration.xml*
    - HTTPDComponent [min 2 - max 8]
    - ToolsComponent [min 1 - max 5]

FTPD Startup: [min 8 - max 38]

- *ftpd-log4j.xml*
  - ServerLog [min 5 - max 25]
- *configuration.xml*
  - FTPDComponent [min 2 - max 8]
  - ToolsComponent [min 1 - max 5]

SSHD Startup: [min 8 - max 38]

- *sshd-log4j.xml*
  - ServerLog [min 5 - max 25]
- *configuration.xml*
  - SSHDComponent [min 2 - max 8]
  - ToolsComponent [min 1 - max 5]

PESITD Startup: [min 8 - max 38]

- *pesitd-log4j.xml*
  - ServerLog [min 5 - max 25]
- *configuration.xml*
  - PesitComponent [min 2 - max 8]
  - ToolsComponent [min 1 - max 5]

Sentinel

Sentinel is a Business Activity Monitoring (BAM) product that collects, aggregates, correlates, and reports events from SecureTransport and other products, applications, and systems throughout your infrastructure. It is a separate product that you can buy from Axway or an authorized partner. Once you license and configure SecureTransport to send file transfer and processing events to Sentinel, data is collected and displayed on a Sentinel dashboard.

Glossary of terms

The following terms and concepts are introduced with Sentinel:

- **Event states**

  Event states indicate the current state of a file transfer. For more information, see *Event states* on page 157.

Example scenario: a user uploads a file to SecureTransport.
For SecureTransport (the receiver of the file), there are two events represented by two different event states:

- `RECEIVING` - when SecureTransport starts receiving the file.
- `RECEIVED` - when SecureTransport has received the whole file.

- **Tracked objects**

  A Sentinel tracked object is a structure that contains a set of attributes which describe event states. For more information, see *Tracked Objects* on page 127.

- **CycleID**

  A CycleID is a standard Sentinel attribute that identifies processing cycles. All events in one processing cycle share the same value for the CycleID attribute. For more information, see *CycleId* on page 162.

- **Processing cycle**

  Example scenario: Transfer CFT successfully sends a file to SecureTransport.
  For Transfer CFT (the sender), there are two events: `SENDING` and then `SENT`.
  For SecureTransport (the receiver), there are two corresponding events: `RECEIVING` and then `RECEIVED`. In total, there are 4 events which are all part of the same processing cycle (the file upload from Transfer CFT to SecureTransport), and therefore have the same CycleID. You can see them linked in Sentinel one after the other.

- **Cycle links**

  With the use of cycle links, you can connect events to represent one whole flow and see it on the Sentinel Dashboard.

  Example scenario: SecureTransport receives a file which triggers an Advanced Routing subscription with transformation and routing steps.
  This results in three sets of events with different CycleIDs: one CycleID for all events related to receiving the file, a second CycleID for all events related to the transformation steps, and a third set of events related to the routing steps, where each step has its own CycleID. You can connect all three sets of events by linking the processing cycles.

The following topics provide additional information on Sentinel:

- *Configure Sentinel reporting* on page 123
- *Event states* on page 157
- *Tracked Objects* on page 127
- *XFBTransfer Tracked Object* on page 128
- *ST_VAS Tracked Object* on page 153
- *Heartbeat Tracked Object* on page 156
- *CycleId* on page 162
- *PeSIT states and roles explained with examples* on page 171
- *Axway Sentinel requests* on page 166

Configure Sentinel reporting

This topic describes the steps you need to take to integrate SecureTransport with Axway Sentinel.

Overview

The information that SecureTransport collects and sends to Sentinel is defined by Tracked Objects. A Tracked Object (TO) is a data container in Sentinel that defines the parameters of a specific application event. The definitions of the Tracked Objects are predefined and delivered by Axway as XML files. They must be imported into Sentinel before you start configuring the reporting functionality in SecureTransport. In SecureTransport, you specify the connection parameters, the events to report, and the fallback settings when the Sentinel server is unreachable or disabled.

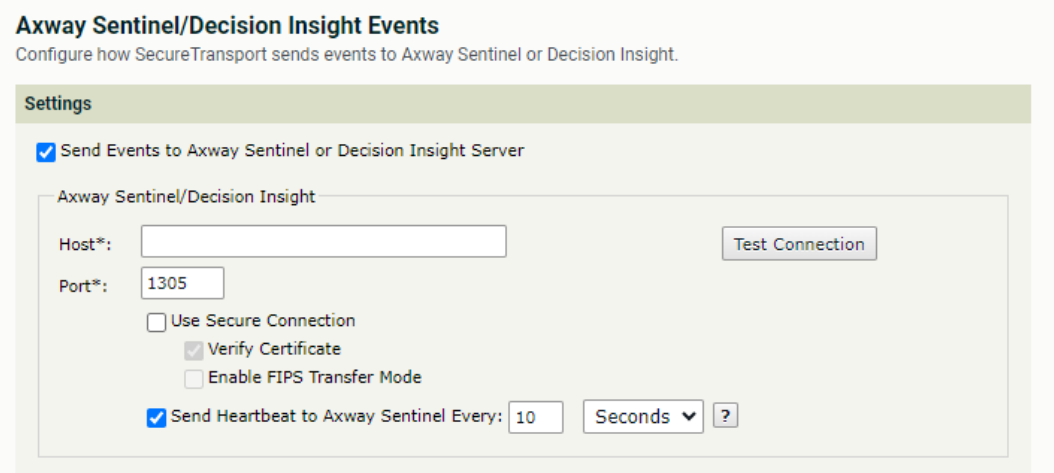
Deploy Tracked Objects in Sentinel

1. Download the XML definition files for the Tracked Objects you want to use. For more information about the TOs that SecureTransport supports, see *Tracked Objects* on page 127.
    - The XFBTransfer and ST_VAS definition files are provided in SecureTransport Package for Sentinel.
    - The Heartbeat definition file is provided with Sentinel. It is located in `<Sentinel_Installation_Directory>/Sentinel/www/extra/trackedobject`.
2. Upload the XML files of the desired Tracked Objects to the following folder on the Sentinel server: `<SentinelInstallationDirectory/broadcast/commit/TrackingObject` Alternatively, you can import Tracked Object definition files via the Sentinel Rest API documented here.

Configure SecureTransport to send data to Sentinel

The following procedure describes how to configure SecureTransport through the Administration Tool to send events to Sentinel. Besides the GUI, you can use the Admin REST API resource *PUT /configurations/sentinel* documented here.

1. Open the Administration Tool and click **Setup > Axway Sentinel/DI**.
2. Select the **Send Events to Axway Sentinel or Decision Insight Server** checkbox to enable the rest of the settings on the page.
3. In the *Axway Sentinel/Decision Insight* pane, specify the connection parameters required for sending event messages to Sentinel:



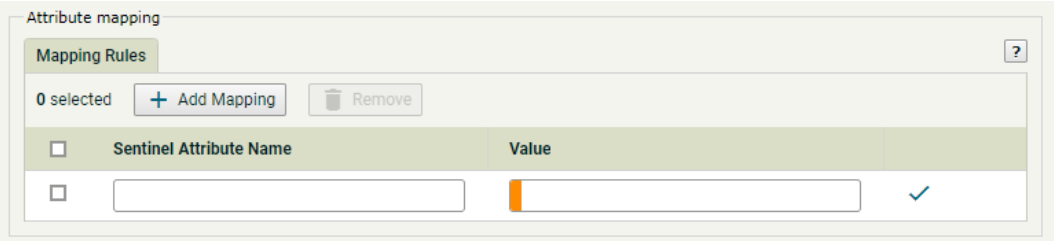
a. In the **Host** field, enter the FQDN or the IP address of the Sentinel server.

b. In the **Port** field, enter either the QLT/XML service port number or the AUTO service port number. These are the TCP/IP ports through which the Sentinel server can receive XML messages (called "Event acquisition ports" in the Sentinel documentation).

c. By default, the Tracked-Event Message is not secured. To secure the communication between SecureTransport and Sentinel, select the **Use Secure Connection** checkbox. You can further increase security by enabling certificate verification or FIPS mode.

Note The Secure Connection settings require a TM restart to take effect.

4. Click the **Test Connection** button. This test indicates whether the port specified on the Sentinel host accepts connections.
5. (Optional) Select **Send Heartbeat to Axway Sentinel Every** and set the heartbeat interval to send periodic messages to Sentinel to tell it that SecureTransport is running and connected. The default interval is 10 seconds.
6. Under *Events*, specify the event states to be reported to Sentinel:
   - To select multiple states for reporting: In the *Available Event States* field, bulk select the desired states. Then, click **>** to move them to the *Event States to Send* field.
   - To report all states: Click **>>**. This will move all available states to the *Event States to Send* field.
     For more information about each state, see *Event states* on page 157.
7. (Optional) In the *Attribute mapping* pane, you create custom **Sentinel Attribute Name** and **Value** attribute mapping rules. To add a new mapping, click the **Add Mapping** button. Specify as many pairs as you need. You can later update or delete them.

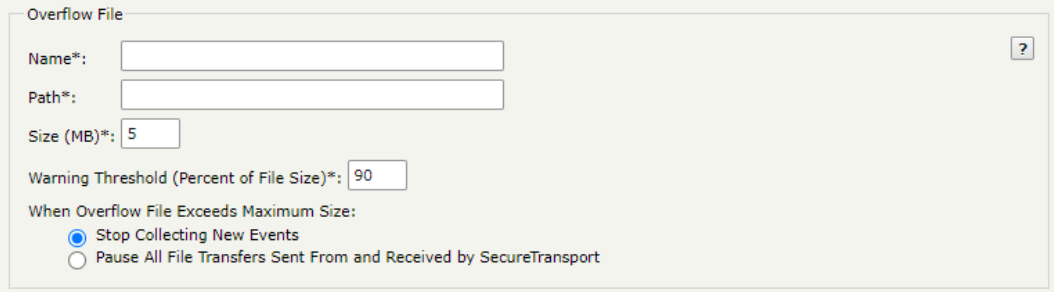


Specifics:

- The mappings created here are saved in the flow attribute context.
- Subscription attributes can be accessed with the expression `${flow.attributes['userVars.ATTRIBUTE_NAME']}`.
- The value of the DXAGENT_ROUTE_SOURCE_FULL_TARGET variable is the absolute path to the file triggered via Advanced Routing. When performing a push transfer via Advanced Routing over PeSIT, the DXAGENT_ROUTE_SOURCE_FULL_TARGET variable is not populated in ACK or ENDED_TO_ACK state.

8. In the *Overflow file* pane, define the attributes of a file to be used as a buffer before the event records are sent to Sentinel. When the Sentinel server is stopped or unavailable due to network issues, this file also acts as intermediary storage for the event records until the connection is restored or the maximum file size is reached, whichever occurs first.

   Note Use the server configuration option `AxwaySentinel.ConnectionTimeout` to define the connection timeout when Sentinel is unavailable. It is set to 5 seconds by default; 0 is interpreted as an infinite timeout and may result in a process hang.



This group of settings applies to all servers in your Enterprise Cluster (EC). Each server must have its own overflow file.

Specify the following information:

Field	Description
Name	The base name of the file to contain buffered SecureTransport event data. The file extension, .dat, is added automatically and cannot be changed. If the file does not exist, SecureTransport will create it.
Path	The path to the directory where the overflow file is located. It could be either a local absolute or relative path. If you specify a relative path, it is created within the `<FILEDRIVEHOME>` directory. Specify a local path, not one in shared storage, because each server in a cluster must have its own overflow file.

Field	Description
Size (MB)	The maximum overflow file size in MB. A 1 MB file can store events from about 150-200 transfers. With a fast and dependable connection between the SecureTransport Server and a dependable Sentinel server, 1-10 MB should be sufficient. If the connection is slow, the file size should be 10-50 MB. When the overflow file size exceeds this size, it invokes the action you configure under **When Overflow File Exceeds Maximum Size**.
Warning Threshold	Specify a percentage of the maximum file size that, when reached, triggers a warning. The value you enter must be an integer between 1 and 94. When the file size reaches the warning threshold, SecureTransport sends a notification to the email address specified in **Setup >Miscellaneous > Notify email** field using the specified *Set up email notifications via SMTP* on page 202. The email message uses the template specified in the `SentinelOverflowFileWarning.xhtml` file under the `<FILEDRIVEHOME>/conf/mailer-templates/` directory.
When Overflow File Exceeds Maximum Size	Select the action to take when the overflow files exceed the maximum size you set. Choose one of the following: **Stop Collecting New Events** – Stop tracking events for Sentinel, but continue to process transfers, run agents and run SecureTransport applications. Use this option in a clustered configuration so that the Sentinel server is not a potential single point of failure. **Pause All File Transfers** – Stop tracking events for Sentinel and stop processing transfers, running agents and running SecureTransport applications. Use this option with care because it is operationally equivalent to stopping SecureTransport. To restart transfers, clear the overflow file or select the other option.

9. Click **Save**.

Configure SecureTransport to maintain link data when Sentinel is disabled

To configure SecureTransport to maintain link data when Sentinel is disabled.

1. Leave the **Send Events to Axway Sentinel or Decision Insight Server** checkbox unchecked.
2. Select the **Maintain link data when Sentinel or Decision Insight reporting is disabled** checkbox.

   The checkbox state is reflected in the `AxwaySentinel.PersistLinkData` server configuration option.
3. Click **Save**.

With this setup, the reporting data is stored in a table called *SentinelLinkData* in the database. In this case, it is advisable to create an instance of the Axway Sentinel Link Data Maintenance application that removes SentinelLinkData entries associated with non-existing files based on a schedule you define.

Test and adapt

When you are ready with the initial configuration, we recommend running a few simple tests to understand how SecureTransport and Sentinel work together. For example, check what is reported in case of a client-initiated download or an upload that triggers a route containing Transformation and Routing steps. Examples of how Sentinel represents successful and failed SIT transfers are available here.

Tracked Objects

For each step of a file transfer process, SecureTransport generates a tracked object (TO) instance, known as *event*, and sends it to Sentinel. This instance is a combination of key-value pairs, where the keys are called attributes. For each tracked object, there is a *State* attribute which identifies a step of the transfer process, for example, `Sending`, `Received`, `Failed`, etc. For details, see *Event states* on page 157.

SecureTransport uses three pre-defined Sentinel tracked objects to report events:

- **XFBTransfer** version 5.5 – to report states that occur during file transmission, for example, `APPLICATION`, `COREID`, and `FILETYPE`. For detailed information, see *XFBTransfer Tracked Object* on page 128.
- **ST_VAS** (Value-Added Services) version 2.0 – to report states that occur during processes other than file transmissions, for example, `DIRECTION`, `USERID`, and `FILESIZE`. For detailed information, see *ST_VAS Tracked Object* on page 153.
- **Heartbeat** version 1.0 – to indicate to Sentinel that SecureTransport is running and connected, for example, `DELAY`. For detailed information, see *Heartbeat Tracked Object* on page 156.

The attributes contained in a tracked object fall into one of two categories:

- **System attributes** that are common to most tracked objects. System attributes identify application and platform events and errors.
- **User attributes** that describe the application- or platform-specific properties of monitored events.

Note Depending on the type of transfer, some Sentinel attributes may not be reported in a `Sending` or `Receiving` state but are correspondingly available for `Sent` or `Received`.

XFBTransfer Tracked Object

The XFBTransfer tracked object is a Sentinel structure used to store transfer-related events coming from the following Axway products: Transfer CFT, Gateway, Gateway Interchange, SecureTransport, and InterPel. The XFBTransfer tracked object is monitored and the errors, tracked-event processing, processing cycles, and tracked-event links are passed to Sentinel.

The following sections list all XFBTransfer attributes:

System attributes

The table presents all system attributes for the XFBTransfer tracked object:

Name	Description
`CYCLEID` *string*	ID used to relate events about the same transfer
`EVENTDATE` *date*	Date (generated by Sentinel)
`EVENTDATETIME` *string*	The date and time event indicating the exact time the event occurred on SecureTransport. This is a UTC standard value and follows the ISO 8601 format.
`EVENTID` *integer*	Unique ID for the event (generated by Sentinel)
`EVENTTIME` *time*	Timestamp (generated by Sentinel)

Name	Description
`GMTDIFF` *integer*	Time zone (generated by Sentinel)
`ISALERT` *bool*	One of the following: • **0**: false • **1**: true, indicates a permanent error or last entry
`ISEND` *integer*	1 or 2 means End
`ISEXCEPTION` *bool*	One of the following: • **0**: false • **1**: true, indicates a temporary failure
`LOCATION` *string*	SecureTransport by default. Its value is controlled by the server configuration option: `AxwaySentinel.Attributes.Attribute.value.Location`
`OBJECTID` *string*	Object ID (generated by Sentinel)
`PRODUCTIPADDR` *string*	Either the FQDN or the hostname of the machine running SecureTransport, depending on the OS, system configuration, and the Java version in place. The hostname is determined via the standard Java `InetAddress.getLocalHost().getHostName()` method. On UNIX systems, as of Java version 11, the method consults first the */etc/hostname* file for resolution and then checks the NSS configuration file (*/etc/nsswitch.con*). With previous Java versions, the NSS configuration file is directly consulted.
`PRODUCTNAME` *string*	SecureTransport
`PRODUCTOS` *string*	Operating system of Axway SecureTransport host (generated by Sentinel)
`RETURNCODE` *integer*	Event exit code
`RETURNMESSAGE` *string*	Error message
`STATE` *string*	State reported by event

XFBTransfer specific attributes

The following tables present the XFBTransfer roles, sender and receivers, product identification, and transfer attributes.

Roles

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`Direction` *integer*	1	One of the following: • **S**: The file is sent (Sender). • **R**: The file is received (Receiver).	All	Direction Corresponds to the `DXAGENT_TRANSFER_DIRECTION` SecureTransport event environment variable.
`IsServer` *integer*	1	One of the following: • **1**: The Sender or the Receiver acts as a Server. • **0**: The Sender or the Receiver is a Requester.	All	Action By

Sender and receivers

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`ReceiverId` *string*	80	Receiver Login Name	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	For PeSIT transfers, corresponds to PeSIT PI 4. If SecureTransport is receiving a file from CFT, the PI 4 value is the login name or the PeSIT ID of the SecureTransport account that receives the file. If SecureTransport is uploading a file to a CFT partner, the PI 4 value corresponds to the transfer site's name or PeSIT ID that identifies the partner. For non-PeSIT transfers, the value of this attribute is the `ip:port` of the remote partner, the hostname, or the SecureTransport login name. For AdHoc transfers: • In case of an incoming AdHoc transfer, the value of this attribute is the account email. • In case of an outgoing AdHoc transfer, the value of this attribute is the recipient email.

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`SenderId` *string*	80	Sender Login Name	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	For PeSIT transfers, corresponds to PeSIT PI3. • In case of SecureTransport receiving a file from CFT, the PI3 value is the name or the PeSIT ID of the transfer site that identifies the partner. • In case of SecureTransport uploading a file to a CFT partner, the PI3 value corresponds to the name or the PeSIT ID of the account that's doing the transfer. For non-PeSIT transfers, reports the SecureTransport login name of the transfer site owner, the hostname, or the `ip:port` of the remote partner where SecureTransport pulled the file from. For AdHoc transfers: • In case of an incoming AdHoc transfer, the value of this attribute is the email of the sender. • In case of an outgoing AdHoc transfer, the value of this attribute is

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
				the account email.
`FinalReceiverId` *string*	80	The Login Name of the final Receiver in case of Store and Forward	All	For PeSIT transfers, corresponds to PeSIT PI 62 or PI 4. • If PI 62 is not present, PI 4 is used in both Sentinel and PeSIT. • In case of SecureTransport initiating a new Store and Forward transfer, the PI 62 value corresponds either to the Final Destination property of the PeSIT transfer site or to the Final Destination property of the Send To Partner step. If both properties are left blank, PI 62 is not populated. • In the PRESERVE store and forward mode, PI preserves the PI 62 value. For non-PeSIT transfers, the value of this attribute is the ReceiverId. In case the ReceiverId value is not present, the value of the FinalReceiverId is the `ip:port` of the Remote Partner or the hostname.

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`OriginalSenderId` *string*	80	The Login Name of the original Sender in case of Store and Forward	All	For PeSIT transfers, corresponds to PeSIT PI 61. • In case of SecureTransport initiating a new Store and Forward transfer, the PI 61 value corresponds either to the Originator property, specified in the PeSIT transfer site settings, or the Originator property in the Send To Partner step settings. If both properties are left blank, PI 61 is not populated. • In the PRESERVE store and forward mode, PI preserves the PI 61 value. For non-PeSIT transfers, the value of this attribute is the login name of the SecureTransport account which received/sent the file from a Remote Partner. If login name is not present, the value is SenderID or UserID.
`Site` *string*	80	Transfer site name	All	For non-PeSIT transfers, reports the transfer site name. If no value is present for the transfer site name, **N2** is reported.

Product identification

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`MonitorVersion` *string*	25	Version of SecureTransport in the X.Y.Z format e.g. 5.3.0 or 5.2.1	All	MonitorVersion
`ProductName` *string*	50	The constant " SecureTransport"	All	ProductName
`ProductIPAddr` *string*	255	Either the FQDN or the hostname of the machine running SecureTransport, depending on the OS, system configuration, and the Java version in place.	All	The hostname is determined via the standard Java `InetAddress.getLocalHost().getHostName()` method. On UNIX systems, as of Java version 11, the method consults first the */etc/hostname* file for resolution and then checks the NSS configuration file (*/etc/nsswitch.con*). With previous Java versions, the NSS configuration file is directly consulted.
`ProductOS` *string*	20	Operating system name as returned by the os.name Java system property.	All	os.name Java system property
`EnvironmentID` *integer*	-	Environment ID as defined in server configuration.	All	EnvironmentID

Users

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`GroupId` *string*	80	The name of the SecureTransport account	All	Account name
`UserID` *string*	80	Transfer site owner	All	For non-PeSIT transfers, reports the transfer site owner. If no transfer site is used, reports the account name of the user who initiated the transfer. When an account template is used, the login name is reported. If no values are present for the account name and the transfer site, **N3** is reported.

Date and time

Sentinel attribute	Description	Applicable states	SecureTransport equivalent
`StartDate` *date*	If the value of the State attribute is: • **SENT**, the value of this attribute is the date on which the Sender began sending the transfer. • **RECEIVED**, the value of this attribute is the date on which the Receiver began receiving the transfer. These dates are expressed in `dd.mm.yyyy` format.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	

Sentinel attribute	Description	Applicable states	SecureTransport equivalent
`StartTime` *time*	If the value of the State attribute is: • **SENT**, the value of this attribute is the local time at which the Sender began sending the transfer. • **RECEIVED**, the value of this attribute is the local time at which the Receiver began receiving the transfer. These times are expressed in `hh:mn:ss` format.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	
`EndDate` *date*	If the value of the State attribute is: • **SENT**, the value of this attribute is the date on which the Sender stopped sending the transfer. • **RECEIVED**, the value of this attribute is the date on which the Receiver stopped receiving the transfer. These dates are expressed in `dd.mm.yyyy` format.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED ROUTED SENT SUSPENDED	
`EndTime` *time*	If the value of the State attribute is: • **SENT**, the value of this attribute is the local time at which the Sender stopped sending the transfer. • **RECEIVED**, the value of this attribute is the local time at which the Receiver stopped receiving the transfer. These times are expressed in `hh:mn:ss` format.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED ROUTED SENT SUSPENDED	

Sentinel attribute	Description	Applicable states	SecureTransport equivalent
`Transmission Duration` *integer*	Transfer duration, expressed in seconds.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED ROUTED SENT SUSPENDED	Duration

Transfer identification

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`Application` *string*	80	Application name.	All	SecureTransport application name.
`CoreId` *string*	100	File identifier. Preserved during all types of processing.	All	Corresponds to the `DXAGENT_CORE_ID` environment variable.
`EventTimeStamp` *string*	100	Event time stamp.	All	

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`FileName` *string*	512	If the value of the CommandType attribute is: • File and the value of the Direction attribute is **E** (Sender), this attribute identifies the file from which the Sender retrieved the transfer data (full path). • File and the value of the Direction attribute is **R** (Receiver), this attribute identifies the file in which the Receiver recorded the transfer data (full path).	All	Full path of the filename Corresponds to the `DXAGENT_FULLTARGET` SecureTransport event environment variable.
`LocalId` *string*	36	Stores the file tracking transfer ID.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	Transfer ID Corresponds to the `TRANSFER_STATUS_ID` SecureTransport event environment variable.

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`ProtocolFile Name` *string*	512	Corresponds to the PeSIT PI 12. Reports the new name of a file that has been renamed using the "Send File As" functionality of an internal or pluggable transfer site.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	File Corresponds to the `DXAGENT_ TARGET` SecureTransport event environment variable.
`ProtocolFile Label` *string*	80	Corresponds to the PeSIT PI 37 – file label. Reports the remote file path and the new name of a file that has been renamed using the "Send File As" functionality.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	When SecureTransport is a sender, this is configured in the Transfer Profile. When SecureTransport is a receiver, this value is configured and supplied by the sender.
`ProtocolId` *string*	80	Corresponds to the PeSIT PI 13 – Transfer ID.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	Transfer ID

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`Protocol Message` *string*	4000	If the value of the CommandType attribute is: • **A**: the value of this attribute is the content of the message sent as part of the acknowledgment. • **M**: the value of this attribute is the content of the PeSIT message • **F**: this attribute is empty.	ACKED ENDED_TO_ACK NACKED	Corresponds to the `DXAGENT_TRANSFER_ACK_MESSAGE` SecureTransport event environment variable.
`Protocol Parameter` *string*	512	Corresponds to PI 99. (ServiceParam or User message); For FTP server-initiated transfers, reports the upload command used in the FTP transfer site.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	User message Upload command in FTP transfer sites

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`User Parameter1` *string*	255	Contains information whether the transfer is with an Internal partner, External partner or Unknown. • For transfer with an Internal partner **I** is reported. • For transfer with an External partner **E** is reported. • If the Administrator has not specified Internal or External – **N** is reported.	All	TransferType in account or site name.
`User Parameter2` *string*	255	Reports the account type: • **User** (meaning that account has been specified in SecureTransport's database). • **Template** (meaning that this is a template). • **Service** (Service account).	All	Account Type
`ParentCycleID` *integer*	-	ID of the parent processing cycle.	SENDING SENT TO_EXECUTE	ParentCycleID

Transfer protocols

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`ServerName` *string*	250	Name of the server that processed the transfer. Applicable when the Sender or the Receiver is a Requester, except for the ENDED_TO_ACK state, where the attribute is set `no matter`.	ACKED CANCELED DELETED ENDED_TO_ACK FAILED INTERRUPTED NACKED RECEIVED RECEIVING RENAMED SENDING SENT	Corresponds to the `DXAGENT_SERVERNAME` SecureTransport event environment variable.
`Protocol` *string*	25	Name of the protocol that operates at the Protocol Layer of the transfer.	All	Corresponds to the `DXAGENT_PROTOCOL` SecureTransport event environment variable.
`IsSSL` *string*	1	One of the following: • **1**: SSL/TLS used for the transfer. • **0**: SSL/TLS not used for the transfer.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	Use TLS/SSL in transfer site.

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`SSLAuth` *string*	1	One of the following: • **S**: The Server sent X.509 certificates to the Requester. For SSH sessions, the value of SSLAuth will be always **S** if the Requester does not present a key. • **B**: Both the Server and the Requester sent X.509 certificates to each other. • **N**: Neither the Server nor the Requester sent X.509 certificates.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	`DXAGENT_SSLAUTH`

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`SSLCypher` *integer*	-	One of the following: • the **RFC code** of the cipher suite that the Server and the Requester used during the SSL/TLS session. The cipher suite identifies the authentication method, the encryption algorithm, and the hash algorithm for MAC calculation. • **0** for all SSH sessions.	All	`DXAGENT_ SESSION_SSL_ CYPHER`

Transfer options

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`Compression` *string*	1	One of the following: • **0**: Undefined • **1**: Horizontal • **2**: Vertical • **3**: Both horizontal and vertical • **4**: Not compressed Corresponds to PI21.	ACKED ENDED_TO_ ACK INTERRUPTED NACKED RECEIVING RECEIVED ROUTED SENDING SENT SUSPENDED	Compression in transfer site.

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`RetryMaxNumber` *integer*	-	Maximum number of times that the Sender can attempt to send transfers.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	`EventQueue.maxRetryCount` on the *Server Configuration* page.
`RetryNumber` *integer*	-	Number of times that the Sender attempted to send the transfer. Each time the Sender established a connection with the Receiver, the Sender counted one attempt.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	Corresponds to the `DXAGENT_PERSISTED_EVENT_RETRY_COUNT` SecureTransport event environment variable.

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`Request Type` *string*	1	One of the following: • **S**: The Sender sent a single transfer to a single Receiver. This corresponds to a normal file transfer. • **F**: The Sender sent a group of transfers to a single Receiver. For each transfer in the group, the product generated one Processing Cycle. This corresponds to multiselect PeSIT option. • **D**: The Sender sent a single transfer to a group of Receivers (diffusion). For each Receiver in the group, the product generated one Processing Cycle. This corresponds to send to multiple transfer sites SecureTransport configuration.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	SecureTransport always reports as value for RequestType.

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`Transfer Type` *string*	1	One of the following: • **S**: The Sender sent a single transfer to a single Receiver. This corresponds to a normal file transfer. • **F**: The transfer belongs to a group of transfers that the Sender sent to a single Receiver. For each transfer in the group, the product generated one Processing Cycle. This corresponds to multiselect PeSIT option. • **D**: The Receiver belongs to a group of Receivers to whom the Sender sent the transfer (diffusion). For each Receiver in the group, the product generated one Processing Cycle. This corresponds to send to multiple transfer sites	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	SecureTransport always reports as value for RequestType

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
		SecureTransport configuration.		

Transfer size

Sentinel attribute	Description	Applicable states	SecureTransport equivalent
`FileSize` *integer*	An estimation of the file size given at the beginning of the transfer and updated upon completion of the transfer with the real value checked by ST using the file system. Corresponds to PI42.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	
`TransmittedBytes` *integer*	Number of bytes transferred, after decompression, to transfer the file. This size is expressed in bytes. Corresponds to PI27. **Note**: For PeSIT, this value sent is crosschecked by both the sender and receiver.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	Corresponds to the `DXAGENT_TRANSFERRED_BYTES` SecureTransport event environment variable.

Transfer structure and content

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`CommandType` *string*	1	One of the following: • **F**: File transfer • **A**: Acknowledgment • **M**: Message	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	
`FileType` *string*	60	One of the following: • **B**: the transferred file is a Binary file. • **T**: the transferred file is a Text file.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	Transfer Mode in Transfer Profile or Data Encoding in Send To Partner step.
`RecordNumber` *integer*	-	Number of record in the file. This size is expressed in bytes. **Note**: For PeSIT, this value sent is crosschecked by both the sender and receiver. Corresponds to PeSIT PI 28.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`RecordFormat` *string*	64	One of the following: • **F**: **fixed**: The transferred data contains fixed-length records. • **V**: **variable**: The transferred data contains variable-length records. Corresponds to PeSIT PI 31.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	Record Format in Transfer Profile or in Send To Partner step.
`RecordSize` *integer*	-	One of the following: • If the value of RecordFormat attribute is **fixed**, the value of this attribute is the size of all records in the transferred file, expressed in bytes. • If the value of RecordFormat is **variable** or **undefined**, the value of this attribute is the size of the largest record in the transferred file, expressed in bytes. Corresponds to PeSIT PI 32.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	Record Length in Transfer Profile or in Send To Partner step.

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`Transcoding` *integer*	-	Character code of the transferred data: • **A**: ASCII • **B**: Binary • **E**: EBCDIC From PI 16.	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	Transfer Mode in Transfer Profile or Data Encoding in Send To Partner step.

Other attributes

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`VirtualDirName` *string*	255	The current folder relative to the home folder.	All	`DXAGENT_TARGET_PATH`
`GroupName` *string*	80	The Business Unit name.	All	`DXAGENT_BUSINESS_UNIT_NAME`
`IdAppl` *string*	255	The ID of the route template package that is executed.	FAILED POST_PROC/ROUTED POST_PROC/ROUTING	`DXAGENT_ROUTE_EXECUTION_ID`
`SessionTag` *string*	255	Represents SecureTransport Session ID. The Session ID could be used to query the server log for a particular session.	All	Session ID

Sentinel attribute	Length	Description	Applicable states	SecureTransport equivalent
`TransferTag` *string*	50	Represents SecureTransport transfer status operationIndex. The Transfer ID could be used to query the file tracking log for a particular transfer .	All	Transfer ID of the current transfer.
`RemoteAddr` *string*	255	Partner remote address (IPv4 and IPv6).	ACKED ENDED_TO_ACK INTERRUPTED NACKED RECEIVED RECEIVING ROUTED SENDING SENT SUSPENDED	The partner's remote address.
`UserState` *string*	-	User state to track pre- or post-transfer processing.	ARCHIVED DENIED ICAP_DENIED ICAP_SCANNED ICAP_SCANNING ROUTED ROUTING SCANNED SCANNING	

ST_VAS Tracked Object

The ST_VAS tracked object is used by SecureTransport to monitor the added value processes on files outside of the transmission. The event messages for this SecureTransport specific tracked object are fully customizable.

The following sections list all ST_VAS attributes:

- *ST_VAS Tracked Object on page 153*
- *Specific attributes on page 155*

System attributes

The table presents all system attributes for the ST_VAS tracked object:

Name	Description
`CYCLEID` *string*	ID used to relate events about the same transfer
`EVENTDATE` *date*	Date (generated by Sentinel)
`EVENTDATETIME` *string*	The date and time event indicating the exact time the event occurred on SecureTransport. This is a UTC standard value and follows the ISO 8601 format.
`EVENTID` *integer*	Unique ID for the event (generated by Sentinel)
`EVENTTIME` *time*	Timestamp (generated by Sentinel)
`GMTDIFF` *integer*	Time zone (generated by Sentinel)
`ISALERT` *bool*	1 means permanent error or last entry
`ISEND` *integer*	1 or 2 means End
`ISEXCEPTION` *bool*	1 means temporary failure
`LOCATION` *string*	SecureTransport by default. Its value is controlled by the server configuration option: `AxwaySentinel.Attributes.Attribute.value.Location`
`OBJECTID` *string*	Object ID (generated by Sentinel)

Name	Description
`PRODUCTIPADDR` *string*	Either the FQDN or the hostname of the machine running SecureTransport, depending on the OS, system configuration, and the Java version in place. The hostname is determined via the standard Java `InetAddress.getLocalHost().getHostName()` method. On UNIX systems, as of Java version 11, the method consults first the */etc/hostname* file for resolution and then checks the NSS configuration file (*/etc/nsswitch.con*). With previous Java versions, the NSS configuration file is directly consulted.
`PRODUCTNAME` *string*	SecureTransport
`PRODUCTOS` *string*	Operating system of Axway SecureTransport host (generated by Sentinel)
`RETURNCODE` *integer*	Event exit code
`RETURNMESSAGE` *string*	Error message
`STATE` *string*	State reported by event

Specific attributes

In addition to the common system attributes, the following are specific to ST_VAS:

Name	Description
`ACTIVITYDURATION` *integer*	Duration of activity in milliseconds
`ACTIVITYNAME` *string*	Name of activity reporting the event
`DIRECTION` *string*	R for receive, S for send
`DIRECTORYNAME` *string*	Directory name
`FILENAME` *string*	Path and name of the transferred file

Name	Description
FILESIZE *integer*	Size in bytes of transferred file
ORIGINALFILENAME *string*	File name before the action of the activity
USERID *string*	Axway SecureTransport Login Name/Account Name

Heartbeat Tracked Object

The Heartbeat tracked object enables you to monitor the functional presence of the applications monitored by Sentinel.

Axway provides the Heartbeat tracked object together with the Sentinel product. To test this functionality in SecureTransport, you must first log into the Synchrony Composer and import Heartbeat. Then you must either generate tracked event records for Heartbeat events, or use a correlation rule to trigger an alert in case of not receiving a Heartbeat message.

The following sections list all Heartbeat attributes:

- *System attributes on page 156*
- *Specific attributes on page 157*

System attributes

The table presents all system attributes for the Heartbeat tracked object:

Name	Description
CYCLEID *string*	ID used to identify the SecureTransport Server
EVENTDATE *date*	Date (generated by Sentinel)
EVENTDATETIME *string*	The date and time event indicating the exact time the event occurred on SecureTransport. This is a UTC standard value and follows the ISO 8601 format.
EVENTTIME *time*	Timestamp (generated by Sentinel)
GMTDIFF *integer*	Time zone (generated by Sentinel)

Name	Description
`LOCATION` *string*	SecureTransport by default. Its value is controlled by the server configuration option: `AxwaySentinel.Attributes.Attribute.value.Location`
`PRODUCTIPADDR` *string*	Either the FQDN or the hostname of the machine running SecureTransport, depending on the OS, system configuration, and the Java version in place. The hostname is determined via the standard Java `InetAddress.getLocalHost().getHostName()` method. On UNIX systems, as of Java version 11, the method consults first the */etc/hostname* file for resolution and then checks the NSS configuration file (*/etc/nsswitch.con*). With previous Java versions, the NSS configuration file is directly consulted.
`PRODUCTNAME` *string*	SecureTransport
`PRODUCTOS` *string*	Operating system of Axway SecureTransport host (generated by Sentinel)

Specific attributes

In addition to the common system attributes, the following one is specific to Heartbeat:

Name	Format	Description
`DELAY`	integer	Heartbeat interval set on the *Sentinel Events* page

Event states

An *event state* indicates the current state of a file transfer. Each transfer passes through different states during execution and Sentinel receives an event each time the state changes. Your selection of event states to report determines the type of information that SecureTransport sends to Sentinel. For details, see *Configure Sentinel reporting on page 123*.

The following table describes the available event states:

Event state	Description	Required by Sentinel	Selected by default and reported via XFBTransfer	State type
ACKED	SecureTransport has sent a positive acknowledgment for a PeSIT transfer.	No	Yes	Processed
AVAILABLE (ST_ VAS)	SecureTransport application has completed successfully.	No	No	Processed
AVAILABLE (XFBTransfer)	SecureTransport has published a file in a target folder using the Publish To Account step.	Yes	Yes	Processed
CANCELED	An application, agent, or user has canceled a file transfer.	Yes	Yes	Processed
DECRYPTED	SecureTransport has performed a successful PGP decryption.	No	No	Processed
DECRYPTING	SecureTransport is starting to decrypt a file.	No	No	In-process
DELETED	A client, post-processing action, File Maintenance application, Subscription Folder purge, or post-client download action has deleted a file.	No	No	Processed
ENCRYPTED	SecureTransport has performed a successful PGP encryption.	No	No	Processed

Event state	Description	Required by Sentinel	Selected by default and reported via XFBTransfer	State type
ENCRYPTING	SecureTransport is starting to encrypt a file.	No	No	In-process
ENDED_TO_ACK	SecureTransport has received an acknowledgment for a transfer.	No	Yes	Processed
ERROR	An error has occurred during file deletion, renaming, transformation, or routing.	No	No	Processed
FAILED	An error has occurred during a file transfer, Advanced Routing execution, or while an agent was running.	Yes	Yes	Processed
FORWARDED	A routing application (such as Standard Router) has completed successfully.	No	No	Processed
FORWARDING	SecureTransport is starting a routing application (such as Standard Router).	No	No	In-process
INTERRUPTED	A remote PeSIT server paused a transfer it initiated.	No	Yes	Processed
NACKED	A negative acknowledgment for a PeSIT file transfer has been sent manually.	No	Yes	Processed

Event state	Description	Required by Sentinel	Selected by default and reported via XFBTransfer	State type
POST_PROC/ICAP_DENIED	Access to the file is denied.	No	No	Processed
POST_PROC/ICAP_SCANNED	The scanning of the file has finished successfully.	No	No	Processed
POST_PROC/ICAP_SCANNING	The scanning of the file has started.	No	No	In-process
POST_PROC/ROUTED	An Advanced Routing application has finished successfully .	No	Yes	Processed
POST_PROC/ROUTING	SecureTransport is starting an Advanced Routing application.	No	Yes	In-process
PRESERVED	The original file was not deleted after a decryption or encryption.	No	No	Processed
RECEIVED	SecureTransport has successfully received a file by a server-initiated pull or a client-initiated push.	Yes	Yes	Processed
RECEIVING	SecureTransport is starting to receive a file by a server-initiated pull or a client-initiated push.	No	Yes	In-process
RENAMED	A client action, a post-transmission action, or a post-processing action has renamed a file.	No	No	Processed

Event state	Description	Required by Sentinel	Selected by default and reported via XFBTransfer	State type
ROUTED	As the intermediate partner in a routed PeSIT transfer, SecureTransport has sent a file to the routing destination.	No	Yes	Processed
SENDING	SecureTransport is starting to send a file by a server-initiated push or a client-initiated pull.	No	Yes	In-process
SENT	SecureTransport has successfully sent a file by a server-initiated push or a client-initiated pull.	Yes	Yes	Processed
SUBMITTED	SecureTransport has sent a wildcard pattern for a server-initiated pull.	No	Yes	Processed
TO_BE_DELETED	A File Maintenance application has marked a file for removal.	No	Yes	In-process
TO_EXECUTE	SecureTransport is ready to start a server-initiated transfer. TO_EXECUTE is reported as the first state in cycle graphs for server-initiated transfers.	No	Yes	In-process
WAITING	SecureTransport is waiting for all files from the contents of a trigger file to be received before proceeding with the flow.	No	Yes	In-process

Specifics:

- Event states that are not reported via XFBTransfer are reported using ST_VAS. See *Tracked Objects* on page 127.
- In-process states correspond to ongoing events.
- Processed states mark the successful or unsuccessful completion of a process.
- Additional event states can be reported via added data transformations and other SecureTransport customizations.

CycleId

The CycleId is an XFB Transfer Tracked Object attribute. It is the unique transfer identifier for a file transfer. The CycleID is calculated based on a set of parameters that are exchanged at the beginning of the transfer, or parameters that make the transfer unique in the product. By linking CycleIDs together, Sentinel can provide end-to-end monitoring of file transfer flows.

This topic describes the internal CycleId structure for PeSIT and SFTP protocol transfers. The CycleId of the original transfer is preserved when retrying or resubmitting failed server-initiated transfers.

CycleId calculation for PeSIT transfers

The CycleID for a PeSIT transfer is a string that represents a concatenation of the following information:

Offset	Length	PI/Value	Description
1	4	"SUIV"	Eye catcher
5	24	PI3 CONNECT	For Transmission
		PI4 CONNECT	For Reception
29	24	PI4 CONNECT	For Transmission
		PI3 CONNECT	For Reception

Offset	Length	PI/Value	Description
53	5	"0 "	For File Transfer
		"65535"	For Message Transfer
		"REPLY"	For Acknowledgment
58	76	PI12	Logical file name
134	8	PI13*	Transfer ID
142	12	PI51	Only the date is used (YYMMDD), and the time is filled with 6 spaces.
154	1	E	For Transmission
		R	For Reception

* The *Transfer ID* (PI13) of the original transfer is preserved in the following cases:

- when resubmitting a successful inbound transfer
- when resubmitting a permanently failed transfer
- when retrying a temporarily failed transfer.

A resubmitted successful outbound transfer is assigned a different PI13 value than the original transfer. This will lead to the creation of a new CycleId.

CycleId calculation for SFTP transfers

The CycleID for an SFTP transfer is a string that represents a concatenation of the following information:

Offset	Length	PI/Value	Comments
1	4	SUIV TEMP	Eye catcher
5	24	Sender identifier	Sender account login name SEND: System login of the process that runs the SFTP client RECV: SFTP login name sent by the client to connect to the SFTP server

Offset	Length	PI/Value	Comments
29	24	Receiver identifier	Receiver login name SEND: SFTP login name sent by the client to connect to the SFTP server RECV: System login of the process that runs the SFTP client
53	5	"0"	For File Transfer
58	76	Virtual Filename	Logical file name, NIDF for CFT
134	8	Sequence number	Unique number identifying the transfer sent, NIDT for CFT
142	12	Date YYMMDD /padded to 12 with spaces on the right/	Only the date is used (YYMMDD), and the time is filled with 6 spaces
154	1	E	For Transmission
		R	For Reception

Generating common CycleID for end-to-end tracking of SFTP transfers

It is possible to reuse the CycleId of incoming files for SFTP transfers (CIT uploads). This helps with the end-to-end tracking of transfers in Sentinel. To report common CycleId, set the configuration option `Ssh.AxwayVendorExtensions.enabled` to `true`. Changing the value requires restarting the SSH services and the Transaction Manager on all Backend and Edge instances.

When SecureTransport acts as Client and Transfer CFT acts as Server, follow the guidelines:

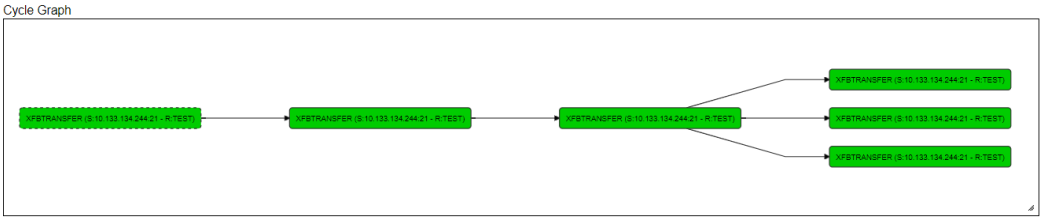
- Name each of your transfer sites to exactly match the corresponding NIDF name in Transfer CFT
- Use only capital letters for each SecureTransport user's account name (Login name).

Sentinel Graphs

The SecureTransport to Sentinel integration adds extended visibility and data tracking functionality. When enabled, SecureTransport sends information to Axway Sentinel each time a pre-determined event occurs. Sentinel records the data in a database table with a structure that is dictated by a Tracked Object and presents it graphically. It provides two options for viewing the processing cycle of events sent by SecureTransport: Cycle Graph and Life Cycle.

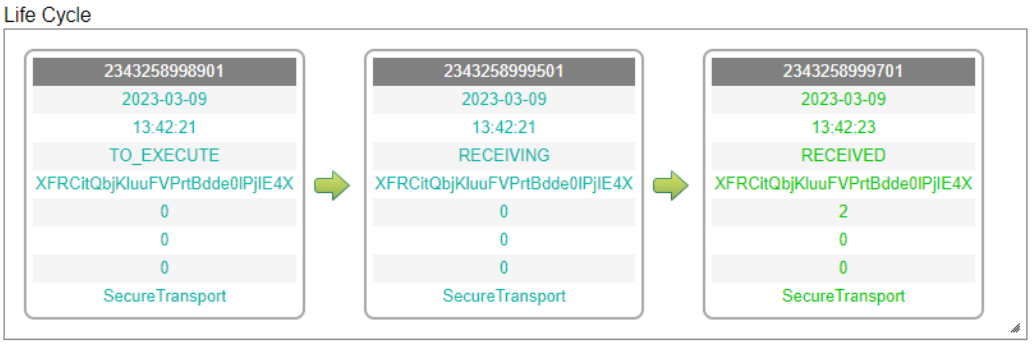
Cycle Graph

Cycle Graph displays a diagram of the processing cycle of events from many tracked objects. It is composed of a series of nodes representing the events linked to the selected record.



LifeCycle

Life Cycle presents a graph of all history events of the selected record. It shows all the (permanent and temporary) states the transfer goes through, allowing you to view the flow of the product life cycle from event to event. Each event is displayed in a box that gives an at-a-glance view of the most important transfer attributes in the following sequence:



- `EventId`- System attribute that has unique values for each tracked event.
- `EventDate` - Date of the event.
- `EventTime` - Time of the event.
- `State` -Identifies the processing phase in which an event occurs. For details, see *Event states* on page 157.
- `CycleID` - Sentinel uses the CycleId attribute to identify every tracked event that belongs to a given processing cycle, as well as processing cycles that sequentially precede and follow one another. For each tracked-event message in a given processing cycle, the value of CycleId is the same. For details, see *CycleId* on page 162.
- `IsEnd` - Indicates whether the state is permanent or temporary.
- `IsException`- Indicates whether the transfer ended with an exception (temporary error).
- `IsAlert`- Indicates whether the transfer ended a permanent error.

The event is displayed using a color code. Its value is calculated by Sentinel as a function of the three attributes `isAlert, isEnd` and `isException`.

Clicking on a box opens the *Event Details*, where you can see all attributes of the selected event. For detail description of each and possible values, see *XFBTransfer Tracked Object* on page 128.

Learn more with examples.

Axway Sentinel requests

SecureTransport provides three requests you can use to display events and cycle links:

- `CurrentAlerts` displays all `ST_VAS` and `XBFTransfer` alerts (events with `IsAlert=1`).
- `CurrentDayTransfersAndVAS` displays all `ST_VAS` and `XBFTransfer` events for the current day.
- `TransfersAndVAS` displays all `ST_VAS` and `XBFTransfer` events.

Sentinel Graphs explained with examples

This topic illustrates how file transfers and transfer flows in SecureTransport are reported and presented in Sentinel. It provides examples of how the processing cycle of a file transfer is represented in Sentinel graphs.

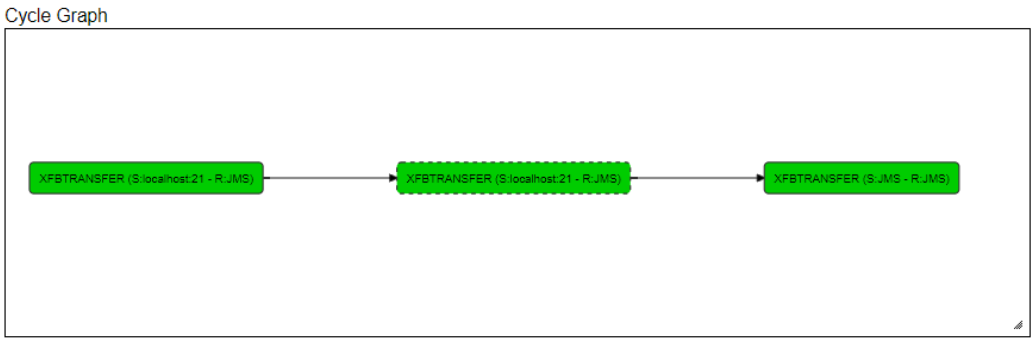
Let's look at the following scenario: SecureTransport performs a SIT pull through a subscription and push the downloaded file through the Send To Partner step.

For completeness, we will describe multiple cases to cover the different outcomes:

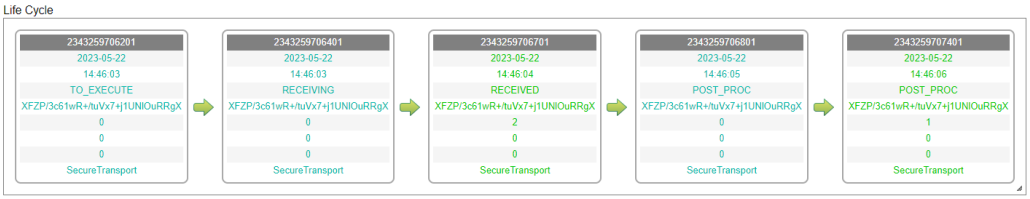
- Successful SIT pull and push
- Successful listing but no files to pull
- SIT pull failure
- SIT push failure
- Trigger a route based on received acknowledgment

Successful SIT pull and push

Upon a SIT pull, SecureTransport first performs the listing and then the actual download. On the *File Tracking* page, there will be a single entry for a SIT pull. However, SecureTransport sends two events to Sentinel - one for the listing and one for the download operation.
In Sentinel, the Cycle Graph view presents a diagram with three nodes - the first two nodes correspond to the successful SIT pull, and the third one represents the successful outbound transfer via the Send To Partner.



The Sentinel Life Cycle view display the history events of the selected record.



In our scenario, it shows five changes in the transfer state:

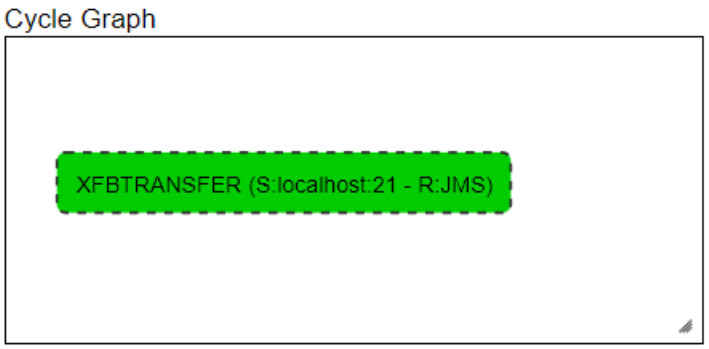
1. TO_EXECUTE indicates a server-initiated pull.
2. RECEIVING: The file is being downloaded.
3. RECEIVED: The file download is complete.
4. POST_PROC: The file is being transmitted.
   The combination of attributes `IsEnd=0, IsException=0, IsAlert=0` in the box means "transmission in progress, no errors, no exceptions".
5. POST_PROC: The file transmission is complete.
   The combination of attributes `IsEnd=1, IsException=0, IsAlert=0` in the box indicates that the transfer finished successfully.

The colors of the boxes indicate a successful transfer.

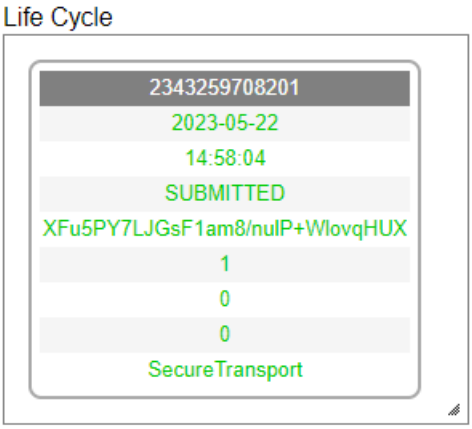
Successful listing but no files to pull

The example above showed the case where both the SIT pull and the subsequent push were successful. Now, let's see what happens if the listing is successful, but there are no files on the remote site.

In Cycle Graph: one node representing the successful pull. There is no file to pull, so the AR step is not executed.



In Life Cycle: one cycle graph event with a SUBMITTED state.



An event with the state SUBMITTED is reported as a parent cycle for any server-initiated wildcard pull transfer. The value of the FILENAME attribute is the filename pattern used for the pull. For example, if the download pattern is *, then the FILENAME attribute value is `*`.

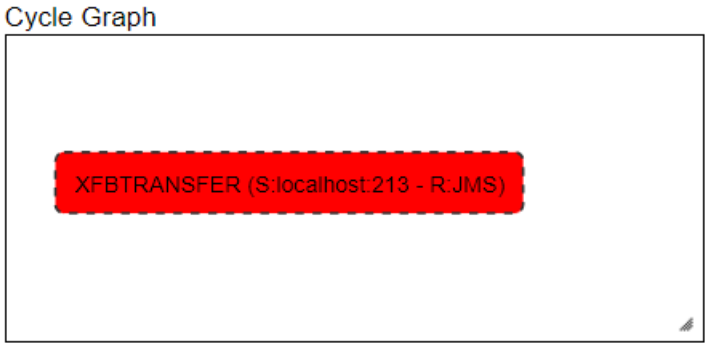
Failed SIT transfers

This section illustrates how Sentinel presents SIT pull and push failures in SecureTransport.

SIT Pull failure

Let's say we have the same steps as above - a SIT Pull and a subsequent push via Send to Partner.

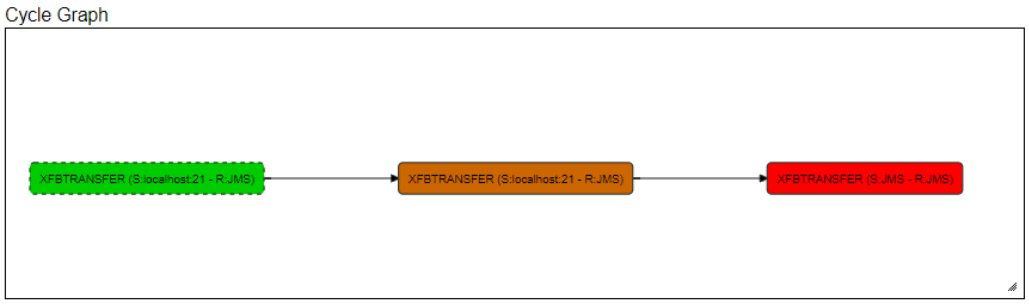
If the pull fails, Sentinel will display one event colored in red.



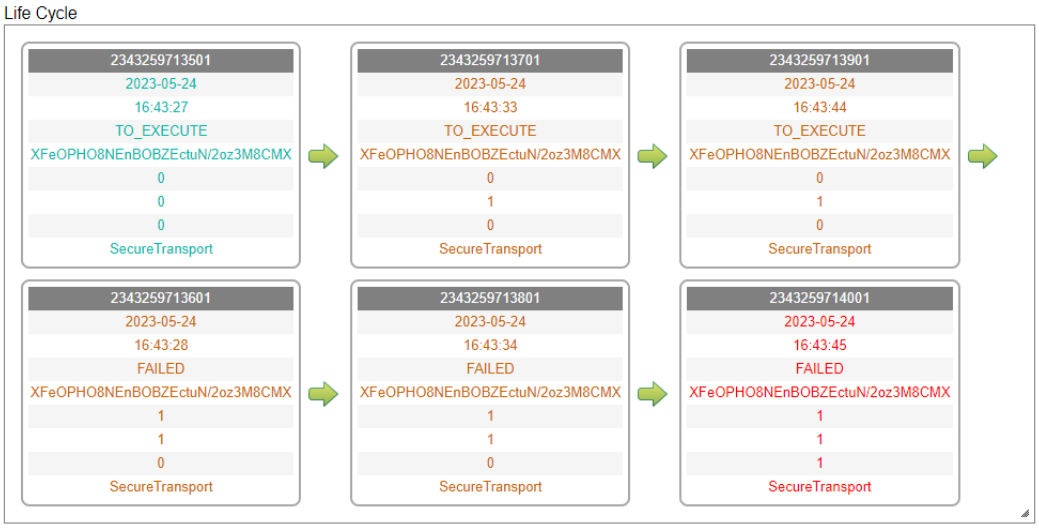
SIT Push failure

If the pull is successful, but the Send to Partner fails to send the file, the graphs in Sentinel will look like this.

In Cycle Graph: successful pull, retry, failed push.



In Life Cycle: successful pull, four retries, permanent failure.



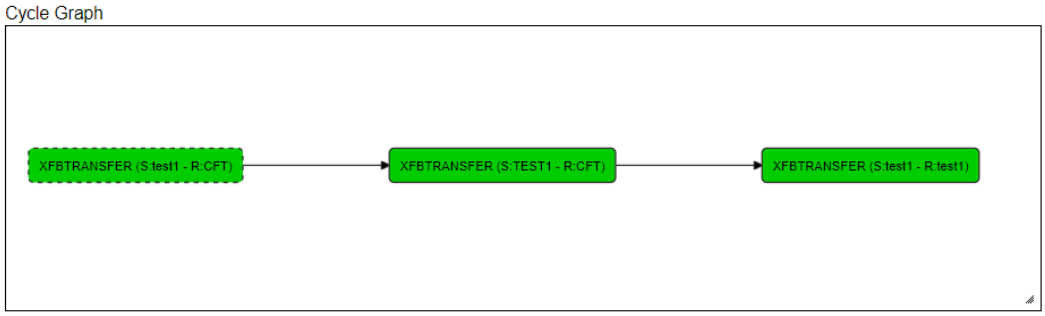
The last event message box is red due to the FAILED state. The combination `IsEnd=1, IsException=1, IsAlert=1` indicates that the transfer resulted in a permanent failure. The brown color of the preceding boxes indicates a retry operation; the retry threshold is not yet reached, so SecureTransport reports a temporary failure (`IsAlert=0`).

Trigger a route based on received acknowledgment

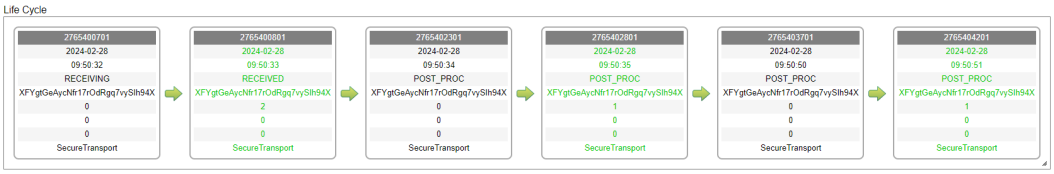
Let's look at the following scenario: Partner A uploads a file to SecureTransport, which triggers an Advanced Routing flow that sends a file to Partner B via PeSIT. The arrival of an acknowledgment (ACK or NACK) from Partner B triggers the second route in the flow that forwards the file to Partner C.

The Cycle Graph in Sentinel presents a diagram with three nodes:

- the first one is the file upload
- the second one represents the outgoing transfer (the sending of the file to Partner B) plus the receiving of the acknowledgment
- the last node is the second outgoing transfer (the sending of the file to Partner C) that was triggered by the acknowledgment



The Sentinel Life Cycle view displays the history of events:



1. RECEIVING: A file is being uploaded to SecureTransport.
2. RECEIVED: SecureTransport received the file successfully.
3. POST_PROC: The file is being transmitted to Partner B.
   The combination of attributes `IsEnd=0, IsException=0, IsAlert=0` in the box means "transmission in progress, no errors, no exceptions".
4. POST_PROC: The file transmission to Partner B is complete.
   The combination of attributes `IsEnd=1, IsException=0, IsAlert=0` in the box indicates that the transfer finished successfully.
5. POST_PROC: The file is being transmitted to Partner C.
6. POST_PROC: The file transmission to Partner C is complete.

PeSIT states and roles explained with examples

This topic provides examples of roles and states for transfers over the PeSIT protocol.

Roles of transfer partners

A transfer occurs between only two transfer partners. Each partner plays two roles:

- **Sender** or **Receiver** (the file is sent or received)
- **Requester** or **Server** (the transfer request is sent or received)

For a given transfer, the only possible combinations of partner roles are the following:

- **Requester/Sender** and **Server/Receiver** (The partner that sent the file requested the transfer).
- **Requester/Receiver** and **Server/Sender** (The partner that received the file requested the transfer).

Server/Sender transfer states

The role of Server/Sender corresponds to a client-initiated download in SecureTransport terminology, where SecureTransport acts as a protocol server and sends a file.

State	Description
`CANCELED`	The file transfer has been canceled by the client.
`ENDED_TO_ACK`	SecureTransport has received an acknowledgment for the file transfer.
`INTERRUPTED`	The file transfer has been locally suspended.
`SENDING`	SecureTransport is sending a file.
`SENT`	SecureTransport has sent a file.
`SUSPENDED`	The file transfer has been remotely suspended.

Server/Receiver transfer states

The role of Server/Receiver corresponds to a client-initiated upload in SecureTransport terminology, where SecureTransport acts as a protocol server and receives a file.

State	Description
ACKED	SecureTransport has sent a positive acknowledgment for the file transfer.
CANCELED	The file transfer has been canceled by the client.
INTERRUPTED	The file transfer has been locally suspended.
NACKED	A negative acknowledgment for the file transfer has been sent manually.
POST_PPROC	Post-processing in progress.
RECEIVED	SecureTransport has received a file.
RECEIVING	SecureTransport is receiving a file.
ROUTED	The file transfer has been successfully routed (only on the relay site in Store and Forward mode).
SUSPENDED	The file transfer has been remotely suspended.

Requester/Sender transfer states

The role of Requester/Sender corresponds to a server-initiated push in SecureTransport terminology, where SecureTransport acts as a protocol client and sends a file.

State	Description
CANCELED	The file transfer has been canceled by the client.
ENDED_TO_ACK	SecureTransport has received an acknowledgment for the file transfer.
INTERRUPTED	The file transfer has been locally suspended.
POST_PROC	Post-processing in progress.
ROUTED	The file transfer has been successfully routed (only on the relay site in Store and Forward mode).

State	Description
SENDING	SecureTransport is sending a file.
SENT	SecureTransport has sent a file.
SUSPENDED	The file transfer has been remotely suspended.
TO_EXECUTE	SecureTransport is about to execute a scheduled job.

Requester/Receiver transfer states

The role of Requester/Receiver corresponds to a server-initiated pull in SecureTransport terminology, where SecureTransport acts as a protocol client and receives a file.

State	Description
ACKED	SecureTransport has sent a positive acknowledgment for the file transfer.
INTERRUPTED	The file transfer has been locally suspended.
NACKED	A negative acknowledgment for the file transfer has been sent manually.
POST_PROC	Post-processing in progress.
RECEIVED	SecureTransport has received a file.
RECEIVING	SecureTransport is receiving a file.
SUSPENDED	The file transfer has been remotely suspended.
TO_EXECUTE	SecureTransport is about to execute a scheduled job.

Integrate Decision Insight

Decision Insight is a Business Activity Monitoring (BAM) product that collects, aggregates, correlates, and reports events from SecureTransport and other products, applications, and systems throughout your infrastructure. Decision Insight is a separate product that you can buy from Axway or an authorized partner. Once you license and configure SecureTransport to send file transfer and processing events to Decision Insight, data is collected and displayed on a dashboard.

Note Decision Insight does not support Attribute mapping in SecureTransport.

Note Decision Insight will not monitor ad hoc activity.

For addition information please refer to Embedded Analytics for SecureTransport.

The following topics provide additional for the SecureTransport Decision Insight integration:

- *Event states* on page 174 - Describes the Decision Insight event states.
- *Tracked objects* on page 175 - Lists the attributes of the Tracked Objects used to report Axway SecureTransport events to Decision Insight.
- *XFB Transfer tracked objects* on page 176 - Describes the XFB transfer tracked objects.
- *Configure SecureTransport to send events to Decision Insight* on page 176 - Provides how-to instructions for configuring SecureTransport to send events to Decision Insight.

Event states

An event state specifies the current state of a file transfer. Every event in the same group about one file transfer is identified using the same cycle ID. For Axway Sentinel events information, refer to *Event states* on page 157.

The following event states indicate a transfer start and end.

Direction	Start	Success	Failure	Cancellation
Incoming	RECEIVING	RECEIVED	FAILED	CANCEL
Outgoing	SENDING	SENT	FAILED	CANCEL

The following event states indicate a post processing action.

Event state	Description
DECRYPTED	SecureTransport has performed a successful PGP decryption.
DECRYPTING	SecureTransport is starting to decrypt a file.
DELETED	A client, post-processing action (PPA), or post client download action has deleted a file.
ENCRYPTED	SecureTransport has performed a successful PGP encryption.
ENCRYPTING	SecureTransport is starting to encrypt a file.
POST_ PROC/ARCHIVED	The file has been archived.
POST_PROC/ICAP_ DENIED	Access to the file is denied.

Event state	Description
POST_PROC/ICAP_SCANNED	The scanning of the file has successfully finished.
POST_PROC/ICAP_SCANNING	The scanning of the file has started.
POST_PROC/ROUTED	An Advanced Routing application has successfully completed.
POST_PROC/ROUTING	SecureTransport is starting an Advanced Routing application.
RENAMED	A client action, or a post-transmission action (PTA), or post-processing action has renamed a file.
ROUTED	As the intermediate partner in a routed PeSIT transfer, SecureTransport has sent a file to the routing destination.

Tracked objects

This topic lists the attributes of the Tracked Objects used to SecureTransport events to Decision Insight. For information on all the of standard attributes, refer to *Tracked Objects* on page 127.

The attributes that Decision Insight uses to build the dashboards are:

Name	Format	Description
COREID	string	File identifier reported with every state
CYCLEID	string	ID used to relate events about the same transfer.
DIRECTION	string	R for receive, S for send
FINALRECEIVERID	string	Name of the final receiver of the transfer
GROUPNAME	string	Axway SecureTransport account business unit
ISALERT	0/1	1 means permanent error on last entry
ISEXCEPTION	0/1	1 means temporary failure
ORIGINALSENDERID	string	Name of the original sender of the transfer

Name	Format	Description
RECEIVERID	string	Name of the receiving partner
REQUESTGROUPID	string	Local identifier of the group to which the requesting user belongs
REQUESTUSERID	string	Local identifier of the user who requested the transfer
SENDERID	string	Name of the sending partner
USERID	string	Axway SecureTransport Login Name/Account Name or account name of the account the triggered the Advanced Routing feature
USERPARAMETER1	string	Account type: E for Partner, I for Internal, N for Unspecified

XFB Transfer tracked objects

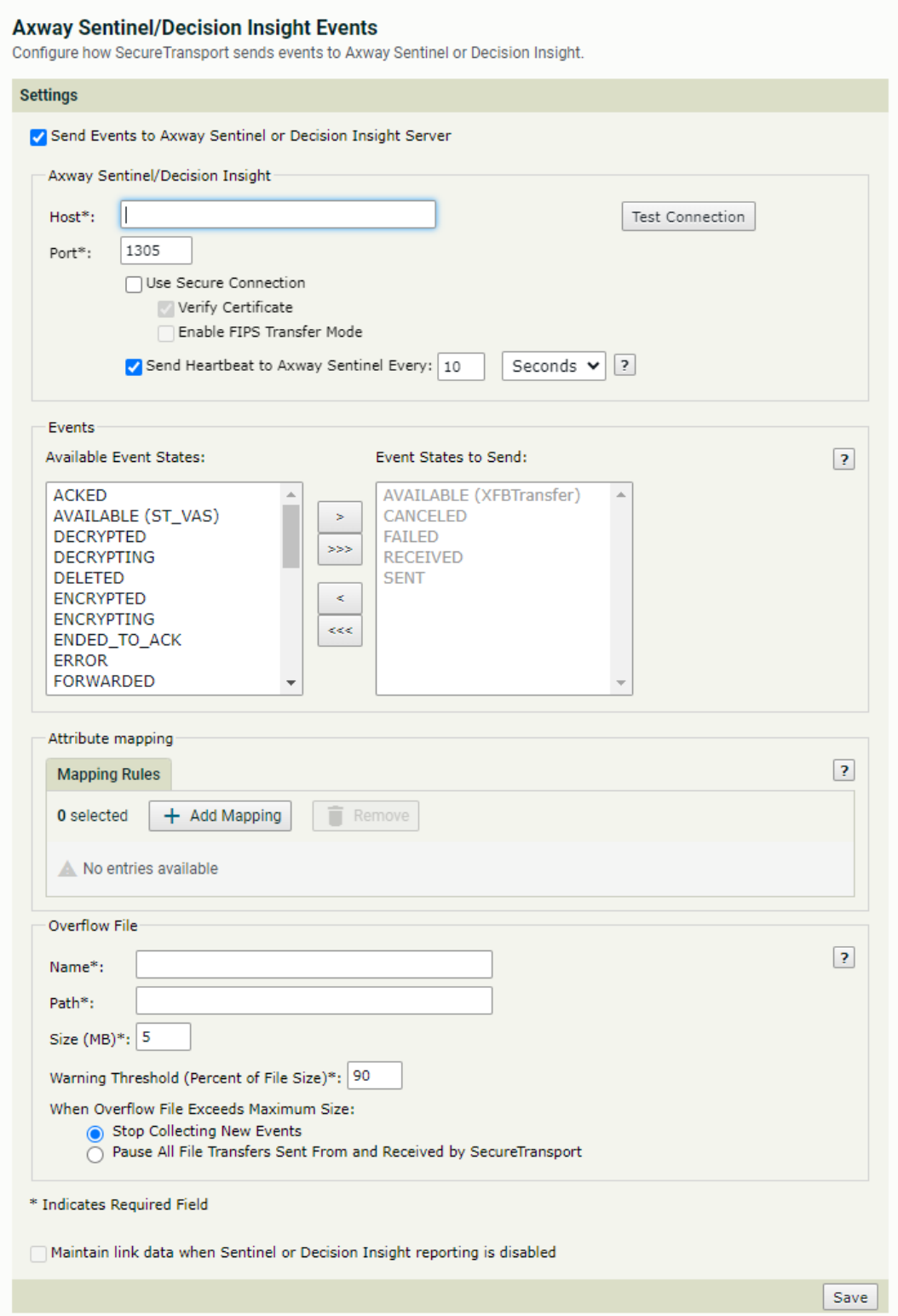
This topic provides the XFB tracked object roles, sender and receivers, production identification, and transfer attributes. For more information, refer to *XFBTransfer Tracked Object* on page 128.

Configure SecureTransport to send events to Decision Insight

Use the SecureTransport Administration Tool to configure SecureTransport to send events to Decision Insight.

The setting applies to all servers in your Enterprise Cluster (EC). Each server must have its own overflow file.

1. Select **Setup > Axway Sentinel/DI** to open the *Axway Sentinel/Decision Insights Events* page.



2. Select the checkbox for **Send Events to Axway Sentinel or Decision Insight Server**.

   The rest of the fields on the screen are enabled and SecureTransport sends events to Decision Insight as configured.

3. In the *Axway Sentinel/Decision Insight* pane, specify the FQDN or IP address of the Decision Insight server in the **Host** field and a valid TCP port on the server to which events will be sent in the **Port** field.
4. (Optional) Select the checkbox for **Use Secure Connection** to enable sending the selected event to Decision Insight over a secured connection.
5. (Optional) Select the checkbox for **Verify Certificate** to enable the SSL certificate verification. The **Verify Certificate** checkbox is selected by default.
6. (Optional) Select the checkbox for **Enable FIPS Transfer Mode** to enable sending events to Decision Insight over the secure connection in FIPS transfer mode.

   Note Changes to these settings in Step 3 through Step 6 take effect the next time you restart the Transaction Manager.

7. (Optional) Click the **Test Connection** button. This test indicates whether the port specified on the Decision Insight host accepts connections.
8. In the *Events* pane, select the event states to send to Decision Insight.

   An Event State specifies the current state of a file transfer. If an Event State is not selected to be sent, SecureTransport performs the processing represented by the state, but it does not send the event that reports the state to Decision Insight.

   Note The Event states that indicate transfer start and end must be enabled to build Decision Insight dashboards. Refer to *Event states* on page 174.

9. In the *Overflow file* pane, specify information about the file to be used to store SecureTransport event data when there is no connection between SecureTransport and Decision Insight. For more information, refer to *Configure Sentinel reporting* on page 123.
10. Click **Save**.

Server licenses

Use the *Server License* page to update SecureTransport licenses. You usually install licenses when you perform initial SecureTransport setup and configuration after installation. Update licenses whenever required, for example, when you receive your permanent license for SecureTransport after an evaluation period.

SecureTransport requires two licenses. The core server license specifies the number of accounts allowed and the number of ad hoc users allowed. The core license can limit the license to a specified host and to a specified date range. The features license specifies if the AS2, SSH, and Connect:Direct protocols are allowed, if SiteMinder integration is allowed, if the Enterprise Cluster (EC) option is included, and the number of cluster nodes allowed.

The FTP and HTTP protocols are included in the core license. For other features, contact your local account executive or supplier.

The following topics describe the account session count and ad hoc licenses and provide how-to instructions for updating the SecureTransport licenses:

Account session count

The number of accounts in the core server license controls the number of connections allowed to the server. An account license is considered in use when a user logs in. A license is also considered in use when used for a site that is initiating transfers. The license is considered in use for 60 days after the initial login or site transfer. The Folder Monitor, AS2 receiving, and asynchronous AS2 MDN receiving are excluded from license counting.

Account licenses apply to all protocols. To limit the number of concurrent users who can connect to the SecureTransport FTP and HTTP servers, see *User limits* on page 806.

Note SecureTransport does not perform DNS lookups. Therefore a single site referred to in one place by name and another place by IP address is counted as two sites.

Ad hoc user licenses

You must install a core license with ad hoc user licenses included to enable users to compose, send, reply to, or forward messages using ST Web Client. There are four categories of ad hoc user licenses:

- **Unlimited ad hoc user licenses**: If your company has purchased an unlimited number of ad hoc user licenses, then the display shows "unlimited" for the number of ad hoc Users.
- **One ad hoc user license for each account license**: If your company has purchased one ad hoc user license for each account license, then the display shows the same number of licenses for Accounts and for ad hoc users.
- **Fewer ad hoc user licenses than account licenses**: If your company has purchased fewer ad hoc user licenses than account licenses, then the display shows the maximum number of users that can compose, send, reply to, or forward messages using ST Web Client. One ad hoc user license is consumed the first time a user performs one of these actions.
- **No ad hoc user licenses**: If your company did not purchase any ad hoc user licenses, then end users cannot use ad hoc file transfers. The display does not include the line with ad hoc users.

Updating SecureTransport licenses

To obtain the text files that contains the server licenses, contact Axway Global Support. For contact information, see *Get more help* on page 27.

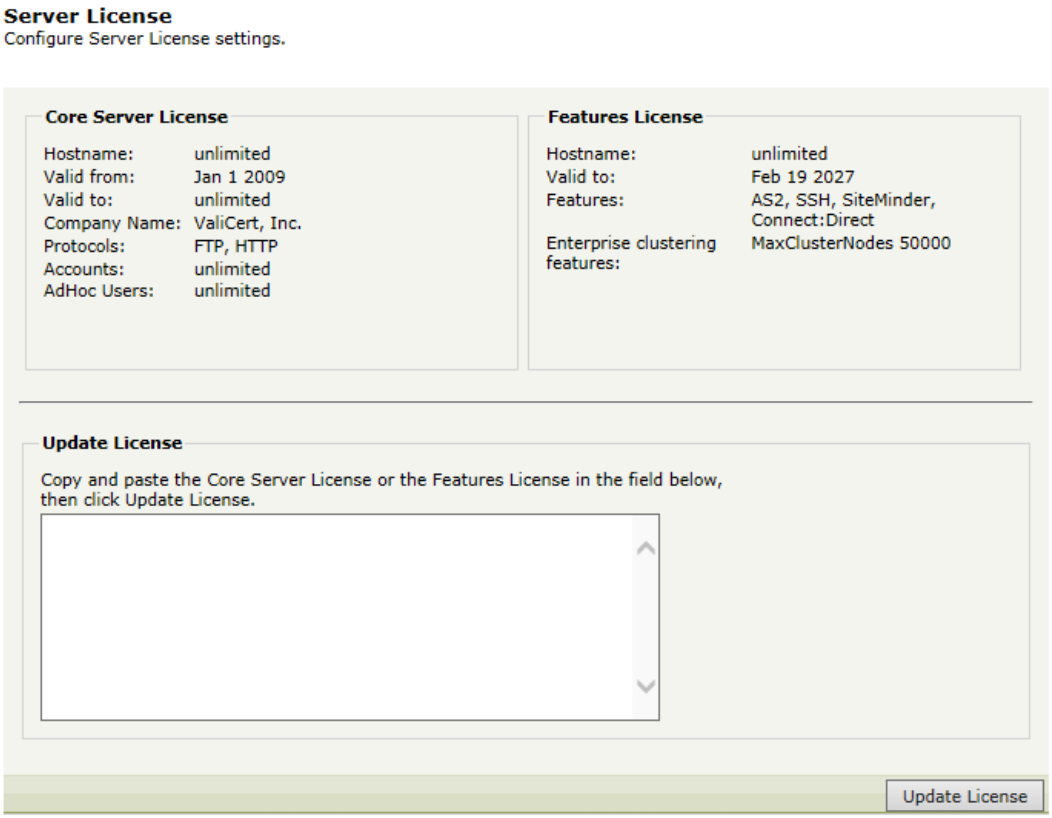
1. Select **Setup > Server License**.

   The *Server License* page is displayed.

2. Copy and paste the entire contents of the Core Server License into the **Update License** text area.
3. Click **Update License**.

   The updated license information is displayed on the *Server License* page.
4. Copy and paste the entire contents of the Features License into the **Update License** text area.
5. Click **Update License**.

   The updated license information is displayed on the *Server License* page.



Note The Connect:Direct license is listed only when the Connect:Direct protocol is enabled.

6. Restart all SecureTransport servers that are running.

   The licenses for your SecureTransport Server are updated.

Set up usage reporting in SecureTransport

Each SecureTransport instance is capable of recording and providing daily usage information through customizable reports. Each report includes both daily statistics and totals for the chosen reporting period, with metrics that can be adjusted to the intended purpose.
SecureTransport provides two report generation options: manual and automatic. The method used depends on the type of report:

- **Ad Hoc reports**
  These reports are created manually on an as-needed basis. They can be tailored for various needs, such as period-to-period comparisons, trend monitoring, or resource optimization. Their configuration is flexible, allowing you to choose the metrics and reporting periods that best fit your objectives.
- **Reports for subscription usage tracking**
  These reports are used by Axway to measure the subscription services you use on a monthly basis. They follow a strict configuration to ensure accurate billing and compliance with the Amplify platform. Their creation and delivery to the Amplify platform can be automated.

Report metrics and configuration

Usage is reported against environments in your organization. For SecureTransport, an environment corresponds to one cluster or one standalone installation. If you have multiple instances, usage reporting must be configured separately on each one. Report configuration is managed on the **Setup > Server License** page.

A usage report may include the following metrics:

Metric	Description	Always Present/Configurable
Count of active users	Active are considered the users who log in to SecureTransport.	Configurable via the *Include count of active users* checkbox To ensure your reports display accurate figures, you must enable data collection in advance by selecting the metric checkbox before the reporting period begins. A value of 0 means no data has been collected. For example, if you want to count active users for May, you must select its checkbox no later than April 30th.
Number of successful outbound file transfers	Includes client-initiated downloads, server-initiated pushes, and file resubmissions. Excludes transfers via Folder Monitor and internal routing (within ST) via Advanced Routing and Standard Router.	Always present, not configurable

Metric	Description	Always Present/Configurable
Number of inbound file transfers	This is the number of successful inbound file transfers with state *RECEIVED*. Internal routing (within SecureTransport) via Advanced Routing and Standard Router is excluded.	Configurable via the *Include inbound transfers* checkbox; Applies only to manually generated reports. It does not affect the automatic reporting.
Incoming file volume	The total file size, in bytes, of all successful inbound file and PeSIT message transfers.	Configurable via the *Include incoming file volume* checkbox

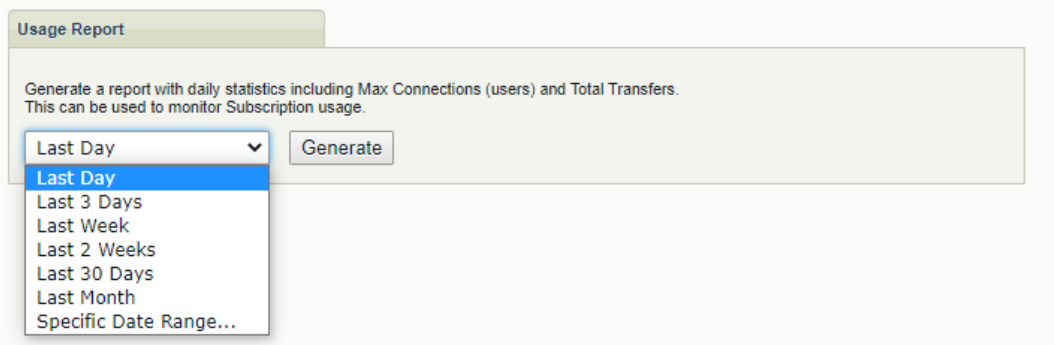
Manual report generation

Follow the instructions to create Ad Hoc reports and manual reports for subscription usage tracking:

1. On the *Server License* page, navigate to the *Usage Report* section.
2. In the *Report Configuration* panel, locate the **Environment ID** option.
    - For subscription usage reporting, this field must contain the Environment ID of the SecureTransport instance. Your organization administrator can manage your environments and obtain the IDs. See *Managing environments*.
    - For Ad Hoc reports, you can leave this field blank. In this case, SecureTransport will generate an ID for your environment.
3. Select the checkboxes for the metrics you add to the report. If you are creating a report for subscription usage tracking, leave the **Include inbound transfers** checkbox blank.
4. Under *Usage Report*, select a reporting period from the drop-down menu.

    Note The usage report includes data only up until the last fully completed day.

    For example, if you want to generate a report for the entire month of May, you should set the start date as May 1st and the end date as June 1st, so that all data from May 1st to May 31st is included.



5. Click **Generate**.
   The report is generated immediately based on the current settings specified at the moment on the *Server License* page, without the need to save those settings for future use. It is automatically downloaded to your default download location.

For instructions on uploading your subscription usage data to the Amplify Platform, see *Manual entry reporting*.

Automatic reports for subscription usage tracking

SecureTransport can automatically create a usage report once a day and deliver it to the Amplify Platform.

Prerequisites

Subscription usage reporting requires that you have an environment and a service account created for your organization in the Amplify Platform.

- For new customers, an organization with a default administrator account is generated. That account is typically an initial contact provided during procurement.
- For existing customers, an organization with a default admin account should already exist. In cases where an organization needs to be added or the default admin account needs to be changed, contact your Axway representative. Administrators can add new service accounts to an organization. Refer to *Amplify Subscription Usage* for more details.

Configuration

1. In the **Report Configuration** panel, fill in the following information:

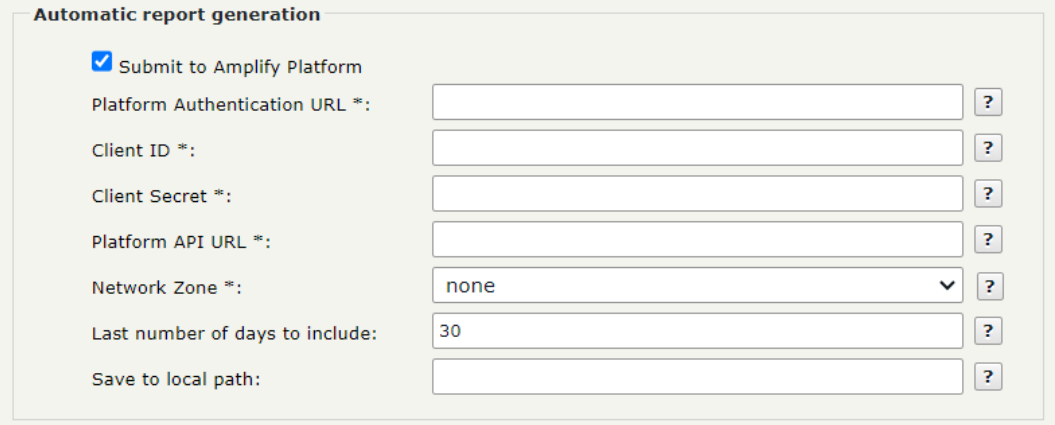
- **Environment ID** - the unique identifier of your product environment. It corresponds to one cluster or one standalone installation. Your organization administrator can manage your environments and obtain the IDs. See *Managing environments*.
- **Environment Name** - the name of your environment associated with the Environment ID.

- **Schema ID** - a fixed URL to the schema that validates the report: *https://platform.axway.com/schemas/report.json*

2. Select the checkboxes for the metrics you wish to add to the report.

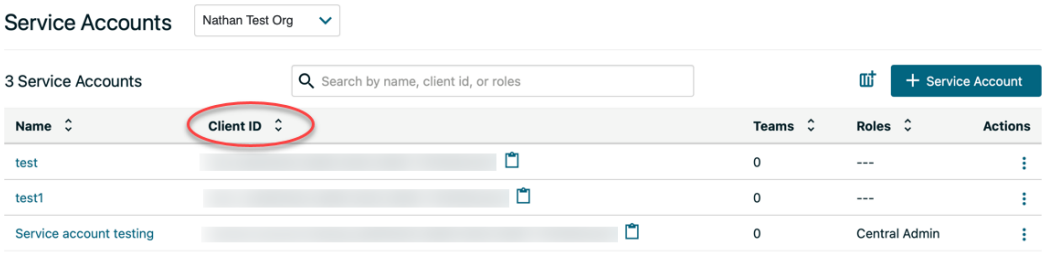
- **Include count of active users** - optional; on-premise customers do not need to report this metric.
- **Include incoming file volume** - optional; on-premise customers do not need to report this metric.
- **Include inbound transfers** - does not affect automatic reports.

3. Select the **Submit to Amplify Platform** checkbox to enable automatic delivery to Amplify:



4. Please enter your connection details and authentication information to access the Amplify platform:

   - **Platform Authentication URL** – a fixed authentication URL: *https://login.axway.com/auth/realms/Broker/protocol/openid-connect/token*
   - **Client ID**- the Client ID of your service account. To obtain this ID, the organization admin must log in to the Amplify Platform, navigate to **Organization**, and click the **Service Accounts** tab.



   - **Client Secret** – the Client Secret to authenticate the service account. It is either manually entered or generated by the system during the creation of the service account on the Amplify Platform.
   - **Platform API URL** – a URL of the API used to upload the report to the Amplify Platform: *https://platform.axway.com*
   - **Network Zone** – If a network zone is selected, SecureTransport uploads the report through the proxy of that network zone.

5. In the **Last number of days to include** field, specify the number of days that you want the report to include. The usage report includes data only up until the last fully completed day, i.e., current day's data is not included. For example, if you enter 2 and today is May 14th, the report will contain data for May 12th and 13th.

   The default reporting period is 30 days, resulting in a report containing 30 individual daily reports for each day plus a total. If there are thousands of transfers, generating this report might take a lot of time and use up significant resources.
6. (Optional) Set **Save to local path** to a local folder for saving the report.
7. Click **Save**.

Each field and setting for automatic report generation has a dedicated server configuration option that gets populated on the *Server License* page. These server configuration options are not editable, modifications to the automatic reporting settings must be made either from the *Server License* page or via the REST API `/statisticsSummary` resource.

Note You can manually generate reports without altering the saved automatic reporting configuration. To do this, adjust the report settings as needed and click **Generate**. Do not click the *Save* button. Instead, exit the page by using your browser's **Back** button.

View usage

For details on monitoring usage data for customer-managed on-premise Axway products that you use under subscription agreements, see *Viewing usage*.

Report structure and metrics

SecureTransport generates usage reports as *.json* files. Each report contains the following parameters:

Element *(type)*	Description
envId *(string)*	Unique identifier of your product environment, a cluster-wide parameter
schemaId *(string)*	The URL to the schema that validates the report
timestamp *(string)*	Timestamp of when the report was generated. This value is in ISO 8601 format: `{YYYY}-{MM}-{DD}T{hh}:{mm}:{ss}.{SSS}{TZ}`.
granularity *(number)*	Report interval granularity: *86400000* milliseconds (24 hours)

Element *(type)*	Description
report *(object)*	Collection of *daily reports*, chronologically ordered by date in the specified reporting period
date *(object)*	The date of the daily report. This value is in ISO 8601 format: `{YYYY}-{MM}-{DD}T{hh}:{mm}:{ss}.{SSS}{TZ}`. The number of daily reports depends on the selected reporting period.
product *(string)*	Name of the MFT product: *SecureTransport*
usage *(object)*	Collection of daily metrics related to SecureTransport. It contains the number of active users and information about the file transfers performed within the day.
ST.ActiveUsers *(number)*	The number of active users for that day. If **Include count of active users** is not selected, the metric is reported as `0`.
ST.TransfersOut *(number)*	Number of successful outbound file transfers for the day.
ST.TransfersIn *(number)*	Number of successful inbound file transfers with state RECEIVED for that day. This metric is present in the report only when the **Include inbound transfers** checkbox is selected.
ST.Volume *(number)*	Total file size, in bytes, of all successful inbound PeSIT message transfers and file transfers for the day. Each PeSIT message counts as an inbound transfer with a default size of 4096 bytes. If **Include incoming file volume** is selected, the metric is reported as `0`.
meta *(object)*	Additional meta information for the daily report
meta *(object)*	Additional meta information for the summary report

Element *(type)*	Description
companyName *(string)*	Company name as specified in the license
envName *(string)*	Custom display name of the environment, specified in the report configuration
productLine *(string)*	Name of the product line: *MFT*
productName *(string)*	Name of the product: *SecureTransport*
productVersion *(string)*	Base product version, e.g., *5.5* (the SecureTransport version)
currentPatch *(string)*	The currently installed update version, e.g., *UPDATE 5.5-20240425*
isECEnabled *(boolean)*	Specifies if *Enterprise Clustering* is enabled: *true* or *false*
isFMEnabled *(boolean)*	Specifies if *Flow Management* integration is enabled: *true* or *false*
isADIEnabled *(boolean)*	Specifies if *AxwayDecisionInsight (Sentinel)* integration is enabled: *true* or *false*
plugins *(object)*	List of installed plug-ins along with their versions
authorization *(object)*	List of installed *Pluggable authorization* on page 767
authentication *(object)*	List of installed *Pluggable authentication* on page 459
site *(object)*	List of installed transfer site plug-ins
customARStep *(object)*	List of installed Advanced Routing step plug-ins

Element *(type)*	Description
reportTimeframe *(object)*	The report time frame
startDate *(string)*	Timestamp of the start date of the reporting period
endDate *(string)*	Timestamp of the end date of the reporting period.
reportSummary *(object)*	Usage aggregations for the selected period based on daily *usage* on page 186 information
ST.ActiveUsers *(number)*	Total number of active users for the reporting period (within the last 60 days)
ST.TransfersOut *(number)*	Number of successful outbound file transfers for the reporting period.
ST.TransfersIn *(number)*	Number of successful inbound file transfers for the reporting period.
ST.Volume *(number)*	Total of the `ST.Volume` values for the reporting period

Configure FTP command log

The SecureTransport command logging feature works as a tracking system. It maintains a log of the commands entered by the users during an FTP session. Command logging is available for the FTP server only. You can use the `Ftp.CommandLogging.File` server configuration parameter to set the location of the command log. By default, the location is `<FILEDRIVEHOME>/var/logs/cmdlog`.

Use the *Command Logging* page to view and determine which user classes should have their commands logged. You can restrict this feature by user class, so that the FTP sessions of only certain user classes are logged.

The following topics provide how-to instructions for configuring and managing the FTP command log:

- *Add a command logging entry* on page 189 - Provides how-to instructions for adding a command logging entry.
- *Enable or disable command logging entries* on page 189 - Provides how-to instructions for enabling or disabling command logging entries.

- *Edit a command logging entry* on page 189 - Provides how-to instructions for editing a command log entry.
- *Delete command logging entries* on page 190 - Provides how-to instructions for deleting command logging entries.

Add a command logging entry

Use the following procedure to add a command logging entry.

1. Select **Setup > Command Logging** to open the *Command Logging* page.

2. Click **Add Logging**.
3. An entry is added to the *Command logging entries* table.
4. Select a **User Class** or * for all classes.

   On SecureTransport Server, the user class is validated. On SecureTransport Edge, the user class is not validated and if it does not exist, nothing is logged.
5. Click the Save icon ( ) in the **Edit** column.

   The status of the new entry is set to disabled.

Note To cancel an add operation, select **Setup > Command Logging** again.

Enable or disable command logging entries

Use the following procedure to enable or disable command logging entries.

1. Select **Setup > Command Logging** to open the *Command Logging* page.
2. Select the entries to enable or disable.
3. Click **Enable** or **Disable**.

   The icon in the **User Class** column changes to show that the entry is enabled or disabled.

If a Command Logging is added, but it is disabled; no Protocol Commands will be logged.

Edit a command logging entry

Use the following procedure to edit a command logging entry.

1. Select **Setup > Command Logging** to open the *Command Logging* page.
2. For the entry to edit, click the Edit icon ( ) in the **Edit** column.
3. Change the value in the **User Class** column as needed.
4. Click **Enable** or **Disable**.
5. Click the Save icon ( ) in the **Edit** column.

Note To cancel an edit operation, select **Setup > Command Logging** again.

Delete command logging entries

Use the following procedure to delete command logging entries.

1. Select **Setup > Command Logging** to open the *Command Logging* page.
2. Select the entries to delete.
3. Click **Delete**.

   The entry is removed from the *Command logging entries* table.

FTP SITE META command

SecureTransport has added support for a new command - SITE META. The command is part of the already existing SITE commands and provides capabilities to store user-specific information as metadata over the FTP protocol. It does not comply with any RFC documents and has a generic syntax.

The command accepts input in the format of key - value pairs. The supplied information is stored in the FTP session and is available until the session finishes or times out. For files uploaded during the same session, the provided information will be stored as file metadata attributes. The information can be evaluated at a later point for each file.

This way, the SITE META input could be used for defining routing rules for server-initiated transfers. For file metadata persistence specifics, see *Subscription flow attributes and FTP related attributes* on page 191.

Note To save the SITE META attributes for a file, the command execution and the file upload must happen within one session. Sharing attributes between different FTP sessions is not supported.

FTP SITE META command syntax

The command accepts arguments in the format of key-value pairs. Everything beyond the META will be considered as a command argument. The format follows the pattern:

```
SITE META <key>=<value>
```

The commands supports multiple arguments as well:

```
SITE META <key1>=<value1>;<key2>=<value2>;<key3>=<value3>
```

Supplying same key names with different values will result in overwriting the previous values with the latest one.

Note The "=" and ';' are considered special symbols:
- the semicolon sign (';') is used to delimit the different pairs,
- the equals sign ("=") splits the key name from its value.
If using them as part of the payload, escaping is required. They must be preceded by "\" or \\, depending on the manner how FTP clients accept command arguments.

When a file is uploaded using a common session, the commands arguments are stored as part of the file meta information. They are written in an existing namespace - the flow attributes user variables (*userVars*). For more information about flow attributes, see *Flow attributes* section in *Mail template commands and variables* on page 197.

The command supports entering keys with empty values:

```
SITE META <key1>=;<key2>=;
```

This will result in deleting existing flow attributes with the same key name (if any) for already existing files. For new files these attributes are discarded and will not be persisted.

Note The supplied key-value pairs are saved on the file system as metadata for each file. The information is stored in a serialized and readable format. We do not recommend storing any sensitive or confidential information.

Evaluate SITE META user metadata

FTP meta information can be evaluated and used to define basic transfer flows for server-initiated transfers. Evaluation of those attributes is possible from the Advanced Routing Steps, Transfer Sites and Subscriptions for fields that support expression evaluation (marked with a yellow stripe).

Within the Routing steps, metadata that comes from the SITE META command, along with any other file flow attributes can be evaluated using:

```
${flow.attributes['userVars.<key>']}
```

When using transfer sites and subscriptions, the file flow attributes (both SITE META and subscription), the expression is:

```
${stenv['DXAGENT_FLOW_USERVARS.<KEY-in-upper-case>']}
```

Subscription flow attributes and FTP related attributes

Key-value pairs from subscription properties and those from FTP SITE META share a common space and are all persisted as file attributes for each file (if any).

Uploading files in the subscription directory within an FTP session with SITE META commands, the subscription flow attribute settings will always take precedence.

For more information about flow attributes settings, see *Configure general settings* section in *Subscribe to Advanced Routing application* on page 888.

Configure transfer log

The transfer log tracks the file uploads and downloads on the system and records a lot of basic and additional information, such as whether the transfer was initiated by the server or by a user, protocol used and other information.

The tracking information is kept in the database and in a log file named `xferlog`, which is located in the `<FILEDRIVEHOME>/var/logs` directory. For information about the `xferlog` log file, see *General log files* on page 1077.

With Enterprise Cluster, you can store the transfer log data in a separate external database from the rest of the SecureTransport data. See *Direct log data to separate Oracle databases* on page 97.

Due to specifics in the Standard Cluster architecture, if the primary server goes down, a secondary server is promoted to primary and starts maintaining the transfer log information. Then if when the former primary server comes back and resumes its primary role, the transfer log information from the temporary primary server is not migrated to the new primary one. See *Consolidated log data representation on page 356*.

Use the *Transfer Logging* page to add and edit logging entries to determine which transfers will be logged. You can also enable this feature for a specific user class.

The following topics provide how-to instructions for managing the transfer log configuration:

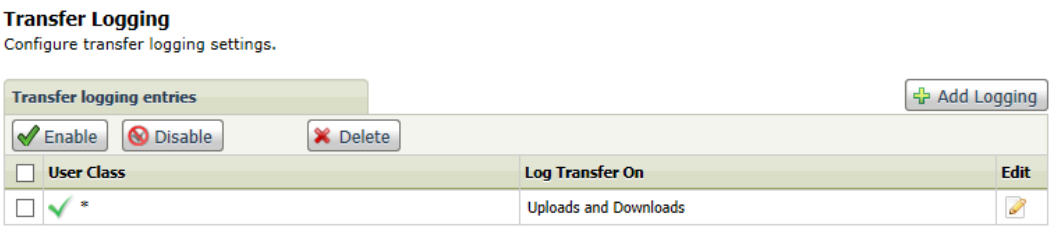
- *Add transfer logging entries* on page 192 - Provides how-to instructions for adding transfer logging entries.
- *Enable or disable transfer logging entries* on page 193 - Provides how-to instructions for enabling or disabling transfer logging entries.
- *Edit transfer logging entries* on page 193 - Provides how-to instructions for editing transfer logging entries.
- *Delete transfer logging entries* on page 194 - Provides how-to instructions for deleting transfer logging entries.

Add transfer logging entries

Use the following procedure to add transfer logging entries.

1. Select **Setup > Transfer Logging**.

   The *Transfer Logging* page is displayed.



2. Click **Add Logging**. A row is added to the *Transfer logging entries* list.
3. Select a **User Class**. The user class must already be defined in the *User Classes* page of the **Access** menu.

   Asterisk (*) means all users.
4. In the **Log Transfers On** list, select `Uploads`, `Downloads`, or `Uploads and Downloads`.
5. Click the Save icon ( ) in the **Edit** column.

   Your entry is added to the *Transfer logging entries*. By default, the status is disabled.

Note To cancel an add operation, select **Setup > Transfer Logging** again.

Enable or disable transfer logging entries

Use the following procedure to enable or disable transfer logging entries.

1. Select **Setup > Transfer Logging**.

   The *Transfer Logging* page is displayed.
2. Select the checkbox for each entry to modify.
3. Click **Enable** or **Disable**.

   The icons in the **User Class** column change to indicate the status of the entries.

Edit transfer logging entries

Use the following procedure to edit transfer logging entries.

1. Select **Setup > Transfer Logging**.

   The *Transfer Logging* page is displayed.
2. Click the Edit icon ( ) in the **Edit** column for the entry to edit.
3. Make the required changes to the fields in the entry.
4. Click the Save icon ( ) in the **Edit** column.

Note To cancel an edit operation, select **Setup > Transfer Logging** again.

Delete transfer logging entries

Use the following procedure to delete transfer logging entries.

1. Select **Setup > Transfer Logging**.

   The *Transfer Logging* page is displayed.

2. Select the entries to delete.
3. Click **Delete**.

   The entry is removed from the *Transfer logging entries* table.

Configure holiday schedule

You can specify holiday dates for the SecureTransport system and use them later when creating scheduled transfers or tasks. Having set up official holiday dates, you can take one of the following actions when creating a scheduled task or transfer:

- Ignore the holiday schedule – this is the default option
- Take no action if the scheduled date happens to be a holiday specified in the holiday schedule

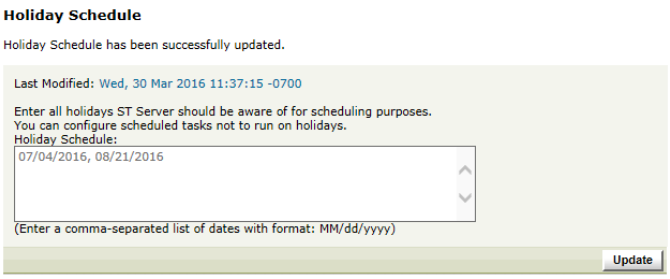
Keep the following items in mind when listing holiday dates:

- The list of holidays applies to all users on the server – it is not dependent on the time zones of users. The dates specified as scheduled are interpreted in local time by the server.
- The Holiday Schedule functionality does not allow for executing a scheduled task on the next working day if the specified date happens to be a holiday – when this occurs, the tasks are not executed.
- Weekend days are treated as holidays by default.

For information about creating scheduled transfers, see *Scheduled downloads and tasks* on page 674 and *Set up a scheduled transfer task for a subscription* on page 675.

1. Click **Setup > Holiday Schedule**.

   The *Holiday Schedule* page is displayed.



2. Enter one or more holiday dates in the **Holiday Schedule** field.

   Use the *MM/dd/yyyy* date format. If you enter more than one date, separate them with commas.

3. Click **Update**.

The configured Holiday Schedule applies to all users of the SecureTransport server and is applicable for all accounts and transfers.

Mail templates

Use the *Mail Template Repository* page to manage the email templates used in Human to Human and System to Human file transfers. SecureTransport uses the mail template selected on the *AdHoc Settings* page to create all notification emails to ad hoc file transfer recipients and senders. To select the mail template, see *Configure AdHoc file transfers on page 87*.

SecureTransport sends the following email notifications to ad hoc file transfer recipients and senders:

- Account enrollment notification to the recipient
- Package delivery notification to the recipient and the sender
- Account enrollment failure notification to the sender
- Package delivery failure notification to the sender

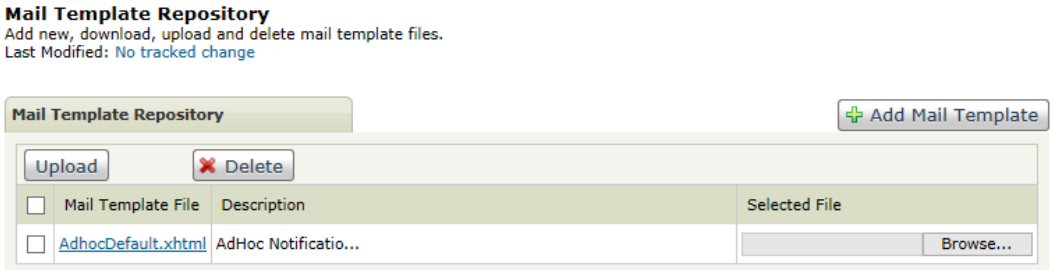
The mail template uses variables in `style` attributes to select the information that SecureTransport includes in each notification.

Add a mail template for AdHoc, Enrollment, or Advanced Routing notifications

Use the following procedure to add a mail template.

1. Click **Setup > Mail Templates**.

   The *Mail Template Repository* page is displayed.

   The default mail template is loaded in the repository initially.

2. Click **Add Mail Template**.

   A line is added to the list.

3. On the new line, click **Browse**.

   A file upload window is displayed.

4. Select a file template file to upload.

5. Click the Save icon ( ) in the **Selected File** column.

   SecureTransport uploads the mail template file and adds it to the repository.

Download a mail template

Use the following procedure to download a mail template.

1. Right-click the template name in the Mail Template File column and select **Save Link As** or **Save Target As**.

   The web browser displays a dialog box.
2. Navigate to the folder and change the file name as needed.
3. Click **Save**.

   The file is saved to the location you specify.

Upload an updated mail template

After you download a mail template and update it, you can upload the update to SecureTransport.

1. In the first column, select the mail templates for which you have updated files.
2. On each selected line, click **Browse** and selected the updated file.

   The file name of the updated template must be the same as the file name of the existing template.
3. Click **Upload**.

   SecureTransport uploads the updated files and replaces the existing files. Any references to the file templates are not changed.

Delete mail templates

Use the following procedure to delete mail templates.

1. In the first column, select the mail template files to delete.
2. Click **Delete**.

   SecureTransport displays a confirmation dialog.
3. Click **OK** to delete the selected mail template files.

See next: *Mail template commands and variables on page 197*

Mail template commands and variables

When you create a custom mail template, make sure to include the CSS in the `<style>` element in the default file, `AdhocDefault.xhtml`. You can download the default mail template, rename it, customize it and upload your custom mail template.

The following topics lists the notification type and information variables and provide an email template variable example:

Notification type variables

Reference the following variables in the `display` property of the `style` attribute of an XHTML element in the mail template to select what information is included:

- `$DISPLAY_ENROLLMENT` – Enrollment notification
- `$DISPLAY_DELIVERY` – Delivery success or failure notification
- `$DISPLAY_PKG_INFO` – Message information including the sender, the recipient, and the message text
- `$DXAGENT_CORE_ID` – File identifier (coreID) of the transferred file(s)
- `$DXAGENT_FORMATTED_END_EVENT_TIME` – Date and time of the event that triggered the notification. Timestamp Format: `MM/dd/yyyy HH:mm:ss.SSS`
- `$DISPLAY_FAILURE_ENROLLMENT` – Enrollment failed
- `$DISPLAY_FAILURE_FORBIDDEN` – Delivery was forbidden based on the delivery method
- `$DISPLAY_FAILURE_GENERAL` – General delivery failure
- `$DISPLAY_FAILURE_RECIPIENT` – Delivery to the recipient failed
- `$DISPLAY_FAILURE_UNRESOLVED` – The recipient could not be resolved

If the value of a variable is the empty string, the information is displayed. If the value is `none`, the information is not displayed.

Notification information variables

The following variables contain information used in account enrollment notifications:

- `$ENROLL_USERNAME` – User name for enrollment
- `$ENROLL_PASSWORD` – Password for enrollment

- $ENROLL_LOGINURL – URL for enrollment using $ENROLL_USERNAME and $ENROLL_PASSWORD

The following variables contain information used in package delivery failure notifications:

- $PKG_FAILURE_ENROLLMENT – Error message for an account enrollment failure
- $PKG_FAILURE_FORBIDDEN – Error message for a forbidden delivery failure
- $PKG_FAILURE_GENERAL – Error message for a general delivery failure
- $PKG_FAILURE_RECIPIENT – Error message when the email cannot be delivered to a recipient
- $PKG_FAILURE_UNRESOLVED – Error message when an account cannot be resolved
- $PKG_TO – Recipients from the TO list
- $PKG_CC – Recipients from the CC list
- $PKG_BODY – Text of the message
- $PKG_ATTACHMENTS – Names of the files attached to the message

Flow and subscription attributes

Flow and subscription attribute user variables (userVars) are defined per subscription in the *Flow/Subscription Attributes* pane. See *Subscribe an account to an application* on page 664.

To use userVars for email notifications, you must create your own mail template and specify the desired value with following expression:

```
$DXAGENT_SUBSCRIPTION_ATTR_USERVARS_{KEY}
```

Where KEY is the flow attribute key.

For example, if you want a user name defined in the Flow Attributes, you must type:

```
$DXAGENT_SUBSCRIPTION_ATTR_USERVARS_NAME
```

If the key of a flow attribute contains periods (.) in its name, you must replace all periods with underscores (_) when using this attribute in an email template.

For example:

To use userVars.name.first in an email template you must type:

```
$DXAGENT_SUBSCRIPTION_ATTR_USERVARS_NAME_FIRST
```

If there is a collision between Flow Attributes keys after they are used for email template notifications, the value to which they will be evaluated in the email template is not determined.

For example:

userVars.name_first and userVars.name.first are both used as:

```
$DXAGENT_SUBSCRIPTION_ATTR_USERVARS_NAME_FIRST
```

It is not clear to which value this expression is going to be evaluated.

Email template variable example

The following example from the default email template uses a variable to select a row of a table that includes information from another variable.

```
<tr style="display: $DISPLAY_DELIVERY">
   <th>To</th>
   <td style="vertical-align: top;">$PKG_DELIVERY_TO</td>
</tr>
```

Configure miscellaneous settings

Use the *Miscellaneous Configuration* page to specify the administrator's e-mail address, set usage monitor options, enable or disable reverse DNS lookups, set the session timeout limits, select a default HTML template, limit FTP login failures, set FTP and HTTP server startup password timeout configuration and suspension options, and define password policy.

The following topics describe and provide how-to instructions for managing the miscellaneous configuration setup options:

- *Miscellaneous options* on page 199 - Describes how to configure the miscellaneous setup options and provides how-to instructions for the miscellaneous setup options.
- *Set up email notifications via SMTP* on page 202 - Provides how-to instructions for configuring SMTP.
- *FTP and HTTP server suspend options* on page 203 - Provides the how-to instructions for configuring FTP and HTTP server suspension options.
- *Set password policy* on page 205 - Describes the password policy and provides how-to instructions for configuring the password policy.

Miscellaneous options

This topic describes how to configure the miscellaneous setup options.

Note As of SecureTransport 5.3.3, Java Applet and ActiveX are no longer available or supported.

The following topics provide how-to instructions for configuring the miscellaneous options:

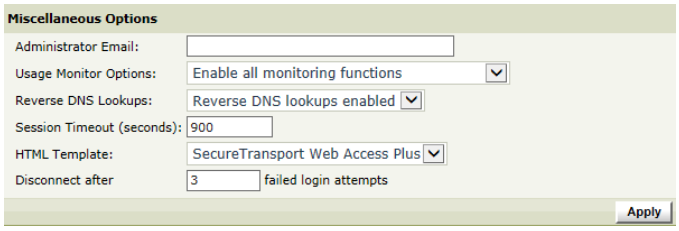
- *Set the administrator's email on page 200*
- *Set usage monitor options on page 200*
- *Enable or disable reverse DNS lookups on page 201*
- *Set the session timeout on page 201*

- *Select a default HTML template on page 201*
- *Limit FTP login failures on page 202*

Set the administrator's email

The administrator e-mail is the email address for the system administrator of the FTP server. This address (if specified) is used in several server response messages and is available (%E macro) for run-time messages. Administrator email is available for the FTP server only.

1. Select **Setup > Miscellaneous** and view the *Miscellaneous Options* pane.



2. Enter an e-mail address in the **Administrator Email** field.
3. Click **Apply**.

Set usage monitor options

The Server Usage Monitor can be configured to monitor several different aspects of the server or cluster nodes. You can also turn it off entirely.

Each of the monitoring options requires additional CPU resources per server process to compute and track the enabled measurements. To improve server performance, disable all unnecessary monitoring functions. The usage monitor is available for the FTP, SSH and HTTP(S) servers only.

The possible option settings are as follows:

Option	Description
Enable all monitoring functions	Enables all monitoring functions
Enable monitor - Measure bandwidth	Keeps track of the instantaneous transfer rate of each FTP server process running
Enable monitor - Display user commands	Keeps track of which FTP command a user is currently executing
Enable monitor - No bandwidth/commands	Displays process information per FTP server connection
Disable monitor	Disables all FTP, SSH and HTTP(S) monitoring

1. Select **Setup > Miscellaneous** and view the *Miscellaneous Options* pane.
2. In the **Usage Monitor Options** field, select a monitoring option or disable monitoring.
3. Click **Apply**.

Enable or disable reverse DNS lookups

Note This procedure has been replaced by the `Server.ReverseDNSLookups` server configuration parameter. Edit the `Server.ReverseDNSLookups` server configuration parameter to enable or disable reverse DNS lookups.

Reverse DNS lookups are used to resolve an IP address into a fully qualified domain name. The domain name is used for logging purposes and for applying access rules that specify a host name (instead of an IP address).

In cases where the DNS server is under heavy load, this can significantly affect the startup time of an FTP, HTTP, or SSH session. If the fully qualified domain name is not needed in the log files, it is best to turn off this feature.

Note This feature is available for the FTP, HTTP, and SSH servers.

1. Select **Setup > Miscellaneous** and view the *Miscellaneous Options* pane.
2. In the **Reverse DNS Lookups** field, choose to enable or disable reverse DNS lookups.
3. Click **Apply**.

To disable reverse DNS lookups for the Administration Tool server, see *DNS settings on page 1060*.

Set the session timeout

You can set a session timeout for SecureTransport so that users are automatically logged out if they are inactive for the specified duration. SecureTransport uses separate values for the session timeout for the SecureTransport Edge and SecureTransport Server. For example, when logged into the SecureTransport Edge, the Edge session timeout value is applied and when logged into the SecureTransport Server, the Server value is applied.

1. Select **Setup > Miscellaneous** and view the *Miscellaneous Options* pane.
2. Enter the number of seconds (60 seconds minimum) in the **Session Timeout** field to specify the session timeout duration.
3. Click **Apply**.

Select a default HTML template

SecureTransport comes with a built-in HTML template for the ST Web Client user interface which can be customized via the *Branding Tool*.

1. Select **Setup > Miscellaneous** and view the *Miscellaneous Options* pane.
2. Select a template from the **HTML Template** dropdown menu. The menu displays all templates that are stored in the following location:
   `<FILEDRIVEHOME>/share/ftdocs/html/skin`.
3. Click **Apply**.

Note The login page of the HTML template you specify here is displayed when the user first accesses this SecureTransport Server. The rest of template is used if the user is configured to use the default HTML template. To set the HTML template for business unit, see *Create or edit a business unit*. To set the HTML template for an individual user, see *Create a user account on page 503*.

Limit FTP login failures

You can limit the number of consecutive failed login attempts before the FTP Server terminates a user connection. Limiting the number of consecutive failed login attempts provides added security.

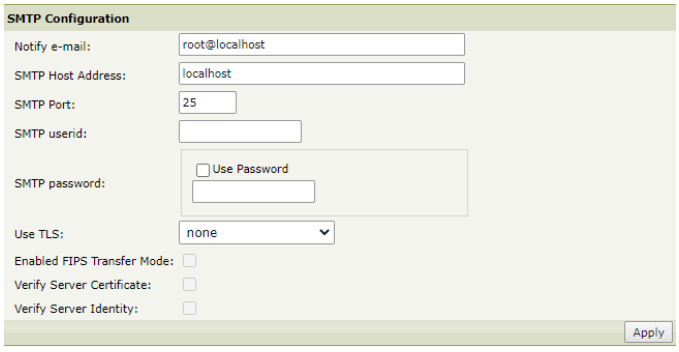
Note This limit is applicable only to FTP login attempts. The HTTP protocol by definition allows only one login attempt per connection.

1. Select **Setup > Miscellaneous** and view the *Miscellaneous Options* pane.
2. In the **Disconnect after _____ failed login attempts field**, type a value greater than zero to identify the number of failed login attempts you want to allow before disconnecting the user.
3. Click **Apply**.

Set up email notifications via SMTP

SecureTransport can be configured to send email notifications on various occasions. The SMTP server settings described below are used for sending email notifications for all SecureTransport features. Note that the approved senders list must also be defined.

1. Go to **Setup > Miscellaneous**.
2. Navigate to the *SMTP Configuration* pane.



3. In the **Notify e-mail** field, enter the email address where notification emails will be sent.
4. Enter the information SecureTransport needs to access your SMTP server:

- the host address of the SMTP email server
- the port used to access the SMTP server. Typically, port 25 is used for connections in plain text, port 465 is used for TLS connections, while port 587 uses STARTTLS.
- the user ID required to access the SMTP server.

5. If the SMTP server requires basic authentication, select **Use Password** and enter the password to authenticate with.
6. From the **Use TLS** drop-down, specify the connection security type required by the SMTP server:

- **none** - TLS is not required; the connection is not secure. This is the default option on upgraded instances and fresh installations of SecureTransport 5.5 up to Update 5.5-20220728.
- **Implicit TLS** (SMTPS) - TLS is negotiated immediately at connection start.
- **StartTLS (optional)** – An insecure connection will be upgraded to an encrypted one if possible. Otherwise, an insecure connection is used.
- **StartTLS (required)** - Enforces upgrading connections to use TLS. If anything fails in the process, the SMTP sending will fail. **STARTTLS (required)** is the default option on fresh installations of SecureTransport 5.5-20220728 or later.

7. Configure the level of security by specifying a TLS protocol version and allowed cipher suites:

- You can configure the acceptable TLS versions via the `Server.Smtp.Ssl.Protocols` configuration option.
- You can modify the list of allowed cipher suites via the `Server.Smtp.Ssl.Ciphers` configuration option.
- You can enable FIPS mode and modify the list of allowed FIPS cipher suites via the `Server.Smtp.Ssl.FIPS.Ciphers` configuration option.

8. Select the **Verify Server Certificate** checkbox to enable certificate verification. In this case, you need to have the SMTP server's certificate added to SecureTransport's list of trusted CAs*Import a trusted CA certificate* on page 56
9. Select the **Verify Server Identity** checkbox to enable identity verification. If selected, SecureTransport checks the server hostname against the server's identity as presented in the server *Certificate* message. The certificate that the SMTP server uses must have the *Subject Alternative Name* field present and configured correctly. Otherwise, the connection will fail.
10. Click **Apply**.

Note These settings can be overridden on a per-template basis by using the custom fields described here.

FTP and HTTP server suspend options

SecureTransport provides following two options for suspending the FTP and HTTP servers:

- Specify a schedule to suspend the server.
- Suspend the server immediately.

The server suspend option only applies to the HTTP and FTP servers.

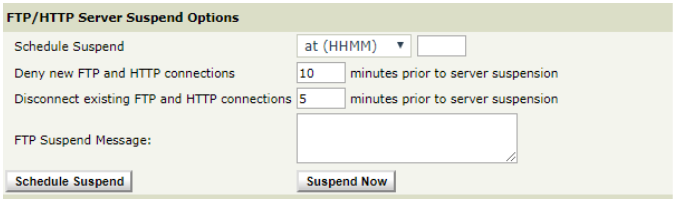
The following topics provide how-to instructions for scheduling server suspensions and immediately suspending the server:

- *Schedule server suspensions* on page 204
- *Suspend now* on page 205

Schedule server suspensions

Use the server suspension options to schedule a specific time of day for the server to suspend and to define a period of time in minutes before the scheduled suspension during which new FTP and HTTP connections are no longer accepted and existing FTP and HTTP connections are disconnected.

1. Select **Setup > Miscellaneous** to and view the *FTP/HTTP Server Suspend Options* pane.



2. Select a format for the time of suspension and type the time for its occurrence. The following time format options are available:
   - **at (HHMM)** – Schedule suspension at exact specified time.

     When defining an exact time, type the time in the HHMM format (a 2-digit hour followed by 2-digit minutes, based on a 24-hour clock) with no spaces or separators.

     If you set a scheduled time to start suspension before the current system time, SecureTransport schedules the suspension for the next day. For example, if the current time is 1400 and a HHMM suspension time of 1330 is specified, the server is scheduled to suspend at 1:30 pm. on the next day.
   - **in (minutes)** – Schedule suspension after the specified number of minutes.
   - **in (hours)** – Schedule suspension after the specified number of hours
3. Type the time, in minutes, when the server must refuse new FTP or HTTP connections before suspension.
4. Type the time, in minutes, when the server must disconnect existing FTP or HTTP connections before suspension.
5. Enter a short text message to be displayed to FTP users before suspension.

   Users might or might not see the suspension message based on how the FTP client settings are configured.
6. Click **Schedule Suspend**.

   The page now displays information about the scheduled suspension. To cancel or modify the scheduled suspension, click **Reschedule Suspend**.

Suspend now

This option allows you to suspend the server immediately. When the server suspends in this manner, it automatically disconnects all FTP users and disables any connections. FTP users cannot make any further requests. Users are not issued a warning with this option. The Administration Tool server continues to run.

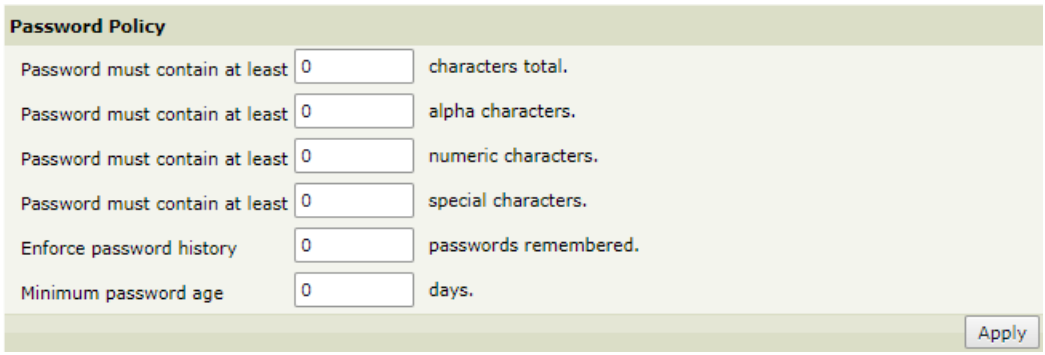
1. Select **Setup > Miscellaneous** to and view the *FTP/HTTP Server Suspend Options* pane.
2. Click **Suspend Now**.

Set password policy

As an administrator, you can define a set of requirements for users to comply with when they change their passwords. In SecureTransport, the password policy includes settings that control the complexity, lifetime, and reuse of passwords.

To configure a password policy:

1. Select **Setup > Miscellaneous**.
2. Navigate the *Password Policy* section.



3. Set password complexity requirements. These requirements are global and apply to all SecureTransport user and administrator accounts. The default value of 0 means there is no restriction.

Field	Description
Password must contain at least [*n*] characters total	The minimum allowed password length.
Password must contain at least [*n*] alpha characters	The minimum number of letters required in the password.

Field	Description
Password must contain at least[*n*] numeric characters	The minimum number of numeric characters required in the password.
Password must contain at least [*n*] special characters	The minimum number of special characters (anything other than letters or numeric characters) required in the password.

4. Set lifetime and reuse of passwords.
These settings are ignored when an administrator performs a password change against another user or admin account, but they do apply when changing their own password.

Field	Description
Enforce password history [*n*] passwords remembered	The number of the last *n* stored passwords which are restricted for reuse. Possible values: between 0 and 50. If the value is set to 0, the user can reuse their old password. **Note** SecureTransport stores the last 50 passwords of an user regardless of the value set in the *Enforce password history* field.
Minimum password age [*n*] days	The minimum number of days a password must be in use before it can be changed. Users can change their password only after the configured period expires. Possible values: positive integers If the value is set to 0, then users can change their passwords as often as they like. This setting is ignored when a user or an administrator is forced to change an expired password.

5. Click **Apply**.

Bandwidth limits

Note Bandwidth limits apply to transfers over the following protocols: HTTP, FTP and SSH. These limits do not affect PeSIT and AS2 transfers.

You can set the global Bandwidth Limits on this panel. You set bandwidth limit for inbound and outbound transfers in kilobytes per second per user account. Apart from global bandwidth limits, you can apply a hierarchy of bandwidth limits on business unit, account template and individual account level.

The hierarchy works in the following way: *global > business unit > account*. Account is on the same level as account template but note that limits set on the account level override global and business unit configurations.

When you add a zero for either limit, no bandwidth limits are applied.

ICAP settings

The Internet Content Adaptation Protocol (ICAP) settings allow the administrator to configure ICAP engines to be used as part of the SecureTransport file transfer processes so that data loss prevention (DLP) is achieved and anti-virus (AV) scans are completed. SecureTransport allows the administrator to use the ICAP connector to set up a SecureTransport server to scan (with an external DLP engine) files and AdHoc messages when delivering them to the recipient folder or mailbox. An ICAP server scan is run before a file is delivered.

Prior to configuring ICAP scanning, verify that the `ICAPScan` rule package is enabled. See *Manage rule packages* on page 223.

Note The SecureTransport administrator can edit the entire DLP/AV ICAP URL in the following format `icap://dlpav-address:port/servicename`. Both the Symantec anti-virus AVSCAN and AVSCANREQ are supported, though AVSCANREQ is preferred.

Note SecureTransport will scan received AdHoc messages and attachments when recipients open a message or download an attachment.
- An AdHoc message, identified as blocked by the DLP policy, will be displayed but the content will be changed to a notification stating that you are not allowed to view this message because it was blocked by the DLP policy. Subjects of messages remain unchanged.
- When downloading message attachments, identified as blocked by the DLP policy, they will be successfully downloaded but the content will be changed to a notification stating that you are not allowed to view the file because it is blocked by the DLP policy. This applies for all file types. File extensions will not be changed.

The ICAP servers provide

- Incoming and outgoing ICAP scanning for all file and message transfers
- Scanning policy support
- ICAP headers reporting: `X-Authenticated-User`, `X-Client-IP`, `X-Server-IP`

Note X-Server-Icap header reports the SecureTransport local IP address with each scanning request. If multiple network interfaces are available on the machine, the reported IP may not match the actual one.

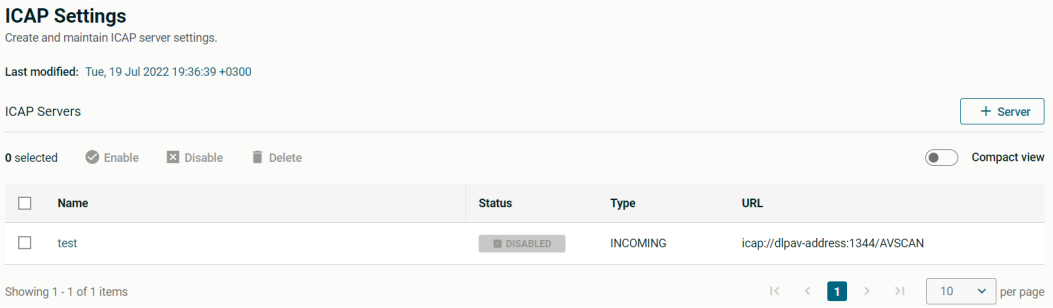
- Custom HTTP headers reporting
- Certain Expression Language variables that can be used in an Advanced Routing configuration

Setup of ICAP servers

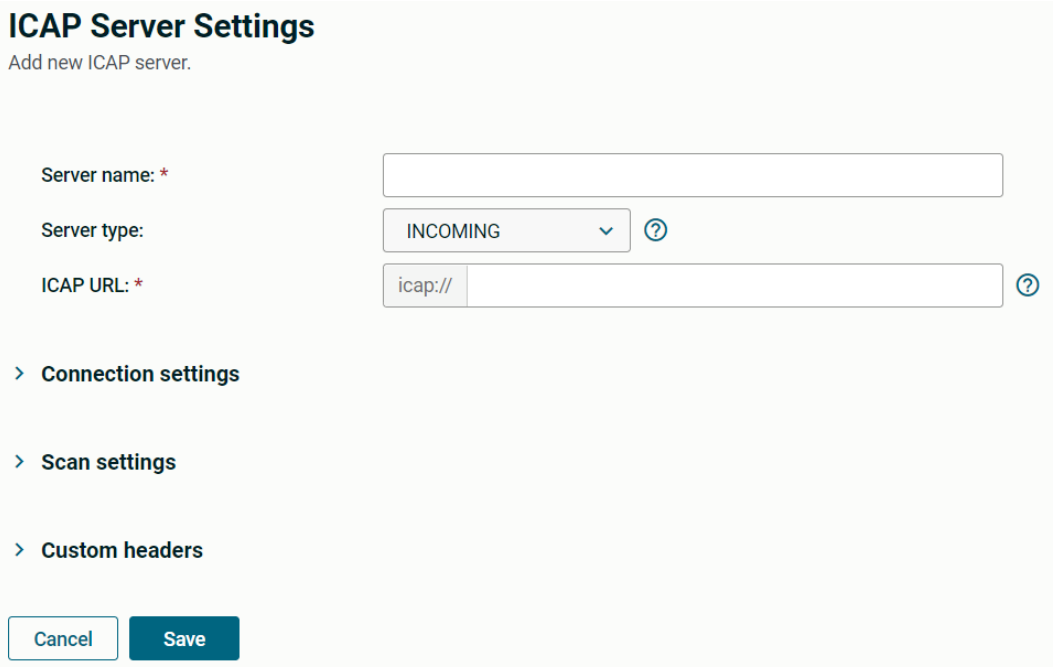
Multiple ICAP servers can be configured. Scanning is performed only by the ICAP servers that are enabled. There is no prioritization – all enabled servers are used for scanning files and messages. If a server along the chain returns a negative result from scanning, the transfer will be denied.

Navigate to **Setup** > **ICAP Settings**. The *ICAP Servers* page presents a list of ICAP servers with basic management controls:

- To add a new ICAP server, click the **+ Server** button.
- To enable or disable a server, select it and click the **Enable** or **Disable** button.
- To edit a server, click on its name.
- To delete a server, select it and click the **Delete** button.



Click **+ Server** to add a new ICAP server. The *ICAP Server Settings* page is displayed.



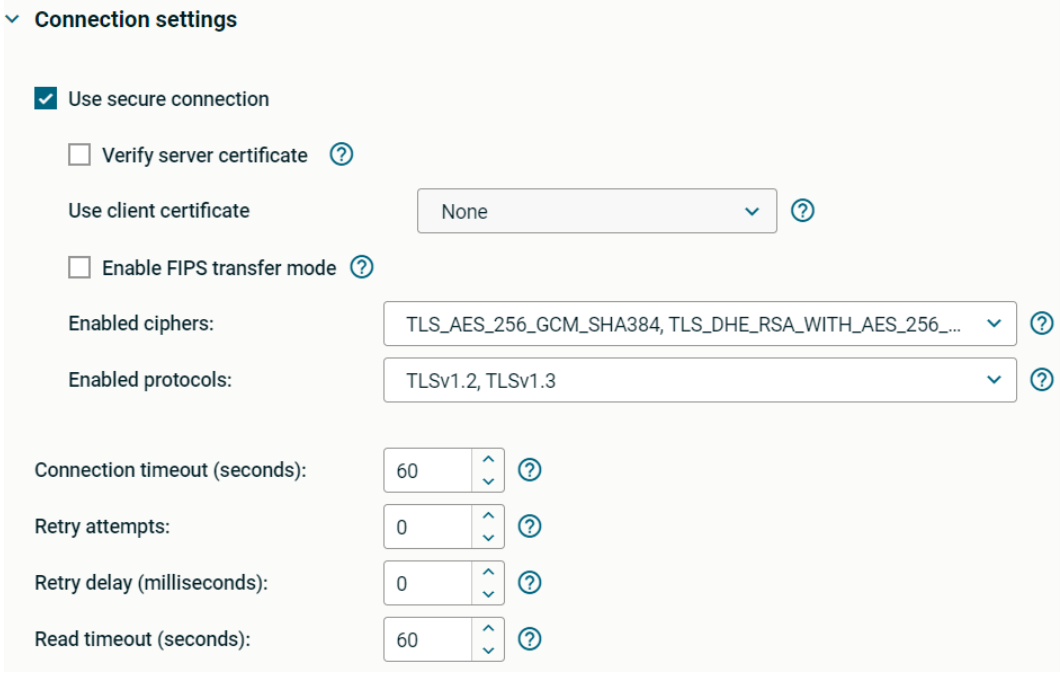
ICAP server configuration

Use the following procedure to configure an ICAP server:

1. Enter the **ICAP server name**. It must be unique; there cannot be two ICAP servers with the same name.
2. Enter the **ICAP server type**. It can be INCOMING, OUTGOING or BOTH.
   - **INCOMING** means that scanning will be performed by this ICAP server for all incoming transfers: File upload, AdHoc message creation, Server-initiated pull (for example from a Transfer Site)
   - **OUTGOING** means that scanning will be performed by this ICAP server for all outgoing transfers: File download, Reading of an AdHoc message, Server-initiated push (for example in the Advanced Routing step: Send to Partner or Publish to Account)
   - **BOTH** means that scanning will be performed by this ICAP server for all types of transfers
3. Enter the **ICAP URL**. Enter the DLP/AV ICAP URL in the following format:
   `icap://dlpav-address:port/servicename`
   The `servicename` can be the same as the mode of operation - `REQMOD` or `RESPMOD`, or something custom and vendor-specific.
   For the exact `servicename`, refer to the Data Loss Prevention (DLP) or Anti-virus (AV) documentation.
   If the default ICAP port (1344) is used, leave the port blank - it will be auto-populated.
   Examples:
   `icap://dlpav-address:1344/AVSCAN`
   `icap://dlpav-address:1344/REQMOD`
   `icap://dlpav-address:1344/RESPMOD`
4. Configure the *Connection settings* on page 209, *Scan settings* on page 210, or *Custom headers* on page 212.
5. Click **Save**.

Connection settings

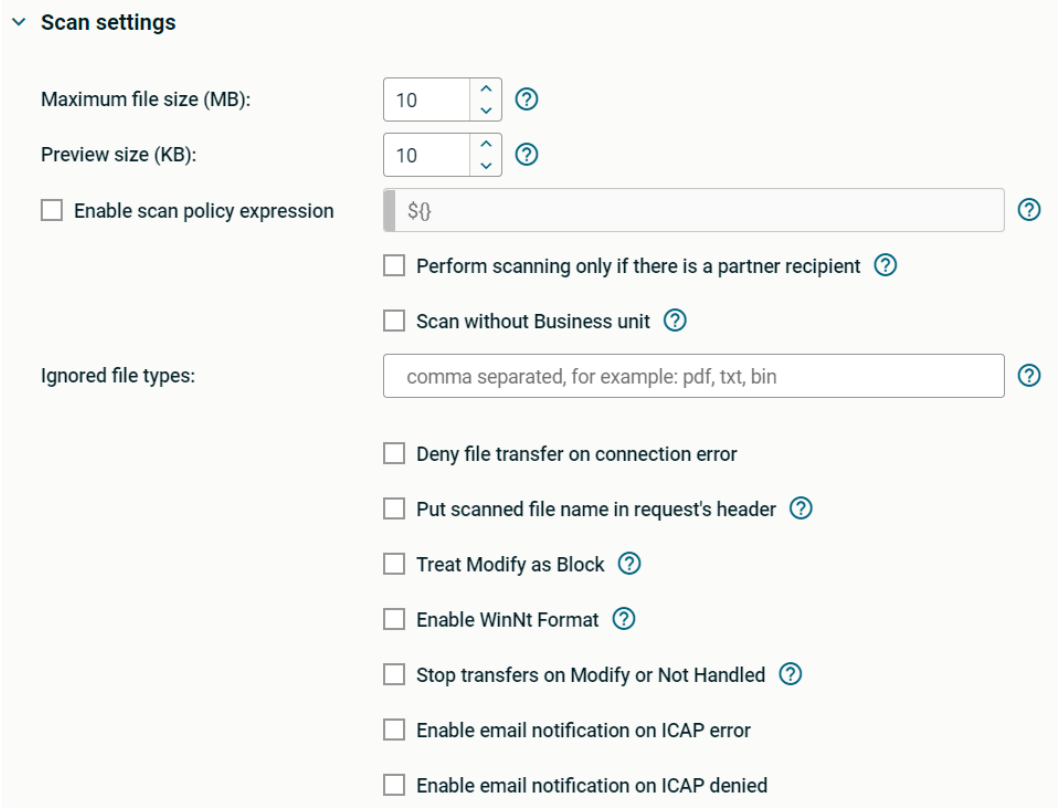
Follow the steps below to configure the ICAP server's connection settings.



1. Enable **Use secure connection** to use TLS secure connection to the ICAP server. Additionally, you can:
    - Enable **Verify server certificate** to use certificate verification to secure the connection to the ICAP server. When selected, SecureTransport verifies whether the server certificate is chained to a trusted root certificate imported in the Trusted CAs store.
    - Select **Use client certificate** to use a certificate from the Local Certificates store to authenticate to the ICAP server or a reverse proxy. To enable mutual authentication, select a certificate from this list and enable the **Verify server certificate** option.
    - **Enable FIPS Transfer Mode** to use only cipher suites certified in accordance with the Federal Information Processing Standard (FIPS).
    - Select **Enabled Ciphers** from the drop-down list of ciphers to be used for a TLS/SSL connection. The list contains only strong cipher suites - the ones marked with Default "Yes" in the list of supported ciphers and cipher suites in the Security Guide (login required).
    - Select **Enabled Protocols** from the drop-down list of TLS/SSL protocols.
2. Set the **Connection timeout**. This is the maximum connection timeout in seconds that the server will wait until it stops trying to reconnect.
3. Specify the retry policy:
    - **Retry attempts**. This is the number of retries in case of connectivity timeout.
    - **Retry delay**. This is the delay in milliseconds between each retry.
4. Set the **Read timeout**. This is the maximum read timeout in seconds.

Scan settings

Follow the steps below to configure the ICAP server's scan settings.



1. Enter **Maximum file size (MB)**.
   The default maximum file size is 10 MB. If the actual file size is larger than the maximum file size, SecureTransport will send up to the maximum configured file size to the ICAP server.
2. Enter **Preview Size (KB)**.
   The default preview size is 10 KB. If the ICAP server requires more data, SecureTransport will send it up to the maximum configured file size.
3. Specify the scan policy:
   - **Scan Policy Expression** if you want to perform scanning only under specific circumstances.
     When you select the **Scan Policy Expression** checkbox, the text box field allows you to use SecureTransport Expression Language. If both settings are disabled, scanning will always be performed. Sample usage - do not scan if the transfer is taking place over SSH protocol: `${session.protocol ne 'ssh'}`.

     Refer to the *ICAP scan policy expression language* on page 213 subtopic for the complete list of available expressions.
   - Select **Perform scanning only if there is a partner recipient** to enable or disable ICAP scanning for AdHoc messages if at least one of the recipients is external. User type - *internal* or *external*- is controlled by the account setting **Account Type**. Possible values are *Internal* - internal accounts, and *Partner* - external accounts.

     If the type of a recipient cannot be identified or is set to *Unspecified*, the account will be considered *external*. If both filtering settings are enabled, this particular setting will be applied over AdHoc messages.

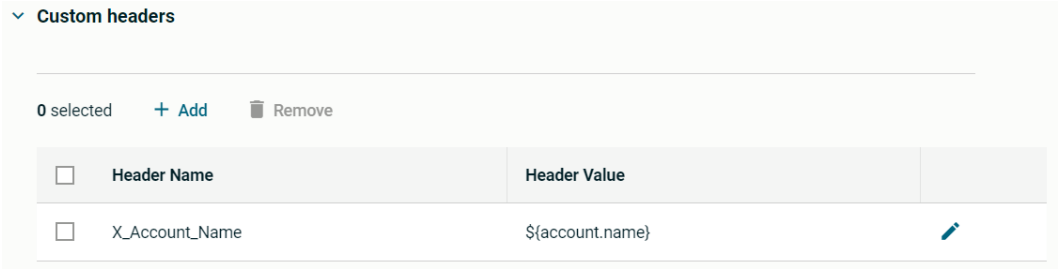
- Select **Scan without Business unit** to choose whether or not to enable ICAP scanning for accounts with no Business unit assigned.

4. Define **Ignored File Types**. Enter a list of file extensions, separated by comma. Files with these extensions will not be scanned.
5. Select **Deny file transfer on connection error** to deny file transfers upon a connection error to the ICAP server.
6. Select **Put scanned file name in request's header** to have the name of the file to be scanned included in the HTTP request line and filename header. For example, the request might look like: "*GET http://base_url/filename.example HTTP/1.1*".
   If the checkbox is not selected, the request will contain the static string "*GET /resource HTTP/1.1*".
7. Select **Treat Modify as Block** to choose whether or not to treat the ICAP MODIFIED action as block.
8. Select **Enable WinNt Format**. With this setting you can choose whether or not to report X-Authenticated-User in WinNT format in case of LDAP authentication.
   - X-Authenticated-User
     X-Authenticated-User is reported with each LDAP request. The header is reported differently depending on user type. Below are the supported X-Authenticated-User formats:
     - User with a local account and a locally stored password: Local://<account name>
     - Real OS user: Local://<login name>
     - Non-LDAP user mapped to a template (SiteMinder or SSO): Local://<login name>
     - LDAP user options:
       - If the WinNT format is not enabled for the server: LDAP://<LDAP domain name>/<user DN>*
       - If the WinNT format is enabled for the server: WinNT://<LDAP domain name>/<login name>
9. Select **Stop Transfers on Modify or Not Handled** to choose whether or not to stop the transfer if the ICAP server returns a MODIFY result or an unhandled status.
10. Configure email notifications:
    - Select **Enable e-mail notifications on ICAP error** - notification emails will be sent when there is a connection failure to the ICAP server.
    - Select **Enable e-mail notifications on ICAP denied** - notification emails will be sent when there is a deny by the ICAP server.

Custom headers

Specify any additional custom headers that must be passed to the ICAP server when making requests, along with their values. The **Header value** fields can either have a static value or a SecureTransport expression-based one. Expressions allow you to dynamically set a value, based on specific context, by

using the SecureTransport session or environment variables.



By default there are no custom headers configured, but you can add any number of headers by clicking the **+ Add** button. To remove a header, select it and click **Remove**.
If a header value is not present or can't be resolved, the header will be added with an empty or null value when sending the request.
Example:

- Header Name: `X_Account_Name`
- Header Value: `${account.name}`
- If a user with the name *user1* has logged in and the ICAP scan is performed, `Header:Value` will be evaluated to `X-Account-Name = user1` and it will be reported to the ICAP server (s).

ICAP scan policy expression language

This topic provides the expression language and variables available with the ICAP **Scan Policy Expression** option as part of the **ICAP Scan settings**.

The samples are distributed in dedicated subtopics, as follows:

- *Transfer-related expressions* on page 214
- *Session-related expressions* on page 214
- *LDAP-related* on page 215
- *HTTP-related expressions* on page 215
- *Flow attributes expressions* on page 215
- *Account-related expressions* on page 216
- *User-related expressions* on page 217
- *Business Unit-related expressions* on page 217

Note All the environment variables that are styled in *`italics`* depend on user input and the values shown in the tables are only samples. For more detailed examples of expression language usage, see *Custom Expression Language functions and variables* on page 1003.

Transfer-related expressions

Transfer-related	
Expression	**Possible/Sample Values**
transfer.targetDirFull	/stusers/sthome/acc1/ The path to the transferred file current directory
transfer.transferredBytes	10 The amount of bytes transferred
transfer.startTime	1520951365644 The difference, measured in milliseconds, between the time the transfer has started and midnight, January 1,1970 UTC
transfer.endTime	1520951366212 The difference, measured in milliseconds, between the time the transfer has ended and midnight, January 1, 1970 UTC
transfer.xferType	'A' – stands for ASCII 'I' – stands for Binary
transfer.targetDir	/ The root directory of the TARGETPATH

Session-related expressions

Session-related	
Expression	**Possible/Sample Values**
session.protocol	HTTP, FTP, SSH, Routing, AS2, PESIT
session.remoteAddress	10.134.12.224 The IP address of the machine from which the transfer has been initiated
session.remoteHost	10.232.15.109 The IP of the ST server that performed the scanning
session.streamingClient	Server, HTTPD, FTPD, SSHD

Session-related	
session.isSSL	0,1
session.siteProtocol	AS2, FTP, HTTP, SSH, PeSIT, FM, SystemToHuman

LDAP-related

LDAP-related	
Expression	**Possible/Sample Values**
ldap.attributes['ATTRIBUTE_NAME'] or *ldap.attributes.ATTRIBUTE_NAME*	ldap.attributes.mail where ATTRIBUTE_NAME stands for any exported LDAP attribute
ldap.domainName	ldapDomain The name of the domain to which a user has been logged in
ldap.dn	cn=mike.smith The distinguished name for that LDAP server
ldap.authByEmail	0, 1

HTTP-related expressions

HTTP-related	
Expression	**Possible/Sample Values**
http.headers['HEADER_NAME'] or *http.headers.HEADER_NAME*	http.headers.myHeader where 'myHeader' is the name of any available HTTP header

Flow attributes expressions

Flow attributes	
Expression	**Possible/Sample Values**
flow.attributes['userVars.NAME'] or *flow.attributes.userVars.NAME*	Flow.attributes.userVars.name By replacing NAME with a value any additional attribute declared in user account userVars and any flow attribute declared in subscriptions userVars can be retrieved

Account-related expressions

HTTP-related	
Expression	**Possible/Sample Values**
account.disabled	0, 1
account.email	acc1@aa.bb The email of the account
account.name	acc1 The name of the account
account.notes	Notes The notes of the account
account.type	template, service, user, unlicensed
account.home	/stusers/sthome/acc1 The path to the account home directory
account.attributes.transferType	'N' – stands for Unspecified 'I' – stands for Internal 'E' – stands for Partner
account.attributes['ATTRIBUTE_NAME'] or *account.attributes.ATTRIBUTE_NAME*	account.attributes.transferType where ATTRIBUTE_NAME stands for any account custom property

User-related expressions

User-related	
Expression	**Possible/Sample Values**
account.user.loginName	acc1 The user login name
account.user.type	virtual, real, sso, siteminder
account.user.class	VirtClass The user class name
account.user.gid	1000 The user unique group identifier by which its belonging to a group of users is determined
account.user.uid	1000 User account unique identifier

Business Unit-related expressions

Business Unit-related	
Expression	**Possible/Sample Values**
account.businessUnit.name	bu1 The name of the business unit to which the account is related

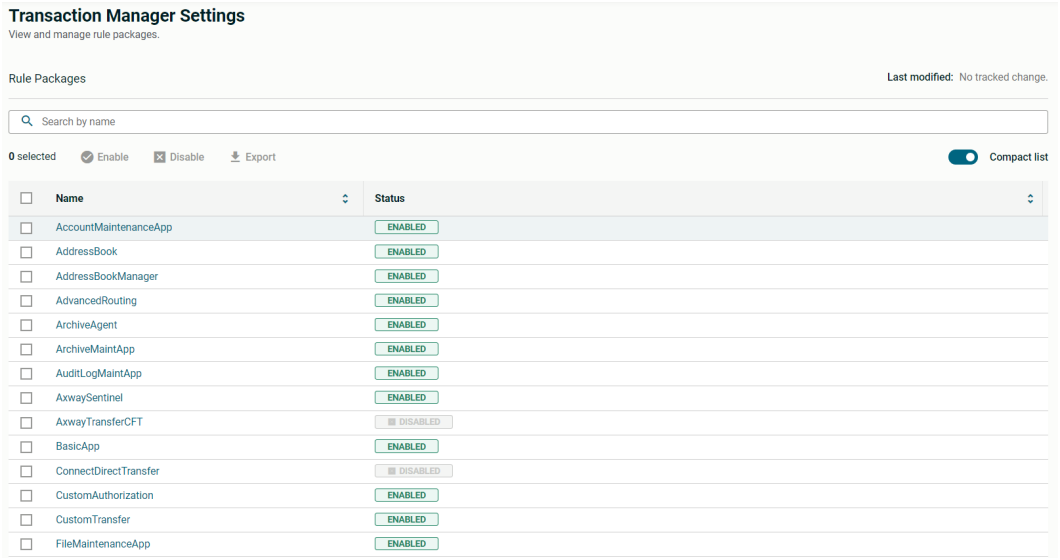
Transaction Manager Settings

The Transaction Manager is an event-based rules engine that provides a SecureTransport Server with extensible server-side functionality used for process automation and real-time delivery of data. On the Transaction Manager Settings page you can view and inspect the existing rule packages and their status.

Note SecureTransport does not automatically copy changes made under the **TM Settings** menu to other servers in your Enterprise Cluster (EC), so you must enable or disable a rule package on all servers in the cluster.

Note As part of the process that was initially started and announced with SecureTransport 5.3.0, as of SecureTransport 5.4, the access to Transaction Manager custom rules is now fully deprecated and no longer a supported feature. The core functionality that is delivered via TM Rules with customer configurable options and parameters, as documented in the Administrator's Guide, remains supported, i.e., Email notifications tuning (Velocity), Dual Authentication, ICAP functionality tuning, SNMP, Connect:Direct configuration, SendToSite functionality, Log rotating related tuning, MDNReceipts tuning, Streaming tuning, Archive agent tuning, Sentinel agent enable/disable actions.

Navigate to **Setup > TM Settings** to display the *TM Settings* page.



Rules

Rules consist of a name, the precedence setting, conditions, and actions.

- Rule Name – A descriptive identifier to distinguish the rules in a rule package.
- Precedence – A number higher than 0 used to determine which rules are executed when the conditions match more than one rule. The lower the number, the higher the priority. Rules with the highest precedence are executed. For example, a rule with a precedence of 50 is executed instead of a rule with a precedence of 100. If two rules have the same precedence value, the rules execute sequentially in no particular order. You can use NextPrecedence along with precedence values to help organize the order in which rules execute.

  Note If you need to have multiple actions fire in a certain order under the same set of conditions, you should use one rule with multiple actions in sequence.

- Condition – A condition is a logical expression that contains a comparison condition or a condition function. A condition can examine events and event attributes.
- Action – An action is a set of agents that are triggered when certain conditions are met. Actions can be either agents written in Java which allow in-process sharing of information between agent invocations or an external mechanism used to integrate with agents written in scripting languages, such as Perl or Python. Such actions can be performed through a shell mechanism.

When a user accesses a SecureTransport Server or a SecureTransport Edge, the Transaction Manager receives events from the SecureTransport Server. Depending on the event, the Transaction Manager selects the rules, matches them, and then executes them.

Built-in rule packages

SecureTransport ships with several predefined rule packages. These packages contain rules used by SecureTransport as part of the standard functionality of the product. Because these rules and packages are part of the product, it is very important that you do not delete or modify them. This topic provides a list of the built-in packages and explains their uses.

Streaming

The following packages control much of the functionality involved with authenticating users and executing transfers. Streaming and InStreaming packages contain the default functionality optimized for performance in the current release.

- Streaming
- InStreaming

Server-initiated transfers

The following packages call agents when a server-initiated transfer is executed. Each rule package represents a different protocol.

- FolderTransfer
- FtpTransfer
- HttpTransfer
- Pesit
- PesitTransfer
- SshTransfer
- STAS2

Note The STAS2 package provides AS2 functionality including file transfer and handling of receipts.

Ad hoc transfers

The following packages support ad hoc file transfers:

- AddressBook
- PackageManager

Applications

The following packages calls the agents used in the corresponding built-in application types:

- AdvancedRouting
- ArchiveMaintApp
- AuditLogMaintApp
- AxwaySentinel
- AxwayTransferCFT
- BasicApp
- FileServicesInterface
- HumanSystem
- LogEntryMaintApp
- PackageRetentionMaintApp
- SharedFolder
- SiteMailbox
- StandardRouter
- TransferLogMaintApp
- UnlicensedAccountMaintApp

The rules are required to trigger the application agent.

Permission checking

InPermissionCheck represents an implementations of checking file permissions before allowing SecureTransport operations to continue.

InPermissionCheck contains an in-process Java agent. It's efficient and provides basic check based on UID/GID and the file permission flags. For details, see *Access* on page 767.

- InPermissionCheck

Other packages

The following packages handle specific SecureTransport functionality.

The following topics describe the other Transaction Manager rule packages:

- *ArchiveAgent* on page 221
- *Axway Sentinel* on page 221
- *ConnectDirectTransfer* on page 222
- *FileServicesInterface* on page 222

ArchiveAgent

Use this package to archive transferred files. This package is enabled by default. The package copies each transferred file to the archive directory specified in the global *File Archiving* configuration page. Copied files are renamed to a unique file name to avoid duplicates. The file name format is:

```
<File_name><unique_file_name_modifier>
```

For example:

```
1223375981000_12233759819160.08926095676257206
```

The location of the files is as follows:

```
<archive_folder>/<account_name>/<login_name>/<relative_path_to_
file>/<archived_file>
```

Where:

- `archive_folder` - The archive folder location configured in the global File Archiving configuration page.
- `account_name` - Name of the account who performed the transfer. When there's no account (for LDAP users for example), account_name has value of NO_ACCOUNT.
- `login_name` - Login name for the user who performed the transfer.
- `relative_path_to_file` - Path to the file relative to the account's home folder.

For more information, see *Archive Maintenance application* on page 826 and *File archiving global configuration* on page 226 .

Axway Sentinel

This package is enabled by default. SecureTransport uses it to send file transfer and processing events to Axway Sentinel and to call the agent used in a Axway Sentinel Link Data Maintenance application. For more information about configuring SecureTransport to send events to Sentinel, see *Sentinel* on page 121.

ConnectDirectTransfer

Enable this package when you want to create and use a Connect:Direct transfer site. This package is disabled by default. For more information, see *Connect:Direct transfer sites* on page 550.

FileServicesInterface

This package is used to implement transfers initiated by SecureTransport using a file services interface protocol.

ICAPScan

This package is used to implement anti-virus or DLP scans initiated by SecureTransport using external ICAP servers. The package is disabled by default and must be enabled if ICAP servers are configured.

MDNReceipting

This package provides functionality to generate MDN receipts for the transferred files. The package is enabled by default.

Note To generate MDN receipts, create an `mdn` certificate in addition to enabling the MDNReceipting package. For more information about the `mdn` certificate, see *Certificates to generate during initial setup* on page 63.

Pesit and PesitTransfer

These packages provide the functionality for PeSIT protocol operations, including authentication, server-initiated transfers, client-initiated transfers, routed transfers, and acknowledgments. They are enabled by default.

PGPTransform

This package handles PGP encryption and decryption when Advanced Routing is not used. The package is enabled by default.

Resubmit

This package contains rules for canceling events. It is enabled by default.

SendToSite

Use this package when you want to upload files to a specific site without subscribing an account to an application. This package is disabled by default. This package is used with the **Send Files Directly To** option. For details, see *Subscribe an account to an application* on page 664.

ServerTransferNotify

Use this package when you want to enable email notifications for permanently failed server-initiated transfers. The notification template, PushDeliveryFailure.xhtml, is located under `<FILEDRIVEHOME>/conf/mailer-templates`. This package is disabled by default. For more information, see *Velocity email notification package* on page 1124.

SNMPTransferNotify

This package implements SNMP notifications for failed server-initiated transfers. This package is disabled by default.

SecureTransport includes support for Simple Network Management Protocol (SNMP) v2 and v3 to help you monitor failed transfers. Note that SNMP v3 authentication and encryption are not supported. Failure notifications are sent when transfers fail permanently after retrying the allotted number of times. The SNMP trap message format is "Transfer with *<TransferID>* has failed", and it is not customizable. SecureTransport provides a trap MIB file located at `<FILEDRIVEHOME>/conf/Transfer.mib`.

1. Navigate to **Operations > Server Configuration**.
2. Search for `TransactionManager.SNMP.enabled` and change the value to `true`.
3. Search for `TransactionManager.SNMP.ManagerHost` and change the value to the IP address of the SNMP manager.
4. Search for `TransactionManager.SNMP.ManagerPort` and change the value to the port of the SNMP manager.
5. Set the other `TransactionManager.SNMP.*` server configuration parameters as required by your SNMP manager and configuration
6. Navigate to **Setup > TM Settings**.
7. Enable the `SNMPTransferNotify` rule package.
8. Select **Operations > Server Control**.
9. Restart the **TM Server**.
10. Repeat steps 6 through 9 on each server in your cluster.

WebServicesAPI

This package supports the REST web service file transfer API.

Manage rule packages

Rule packages are a collection of rules applicable to a business process.

System administrators can view, enable, disable and export rule packages by navigating to **Setup > TM Settings**.

Note All changes to rule packages are local to the SecureTransport server you are logged on to. In an Enterprise Cluster configuration, you must enable or disable a rule package on all servers in the cluster.

View a rule package

Click on a rule package to open the XML viewer and view the rule content.

```
<brpackage xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="BasicApp.xml" version="1.0" xsi:noNamespaceSchemaLocation="businessrules.xsd">
    <rule name="BasicAppAgent" type="static" precedence="100">
        <condition>
            <expression>
                <item>
                    <attribute>EventType</attribute>
                    <comparator name="equal"/>
                    <value>Application - Incoming</value>
                </item>
                <operator name="and"/>
                <item>
                    <attribute>DXAGENT_APPLICATION_TYPE</attribute>
                    <comparator name="equal"/>
                    <value>Basic</value>
                </item>
            </expression>
        </condition>
        <action Order="yes">
            <inprocess-agent id="1" executeafter="" streamaccess="none" class="com.tumbleweed.st.server.basicapp.agents.BasicAppAgent"/>
            <inprocess-agent id="2" executeafter="1" streamaccess="none" wait="yes" class="com.tumbleweed.st.server.tm.agents.NextPrecedence"/>
        </action>
    </rule>

</brpackage>
```

Enable a rule package

To enable a rule package, select one or more that are disabled and click the **Enable** button.

Disable a rule package

To disable a rule package, select one or more that are enabled and click the **Disable** button.

Export rule packages

Use the following procedure to export one or more rule packages.

1. From the **Select** column, select the rule package or packages to export.
2. Click **Export**.

   Single packages are exported in an XML file, whereas multiple packages are exported in a ZIP file.
3. Save the file in a location of your choosing.

File archiving

The file archiving feature enables the archiving and retrieval of files at the global, business unit, and account levels.

The global File Archiving configuration page is found at **Setup > File Archiving**. There you can enable the feature and configure the global archiving policy, the archiving folder, whether or not encryption is required including the encryption certificate, and how long to keep the archived files. To configure the file archiving global configuration, refer to *File archiving global configuration* on page 226.

The file archiving configuration for each business unit can be inherited from the global file archiving configuration or changed for the current business unit. To configure file archiving for a business unit, refer to *Business units* on page 746.

For individual accounts the file archiving policy can be inherited or overwritten. To configure the file archiving policy for an account, refer to *Accounts* on page 500. File archiving can also be configured for account templates. To configure the file archiving policy for an account template, refer to *Manage account templates* on page 719.

To configure the maintenance schedule for the archive folder, refer to *Archive Maintenance application* on page 826.

Note The file archive folder and user home folders should reside on a separate storage devices. There is a negative performance impact when the archive folder is on the same storage device as user home folders due to writing data twice on the same storage device.

The archived files are stored using a structured archive format so that the following use cases are supported:

- When global archive folder is moved along with its content, the archived files are usable in their new location.
- When account is moved from one business unit to another business unit, the archived files are usable without any additional copying.
- When business unit archive folder is moved along with its content, the archived files are usable in their new location.
- When the business unit archive is moved from the global archive one to a dedicated archive, the archived files are usable without any additional copying.

The files in the structured archive folder are searchable by the following parameters:

- transfer id
- cycle id
- start date
- end date
- account name
- file name
- folder (relative to the account home folder)

The decision whether or not to archive a file is based on the following parameters:

- If an account is configured with archiving on, the file will be archived.
- If an account is configured with archiving off, the file will not be archived.

- If an account is configured with archiving inherited from a business unit and the business is configured with archiving on, the file will be archived.
- If an account is configured with archiving inherited from a business unit and the business is configured with archiving off, the file will not be archived.
- If an account is configured with archiving inherited from a business unit and the business unit is configured with archiving inherited from the global archiving configuration, the global configuration will determine whether of not a file is archived.

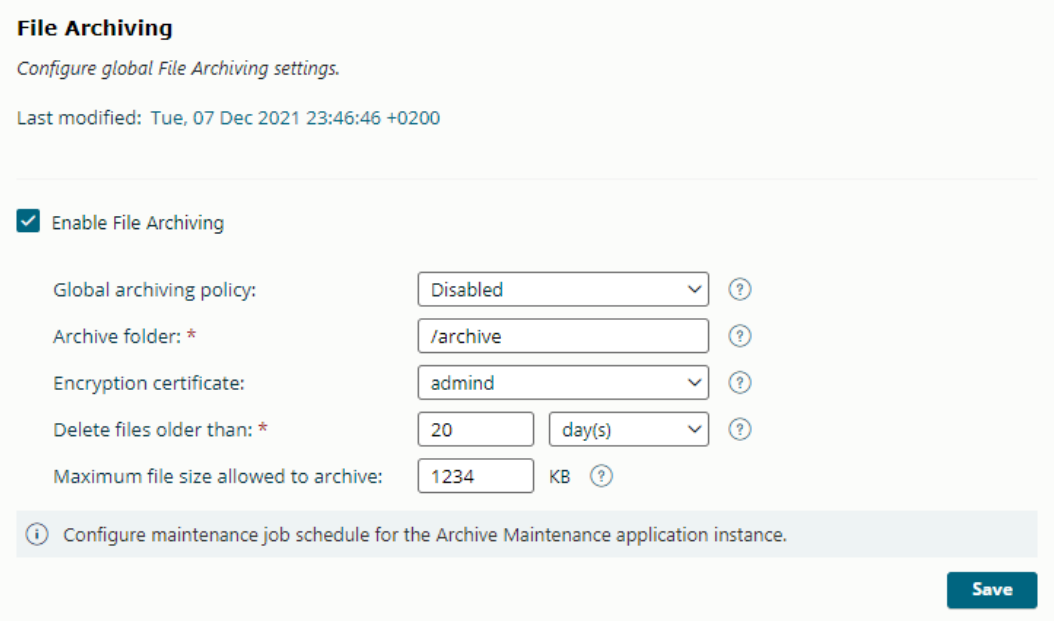
File archiving global configuration

The global file archiving configuration provides the inheritable file archiving configuration for users accounts and business units. For additional information on the inheritance of the file archiving configuration, refer to *File archiving* on page 224

To enable and configure the global file archiving configuration:

1. Navigate to **Setup > File Archiving**.

   The *File Archiving* page is displayed.



2. Select **Enable File Archiving**.

   The additional field and menus on the *File Archiving* page become active.

3. In the *File Archiving Settings* pane:

   a. (Optional) Determine the default *Global archiving policy* to be applied by selecting either **Enabled** or **Disabled**. The default is **Disabled**. The policy configured here can be overridden at Business Unit or Account level.

b. In the **Archive folder** field, enter the absolute path to the global archive folder. The folder location can be overridden at Business Unit level.

Note When resubmitting a file transfer, SecureTransport uses the current Archive folder path to search for the archived file. Therefore, files that were archived in a different folder cannot be resubmitted.

c. (Optional) Determine whether or not to encrypt the global archive folder by selecting **Do not encrypt** or which configured certificate to use for encryption from the *Encryption Certificate* menu. The default is **Do not encrypt**. The chosen certificate can be overridden at Business Unit level. The encryption certificate must be a local x.509 certificate. For information on adding local certificates, refer to *Manage local certificates and certificate signing requests on page 48*.

Note When you delete or overwrite a certificate which previously was used for encryption, all files encrypted with this certificate will be useless and can't be restored.

Note When changing the encryption certificate, the Transaction Manager should be restarted in order for the changes to be applied.

d. Set the options for **Delete files older than** field. You can specify days or hours and any number equal to or greater than 1. Files older than the configured period will be deleted by the maintenance job (if enabled).

e. Set the **Maximum file size allowed to archive** in Kilobytes. If the file is bigger or equal than the file size limit, it will not be archived. Empty or zero value means that no file size limit applies.

4. Click Save.

Note You must also configure maintenance job schedule for the Archive Maintenance application instance. For information on configuring the Archive Maintenance application instance, refer to *Archive Maintenance application on page 826*.

Note If you disable the File Archiving after being enabled and saved, the fields will stay populated with the old values, so if you enable it again you do not need to enter anything. If a certificate is selected, you cannot delete or overwrite it, even if file archiving is enabled or disabled. If you want to delete or overwrite it, you have to change the certificate option on the *File Archiving* page to another certificate or to **Do not encrypt**.

Communication across Transaction Manager, protocol, and proxy servers

SecureTransport uses a streaming protocol for communication between the protocol servers running on SecureTransport Edge and the Transaction Manager (TM) server running on SecureTransport Server. The streaming protocol abstracts all file transfer protocols and unifies and secures this central communication. When you deploy one or more SecureTransport Edge servers in a peripheral network (DMZ), the deployment is called *streaming* because no file transfer data is stored on the SecureTransport Edge server. The protocol servers translate the protocol they are serving to the streaming protocol but do not read or write files.

With a streaming deployment, the TM Server connects to the protocol servers on the configured SecureTransport Edge servers to establish the connections for the streaming protocol, so no process on a SecureTransport Edge ever makes a connection from the DMZ into the internal secure network. For more information, see *SecureTransport Edge* on page 31. (The TM server and protocol servers running on SecureTransport Server also use the streaming protocol internally.)

Note Unless a number is specified via the corresponding system properties, the number of established streaming connections from Transaction Manager to a single protocol daemon (Admin, FTP, HTTP, and so forth) is calculated with the help of formula: min(20, 2*CPUs)

SecureTransport, when used as a client, can use a proxy server. With AS2 and HTTP/S protocols, any RFC 7231 compliant HTTP proxy is expected to work. With all other protocols, SecureTransport can use the SOCKS5 proxy component of SecureTransport Edge.

You configure the communication between the TM server and protocol servers and access to SOCKS5 and HTTP proxies by defining network zones. Each network zone on a SecureTransport Server can have one or more network zone nodes that define access either within the SecureTransport Server or between the SecureTransport Server and one or more SecureTransport Edge servers. This is also applicable when your implementation uses HTTP proxy servers or the SOCKS5 proxy components of the relevant SecureTransport Edges. The TM Server connects to all protocol servers configured in all network zones. In the configuration of each transfer site, you can select a network zone to specify which proxy (HTTP or SOCKS5 on Edge) will be used for server-initiated transfers through that transfer site. SecureTransport selects a node from the network zone using a load-balancing policy when a server-initiated transfer uses the network zone.

Note Remember that HTTP proxy is only supported with HTTP(S) and AS2 transfer sites. Proxying server-initiated transfers over the other supported protocols (SSH, FTP(S), PeSIT, etc.) requires the use of the SOCKS5 proxy on SecureTransport Edge.

Because you can specify multiple SecureTransport Edge addresses in a node and multiple nodes in a network zone, you can implement any required many-to-many communication between TM servers (on SecureTransport Server nodes) and protocol servers and / or SOCKS5 proxies (on SecureTransport Edge servers).

On a SecureTransport Server, a special network zone named `Private` defines the ports used for internal communication between the TM Server and the protocol servers and the Administration Tool server running on the SecureTransport server.

On a SecureTransport Edge server, there is only one network zone, a special one named `Private` that specifies the ports that the protocol servers listen on for connections from TM servers on SecureTransport Servers. The port number must match the port number configured in the network zone on the SecureTransport Servers that defines the connection to the SecureTransport Edge server. You cannot define more network zones on a SecureTransport Edge server.

The following topics describe the streaming deployment and managing the Transaction Manager (TM), protocol, and proxy server communication:

- *Streaming deployment* on page 229
- *Manage the communication across Transaction Manager, protocol and proxy servers* on page 230

Streaming deployment

The following diagram illustrates a streaming deployment with clients and servers in both the public network and the internal secure network. The arrows show the direction of network connections for all the protocols. Data flows in both directions after the connection is made.

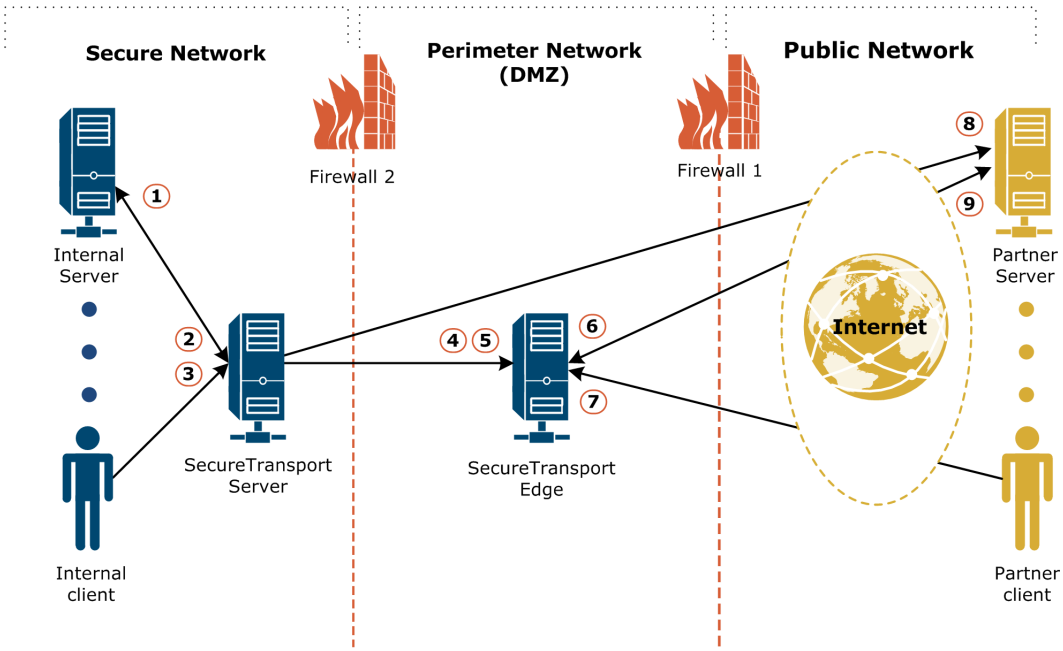


Figure 1. Streaming deployment network connections

Some lines represent two network connections. The connections are labeled as follows:

1. From the SecureTransport Server to an internal server for a server-initiated transfer
2. From an internal server to the protocol servers on the SecureTransport Server for a client-initiated transfer
3. From an internal client to the protocol servers on the SecureTransport Server for a client-initiated transfer
4. From the TM server on the SecureTransport Server to the protocol servers on the SecureTransport Edge so that the protocol servers can stream client-initiated transfers from partners using the streaming protocol
5. From the SecureTransport Server to the SOCKS5 proxy server on the SecureTransport Edge for server-initiated transfers
6. From a partner server to a protocol server on the SecureTransport Edge for a client-initiated transfer
7. From a partner client to a protocol server on the SecureTransport Edge for a client-initiated transfer
8. From the SecureTransport Server directly to a partner server for a server-initiated transfer

9. From the SOCKS5 proxy server on the SecureTransport Edge to a partner server for server-initiated transfers

The diagram does not illustrate the following network connections that you might need to configure depending on the requirements of your deployment:

- From the TM server to the protocol servers within the SecureTransport Server so that the protocol servers can serve client-initiated transfers from the secure network
- From the SecureTransport Server to a separate HTTP proxy server for server-initiated transfers
- From a separate HTTP proxy server to a partner server for server-initiated transfers
- From the TM server on the SecureTransport Server to the protocol servers and the SOCKS5 proxy server on a SecureTransport Edge in a less restrictive DMZ for internal clients and servers
- From the TM server on the SecureTransport Server to the protocol servers and the SOCKS5 proxy server on a SecureTransport Edge in a DMZ that has a VPN connection to a specific partner

Manage the communication across Transaction Manager, protocol and proxy servers

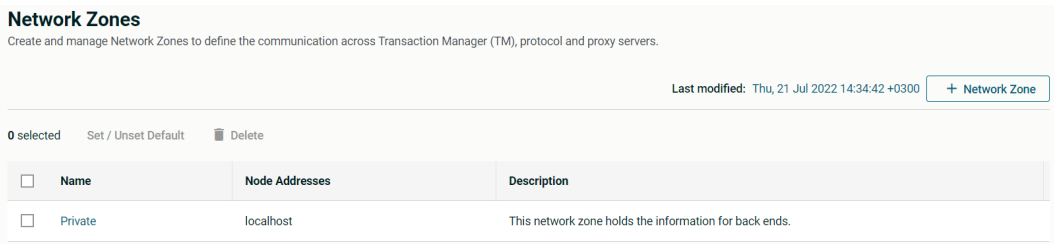
This topic describes managing the Transaction Manager (TM), protocol, and proxy server communication.

Note If the IP address in a Host of a network zone is changed, the corresponding SecureTransport Server or SecureTransport Edge must be restarted using the `<FILEDRIVEHOME>/stop_all` and `<FILEDRIVEHOME>/start_all` scripts. If the IP address in a Host of a non-private zone in a cluster environment is changed, all nodes in the cluster need to be restarted.

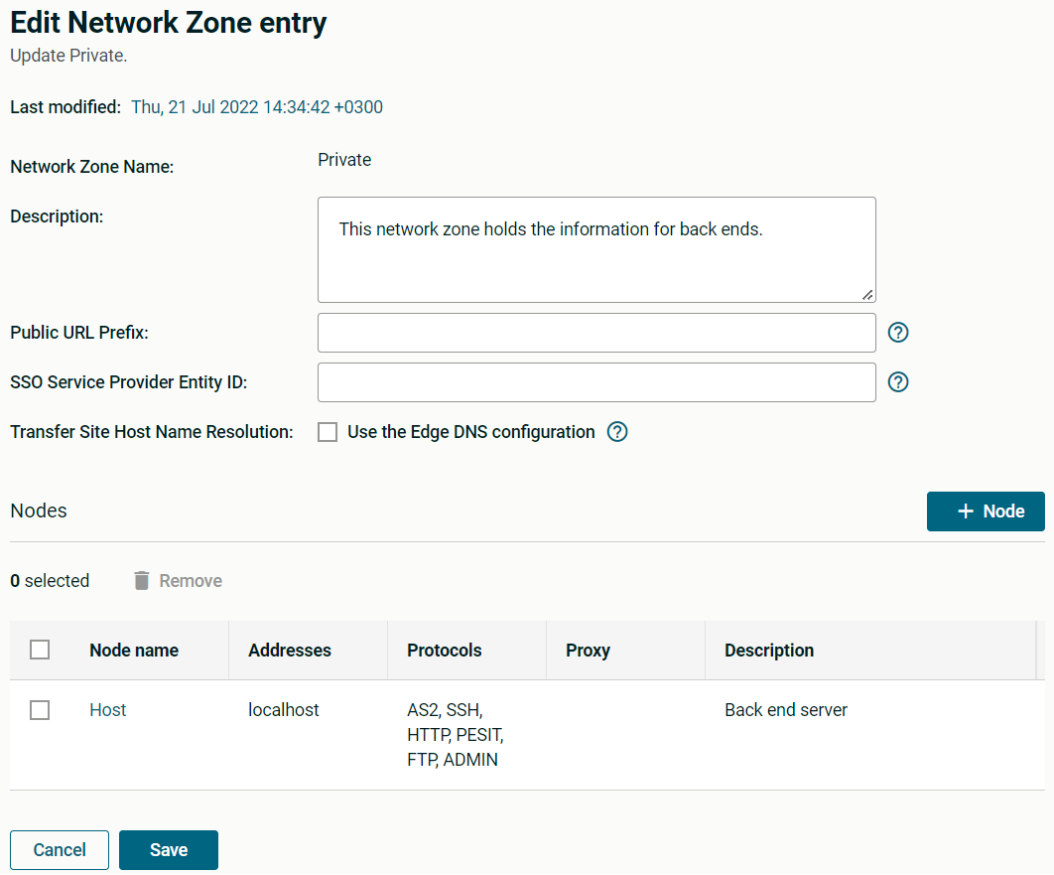
Specify ports for internal TM server communications

Use the `Private` network zone to specify the ports and certificates used for communication between the TM Server, the protocol servers and the Administration Tool server running on the SecureTransport Server. The `Private` network zone defines one node named `Host` with one address, `localhost`. For the `Private` network zone and internal Transaction Manager server communications to work correctly, `localhost` must resolve to the IP address of the system loopback device.

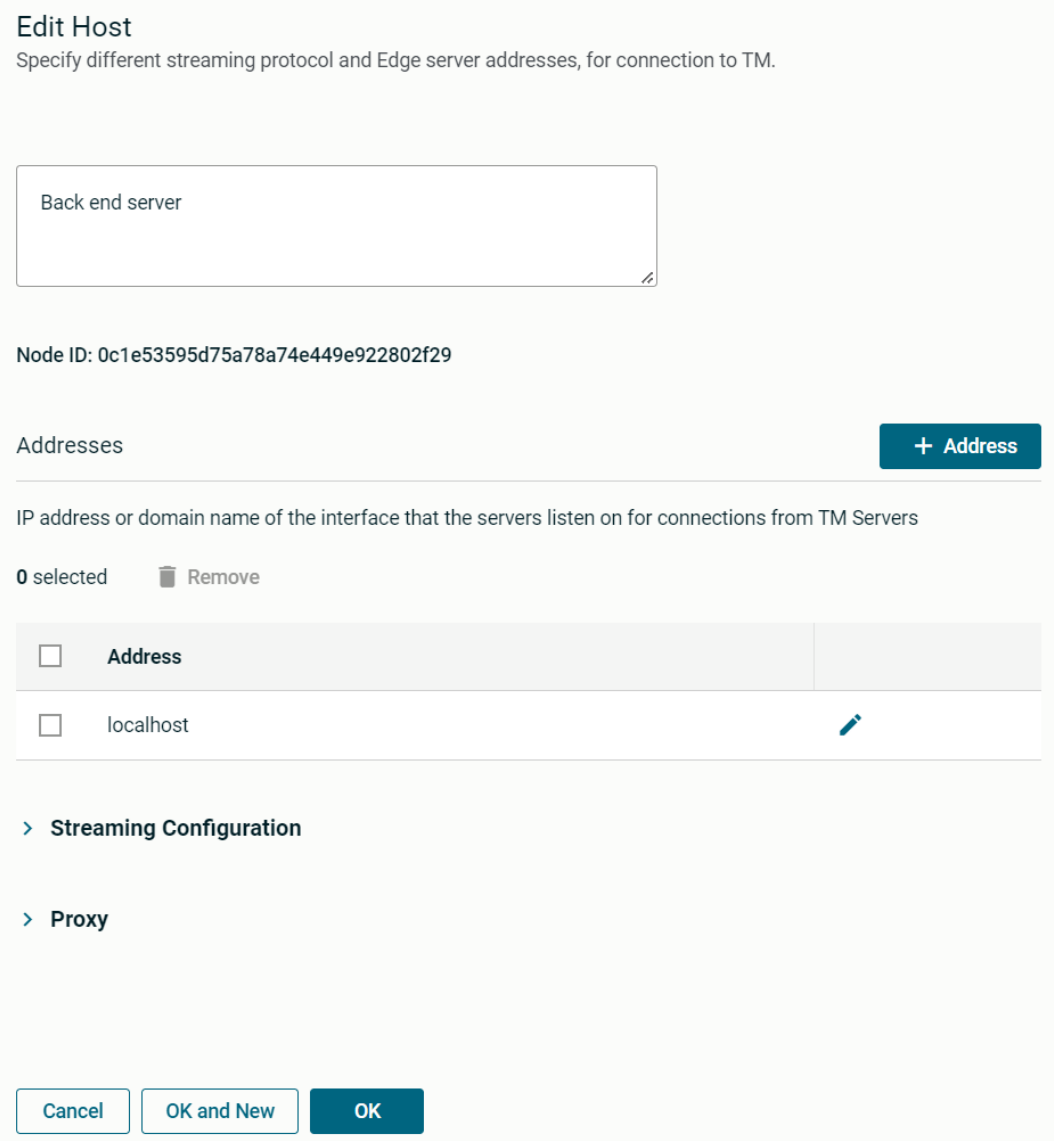
1. Navigate to **Setup > Network Zones**.



2. Select `Private` from the list. The *Edit Network Zone entry* page is displayed.



3. Select `Host` from the *Nodes* list. The *Edit Host* page is displayed.



Note The name of the `Host` node in a `Private` network zone cannot be changed.

4. Under *Streaming Configuration*, enter ports for the TM Server to connect to for each enabled protocol server and for the Administration Tool server.

   In the special `Private` network zone on a SecureTransport Server, these settings define both the ports that the TM Server connects to, and the ports that the protocol servers and the Administration Tool server listen on for connections from the TM Server.

5. (Optional) Under *Streaming Configuration*, select SSL key aliases to secure the SSL communication between the TM Server and the other servers. Do not select any of the SSL key aliases selected to secure protocol communications on the *Server Control* page. Also, add to the value of the `Streaming.TrustedAliases` server configuration parameter the aliases of the CAs that issued the certificates. For details, see the description of the `Streaming.TrustedAliases` server configuration parameter on the *Server Configuration* page.

6. Click **Save** to save your changes.

Create a network zone to define communications with SecureTransport Edge Servers

Each network zone can define the communications between the TM Server on a SecureTransport Server and one or more SecureTransport Edge Servers. If different protocol servers run on different SecureTransport Edge Servers, you can control which protocol servers on which SecureTransport Edge Servers the TM Server connects to by specifying different protocol servers and different SecureTransport Edge Servers in different nodes.

Create a network zone

1. From the main menu, click **Setup > Network Zones**.
2. Click **+ Network Zone**.

   The *New Network Zone* page opens.
3. In the **Network Zone Name** field, type a name for the network zone. You use this name to refer to that network zone in other configurations, like business units and transfer sites.
4. In the **Description** field, enter a description that helps understand the purpose and use of the network zone.
5. In the **Public URL Prefix** field, enter a prefix for the URLs displayed in the notification emails that are sent to ad hoc file transfer recipients.
   Keep the following in mind:
   - If a user is assigned to a business unit that references a network zone, emails sent to that user use the prefix specified in that network zone.
   - If a user is not assigned to a business unit, emails to that user use the prefix specified in the default network zone. See *Set a default network zone on page 236*.
6. In the **SSO Service Provider Entity ID** field, enter the Entity ID for the SSO service provider. If you leave the field empty, SecureTransport will use the entity ID specified in the `sso-enduser.xml` file under the `<ServiceProvider>` element.

   Note After you specify an Entity ID and restart the TM, SecureTransport will clone the `sso-enduser.xml` file (if it exists) and change the `entityId` attribute in the *<ServiceProvider>* element, which will be used by the corresponding Edges when the user is trying to authenticate via this network zone.
7. If the *SOCKS5* proxy is enabled in this network zone and DNS is not available on this SecureTransport Server, select the **Use the Edge DNS configuration** checkbox..
8. Add a node. See *Add a network zone node on page 234*.
   Nodes can be added at any time.
9. Click **Save**.

   SecureTransport saves the network zone in the database.

Add a network zone node

1. From the main menu, click **Setup > Network Zones**.
2. Click **+ Node**.
3. Type a name for the new node.
4. If this server is part of a Disaster Recovery deployment, select the value for **Deployment Site**. For more information, see *Set up a disaster recovery cluster on page 383*.
5. In the **Description** field, enter information to help you understand the purpose and use of the node.
6. Click **+ Address**.

   The **Addresses** section expands with a text field for entering a single address. Addresses are added only manually one by one.

   Note On test environments only, you may try out the Dynamic Node IP Addresses Discovery feature. It is still in beta and must not be used in production environments. See *Dynamic Node IP Discovery* .
7. In the **Address** field, enter either the IP address, the hostname or the FQDN of an Edge Server that runs the protocol (or proxy) servers configured for this network zone node.
8. Optional: To add more addresses for a node, repeat steps 6 and 7.
9. Under *Streaming Configuration*, select the protocol servers that the TM Server connects to using this node of this network zone.
10. For each enabled protocol server, enter the port for the TM Servers to connect to.

    For a connection to be successful, ensure that for each SecureTransport Edge Server configured under *Addresses*:

    - the SecureTransport Edge Server allows connection from this SecureTransport Server
    - the `Private` network zone specifies the same port for the protocol server
    - the client certificate public part is imported into the trusted SecureTransport Edge CAs.

    For more information, see *Specify allowed SecureTransport Servers on SecureTransport Edge on page 238* and *Specify TM Server communication ports and IP address for protocol servers on SecureTransport Edge on page 236*.
11. Click **OK** to save this node, or **OK and New** to save this node and add another.

Specify proxy settings in a network zone

Use network zones to specify how SecureTransport connects to the SecureTransport Edge SOCKS5 or HTTP proxy for server-initiated file transfers. You can use the same network zone node to define streaming and proxy configuration, or you can define only streaming or only proxy configuration in a node or a network zone.

Note When proxying server-initiated transfers over SSH, FTP(S), PeSIT, etc., using the SOCKS5 proxy on SecureTransport Edge is required. An external HTTP proxy can be used with HTTP(S)-based and AS2 transfer sites.

When you define an AS2, FTP(S), HTTP(S), PeSIT, or SSH transfer site, you can select a network zone to specify the proxy servers for transfer through that transfer site.

1. Navigate to **Setup > Network Zones**
2. Click on a name in the list, or create a new network zone. See *Create a network zone to define communications with SecureTransport Edge Servers on page 233*.
3. Click on a name in the *Nodes* list, or click **+ Node**.
4. For a new node, enter values in the **Node name** and **Description** fields as needed.
5. Under *Proxy*, select **Enable Proxy**.
6. Select the proxies, **SOCKS5** or **HTTP**, to configure in the node.
7. For each selected proxy:
    a. Enter the **Port**. The proxy server must listen for connections on this port. For the SOCKS5 proxy on SecureTransport Edge, configure the port for the proxy server on the *Server Control* page. See *Manage the Proxy server on SecureTransport Edge*. If you use a third-party proxy server, refer to its documentation to configure the port.
    b. (Optional) Enter a **Username**, select **Use Password** and enter a **Password**, if required by the proxy server. The SOCKS5 proxy on SecureTransport Edge does not use this authentication.
8. For a new node, add the address of the SecureTransport Edge Servers that run the SOCKS5 proxies, or the addresses of the other proxy servers. If you selected **SOCKS5**, you must use IPv4 addresses. SecureTransport Server selects the servers at these address sequentially (round-robin) when it uses this node for server-initiated transfer.
9. Click **OK** to save this node, or **OK and New** to save this node and add another.
10. If DNS is not available on this SecureTransport Server, select **Use the Edge DNS configuration** to resolve transfer site host names and FQDNs using the DNS on the SecureTransport Edge, so that SecureTransport can route transfers correctly.
11. Click **Save** to save the network zone.

Proxy states and blacklisting

SecureTransport provides a mechanism for dynamic blacklisting of SecureTransport Edge proxies in case of failure. An Edge proxy can be in one of three states: *available*, *failed*, or *denied*. It is added to a list of *failed* proxies if a connection cannot be established through it. The failed proxy then becomes *denied* when the number of failed connection attempts, defined in the `Proxy.Max.Failure.Series` option (3 by default), is reached within the time interval specified by the `Failed.Proxy.Timeout` option value (3 minutes by default). At this point, the *denied* proxy is blacklisted, and no connections are routed through it, for the period of time specified by the `Denied.Proxy.Timeout` configuration option (10 minutes by default). After the blacklisting timeout period elapses, the proxy returns its state to *available*.

The *failed* and *denied* Edge proxies lists are local to each server node and cleared upon TM restart. As a result, a backend node in a multi-node cluster may still make connection attempts to a proxy that is in another node's *denied* proxies list. This may result in errors and even failed server-initiated transfers, as the backend node is yet to detect a failing Edge proxy.

To minimize the likelihood of such an event occurring, set the configuration options `Denied.Proxy.Timeout`, `Failed.Proxy.Timeout`, `Proxy.Max.Failure.Series`, and `EventQueue.maxRetryCount` to values that will allow all nodes in the cluster to detect potential unavailability earlier.

By default, proxy blacklisting is enabled. To disable it, set the `Proxy.Blacklisting.Enabled` configuration option to `false`. Changing the value requires a TM restart.

When the blacklisting functionality is disabled, the network zones are always considered alive, the `Denied.Proxy.Timeout`, `Failed.Proxy.Timeout`, and `Proxy.Max.Failure.Series` values are disregarded and backend nodes will not detect potential Edge proxies unavailability.

Set a default network zone

Business units can specify a network zone that defines the public URL for users in that business unit. For details, see *Create or edit a business unit*. AS2, FTP(S), HTTP(S), PeSIT, and SSH transfer sites can specify a network zone that defines the proxy for that site. For details, see *Transfer sites on page 540*. In all cases, you can select **Default** in the business unit or transfer site.

The default network zone defines the public URL for users in business units where you selected **Default** and for users who are not in a business unit. The default network zone also defines the proxy for transfer sites where you selected **Default**. If a transfer site selects the default network zone and no default zone is defined, transfers from that site fail.

1. Navigate to **Setup > Network Zones**.
2. Select a network zone from the list.
3. Click the **Set / Unset Default** button.

   The selected network zone is marked as default.

Specify TM Server communication ports and IP address for protocol servers on SecureTransport Edge

The `Private` network zone is the only network zone defined on the SecureTransport Edge Server. You cannot delete it or define more network zones.

Use the `Private` network zone to specify the ports and IP address that the protocol servers and the Administration Tool server listen on for connections from TM Servers. The `Private` network zone defines one node named `Host`.

1. Navigate to **Setup > Network Zones**.
2. Click on `Private`.

3. In the *Nodes* list, click on `Host`.
4. For each enabled protocol server, enter the port that the protocol server listens on for connections from TM Servers.

   For a connection to be successful, ensure that:

   - the SecureTransport Edge Server allows connection from this SecureTransport Server
   - there is a network zone on a SecureTransport Server with these same ports configured for the host name or IP address of this SecureTransport Edge Server
   - the certificate is trusted on the SecureTransport Server.

   For more information, see *Specify allowed SecureTransport Servers on SecureTransport Edge on page 238* and *Create a network zone to define communications with SecureTransport Edge Servers on page 233*.
5. Under *Addresses*, click the edit icon ( ) for the `localhost` entry.
6. In the **Address** field, enter the IP address of the interface that the servers listen on for connections from TM Servers.
7. Click the Save icon ( ).
8. Click **OK** to save your changes.

Secure the communication between the TM server and the protocol servers

You can use TLS/SSL to secure the communication between the TM server running on theSecureTransport Server, and the protocol servers running on SecureTransport Edge.

In a streaming deployment, the protocol servers on SecureTransport Edge have the role of server because the TM server on SecureTransport Server connects to them and the TM server on SecureTransport Server has the role of client because it connects to the protocol servers on SecureTransport Edge.

1. Generate or obtain the following certificates:
   - The TM server client certificate with `extendedKeyUsage` = `clientAuth` and `keyUsage` = `digitalSignature`
   - The protocol servers server certificate with `extendedKeyUsage` = `serverAuth` and `keyUsage` = `digitalSignature`, `keyEncipherment`
2. On SecureTransport Edge:
   a. Import into the trusted CAs the public part of the certificate for the CA used to generate the TM server client certificate.
   b. Add to the `Streaming.TrustedAliases` server configuration parameter the alias you specified when you imported the certificate in step a.
   c. Import into the local certificates the protocol servers server certificate.

d. In the `Private` network zone, open the *Edit Node* page. To secure the TLS/SSL communication for a protocol, under *Streaming Configuration* select the **SSL Key Alias** you specified when you imported the certificate in step c.

3. On SecureTransport Server:
   a. Import into the trusted CAs the public part of the certificate for the CA used to generate the protocol servers server certificate.
   b. Add to the `Streaming.TrustedAliases` server configuration parameter the alias you specified when you imported the certificate in step a.
   c. Import into the local certificates the TM server client certificate.
   d. In the network zone you created to define communications with SecureTransport Edge Servers, open the *Edit Node* page. To secure the TLS/SSL communication for a protocol, under *Streaming Configuration* select the **SSL Key Alias** you specified when you imported the certificate in step c.

Specify the SecureTransport Edge load-balancing policy

When a client establishes a user session to a protocol server running on SecureTransport Edge, the load-balancing policy that SecureTransport Edge uses to allocate a Transaction Manager connection to that session is specified by the `Streaming.LoadBalancingPolicy` server configuration parameter. The valid values are:

- **`Round-Robin`** — SecureTransport Edge directs connections to connected Transaction Managers in a circular sequence and allocates the least-used connection to that Transaction Manager to the session.
- **`Random`** — SecureTransport Edge randomly selects a connection from all the connections to all the connected Transaction Managers and allocates it to the session.
- **`Blacklist-Round-Robin`** — SecureTransport Edge directs connections to connected Transaction Managers in a circular sequence and allocates the least-used connection (but filters bad connections) to that Transaction Manager to the session. `Blacklist-Round-Robin` is the default and recommended value.
- **`Blacklist-Random`** — SecureTransport Edge randomly selects a connection from all the connections (but filters bad connections) to all the connected Transaction Managers and allocates it to the session.

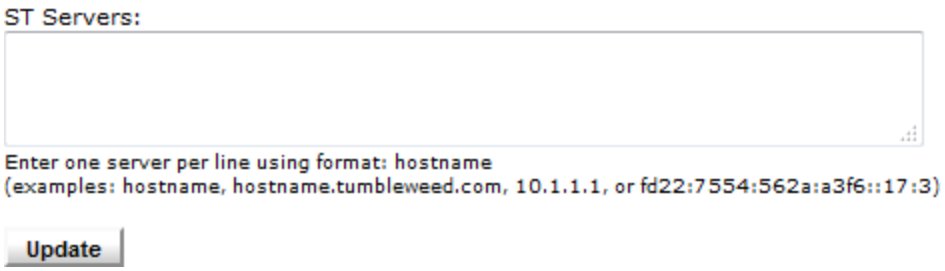
In all cases, the user session uses the allocated connection until it terminates.

Specify allowed SecureTransport Servers on SecureTransport Edge

The SOCKS5 proxy on a SecureTransport Edge Server accepts connections for server-initiated transfers only from the listedSecureTransport Servers. There is no HTTP proxy on the SecureTransport Edge Server. The Transaction Manager on any SecureTransport Server can connect to a protocol server on any SecureTransport Edge Server.

1. Navigate to **Setup > Allowed ST Servers**.

   A page is displayed with the servers list.

2. Under *ST Servers*, list the host names or IP address of the SecureTransport Servers that are allowed to connect to this SecureTransport Edge Server, either to a protocol server, or to a SOCKS5 or HTTP proxy.
3. Click **Update**.
4. Restart the SOCKS proxy on the SecureTransport Edge Server.

Configure SecureTransport Server to Edge streaming communication

The Edge to Server communication uses Network Zones that allow one or multiple Edge servers to be grouped in a zone that all inbound and outbound communications will go through. This makes it easy to set up multiple peripheral networks, for example, one for intranet and one for Internet traffic. Additionally, the communications are entirely outbound from Server perspective. That means that no inbound ports in the firewall between the Edge and Server need to be opened.

To configure streaming communications between the Edge and Server, take the following steps:

1. Exchange the CA certificates of the Server and Edge: the CA certificate of the Server should be exported and imported on the Edge, and vice versa. For additional information, refer to *Manage trusted CAs on page 55*.
   a. On the Server, go to **Setup > Certificates**.
   b. Click the **Trusted CAs** tab.
   c. Navigate to the page that lists the certificate labeled *ca*.
   d. Click the alias of the CA certificate.
   e. Click **Export** and save the certificate file.
   f. On the Edge, go to **Setup > Certificates**.
   g. Click the **Trusted CAs** tab.
   h. Click **Import**.
   i. Browse and select the CA certificate exported from the Server.

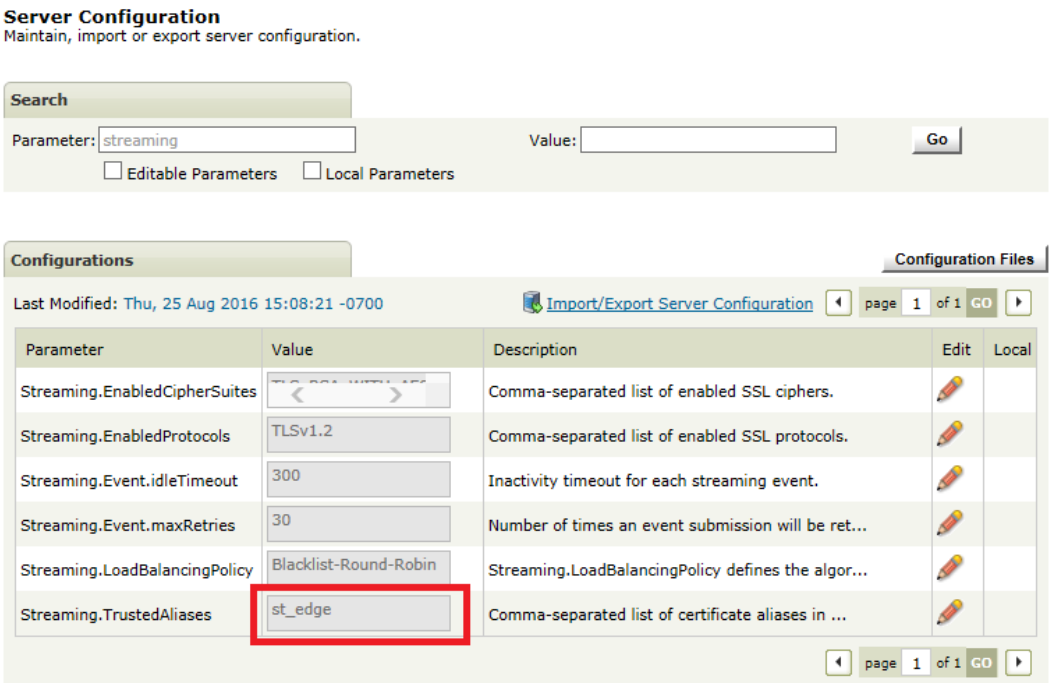
j. Enter a unique alias for the CA certificate. Do not overwrite existing certificates.

k. Click **Import**.

l. Follow the steps **a** to **k** to export the internal CA of the Edge and import it on the Server.

2. On the Server, generate a self-issued local certificate that will only be used for streaming communications. For additional information, refer to *Generate a self-issued server certificate on page 49*.

3. Set the `Streaming.TrustedAliases` configuration option to the alias of the imported CA certificate of the Edge.



4. Add a new zone on the Server for the Edge. See *Manage the communication across Transaction Manager, protocol and proxy servers on page 230*

   Note The Private zone on the Server should not be modified.

5. Specify the title and description, and add nodes.
6. In the *Streaming Configuration* pane, select the protocols to be used for the streaming communications.
7. Enter the port number of each selected protocol.
   The Server will connect to the Edge to establish the streaming connections using these ports. The streaming connections are established in outbound direction - from Server to Edge. The port numbers must match between the Server and the Edge.

   Note The streaming port numbers and certificate alias must be different from the ones used for server control.

8. For the **SSL Key Alias** on each selected protocol, select the alias of the certificate generated in Step 2.

   Note **ADMIN** must be selected whenever another streaming protocol is selected. The ADMIN protocol is used for internal functions like DNS resolving and DNS reverse lookups, see *Enable or disable reverse DNS lookups on page 201*.

9. In the *Addresses* section, add the IP address of the Edge.
10. On the Edge, generate a self-issued local certificate server that will only be used for streaming communications. For additional information, refer to *Generate a self-issued server certificate on page 49*.

11. Edit the Network Zone node to change the default **localhost** to the IP address of the Edge server itself, see *Create a network zone to define communications with SecureTransport Edge Servers on page 233*.

    Note **ADMIN** must be selected whenever another streaming protocol is selected. The ADMIN protocol is used for internal functions like DNS resolving and DNS reverse lookups, see *Enable or disable reverse DNS lookups on page 201*.

12. Verify that the ports for the selected protocols are the same and match the ports on the Server.
13. For the **SSL Key Alias** on each protocol, select the alias of the certificate generated in Step 10.
14. Set the `Streaming.TrustedAliases` configuration option to the alias of the imported CA certificate of the Server.
15. Restart all of the services (`stop_all`/`start_all`) on both the Server and the Edge.
16. Wait for the Transaction Manager to connect to the Edge (at least 2 minutes). If the streaming communication is successful, you will see "Established streaming connection" log messages on the *Server Log* page.
17. Test client-initiated transfers by logging in using the address of the Edge.

Note By default, the file timestamp is calculated based on where the client has logged in from - the Edge or the Server. If they are in different time zones, there will be a mismatch in the displayed timestamp for the same file. You can set the Edge to use the Server's time zone by changing the `Edge.TimeZone.syncWithBackend` server configuration option to `true`. Restart the Edge services to apply the change.

Address Book

The Address Book feature provides built-in and custom address books data sources to the SecureTransport Server. End users are allowed via the ST Web Client to send messages or share folders to a predefined list (address book) of users and groups. End users are able to send or share folders directly by using the display name defined in the address book. Implementing the Address Book functionality allows SecureTransport administrators to control the users' collaboration with external (non-address book) users.

The global Address Book configuration page is found at **Setup > Address Books**. There you can configure the global address book sources list, enable or disable global address books, and determine whether or not to allow address book collaboration. To configure the global address book sources list, refer to the *Address Book global level configuration on page 245*.

Note The *Address Books* page is only displayed if the Address Book feature is enabled (the value of the `AddressBook.Enabled` configuration option is set to **true**).

The Address Book sources list configuration for each business unit can be inherited from the global Address Book sources list configuration, or changed for the current business unit. To configure the Address Book sources list for a business unit, refer to the *Address Book business unit level configuration on page 248*.

For individual accounts the Address Book sources list can be inherited or overridden. To configure the Address Book sources list for an account, refer to the *Address Book account level configuration on page 249*.

Additionally, the LDAP address book settings can be configured, see *LDAP domains on page 478*.

Address Book sources

The Address Book feature has two built-in address book sources – Local source and LDAP source. These two sources become available by default when the Address Book functionality is enabled. Multiple custom address book sources can also be registered on the SecureTransport Server.

Local source

The local Address Book source includes local SecureTransport accounts with a registered email address, and has two options for specifying the address book entries:

- User's own Business unit – This includes entries which belong to the same business unit as the account the local source is assigned to (including sub business units).
- All Business units – This includes all virtual accounts in the SecureTransport Server which have a registered email address.

All entries are populated in a tree structure; for example, there is one parent group for all entries – Parent Display Group. This parent group name is based (evaluated) on the Address Book source configuration – either account, business unit, or global level. Each business unit is also considered a parent group to all accounts that belong to it. Therefore each account produces an address book user entry with the parent group as its business unit (or the parent group defined for the local source if the account does not have a business unit), and each business unit produces an address book group entry. If there are nested business units, the resulting Address Book entries structure is nested in the same way.

LDAP source

The LDAP source specific settings are defined in the LDAP configuration page. For more information on configuring the Address Book settings for a domain, refer to *Define Address Book settings for a domain on page 492*.

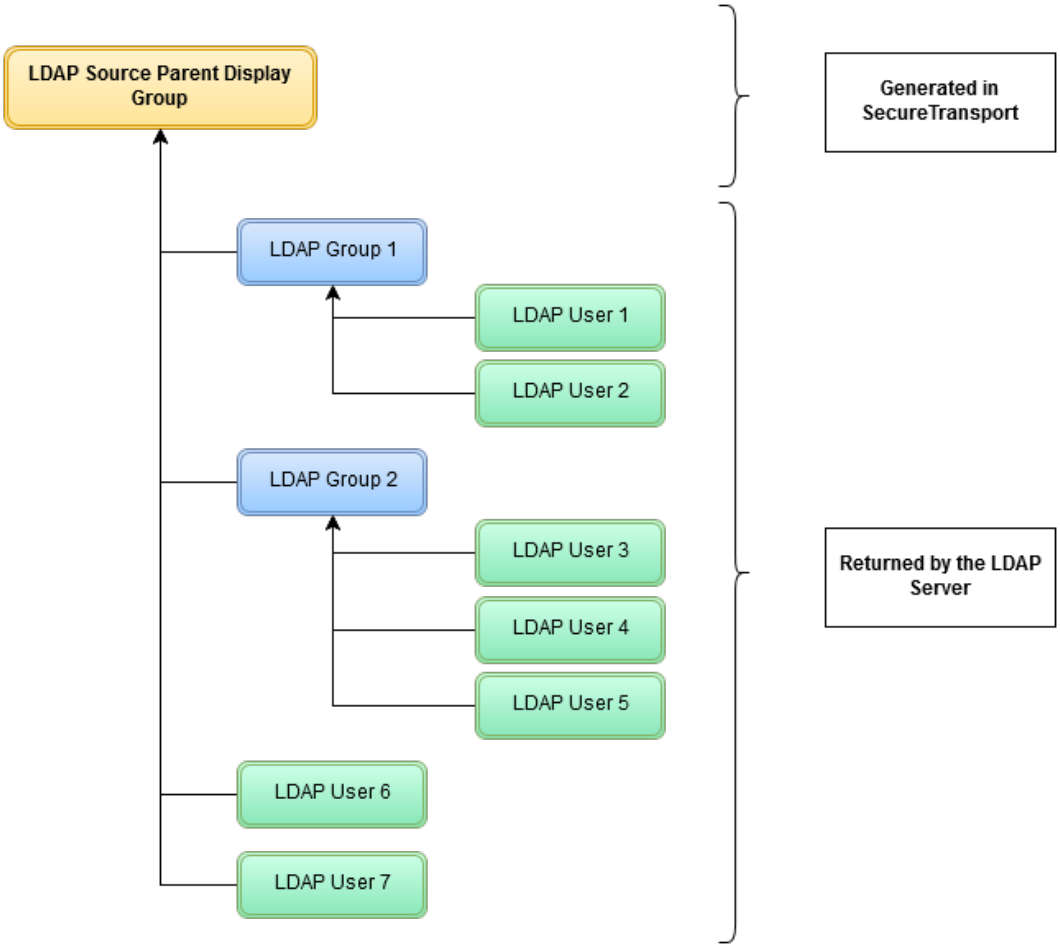
The Address Book entry map settings, as well as the search query, are displayed on the LDAP configuration page. There are 3 predefined properties for an entry:

- Primary Email
- Display Name
- Parent Group(s)

The mapping configurations allow defining new properties, which can be extracted from the LDAP object. All additionally added key value pairs are populated as custom properties of the address book entry.

When fetching new entries from the LDAP source, if a `parentGroup` mapping has been specified by the SecureTransport administrator, its value will override the `parentGroup` value defined for the LDAP source. For example, the LDAP source is configured as an Address Book data source, it is enabled, has LDAP Source as its parent group, and the `parentGroup` property has been mapped to the LDAP attribute group. All entries which are fetched from this source will override the LDAP source value and will map the value of the group attribute to the `parentGroup` property instead. Only the direct parent of an entry will be displayed in its `parentGroup` property.

All entries are populated in a tree structure; for example, there will be one parent group (root) for all LDAP entries. The parent group name is based (evaluated) on the Address Book source configuration – either account, business unit, or global level. All nested parent groups will depend on the `parentGroup` attribute mapping. In other words, the tree structure of entries returned by the LDAP server will be attached as child of the evaluated root parent group, defined for the LDAP source. In that case, all LDAP groups and user entries which do not belong to a group will be considered as direct children of the root parent group. This allows all entries coming from a source to be addressed using a single display name – the source Parent Display Group.



Custom source

For instructions on how to implement a custom Address Book source, refer to the *SecureTransport Developer Guide*.

Address Book configuration settings

The Address Book data source configurations are arranged hierarchically in three levels. The first level of configuration is the global level, the second is the business unit level, and the third is the account level. Each level can either inherit or override the Address Book configuration settings of the previous level.

For example:

A custom source "MSSQL DS" is enabled at the global level and then "MSSQL DS" is assigned to a business unit. For this business unit, the SecureTransport administrator can enable or disable the source, specify the group property, and set permissions if these settings can be overridden by delegated administrators for accounts in this business unit. All of these settings are taken with higher precedence than the global settings.

The order of the Address Book data source hierarchy is:

- Account level
    - Business unit level
        - Global level

Address Book global level configuration

The Address Book feature is disabled by default, and the Address Books page where you configure the global level Address Book settings is hidden. To use the feature, you need:

- An email address configured in the user account settings. See *Create a user account* on page 503.
- The `AddressBook.Enabled` configuration option set to **true**.

  When enabled, the Address Book user interfaces appear in the Administration Tool, allowing configuration on global, business unit, and account levels. New REST API endpoints also become available for configuring and querying the Address Book. When disabled, the Address Book related UI is hidden, REST API endpoints return empty result set to the client, and all Address Book related configuration settings (described below) are not taken into consideration.

All Address Book configuration settings are stored under the `AddressBook` namespace in the server configuration:

- `AddressBook.AllowCollaboration` - Default setting for controlling Address Book collaboration with external recipients. This setting can be set from the Address Book global setting **Setup > Address Books > Allow collaboration with non-Address Book recipients** and can be overridden on the business unit and account levels. If the Address Book feature is disabled, this setting is ignored.

- `AddressBook.Limit.AdHoc.Recipients` - The total number of recipients allowed for AdHoc packages and Shared Folders collaboration. If the recipient count exceeds this value, an error will be returned to the client. The default value is **100**.
- `AddressBook.Limit.DefaultDisplayEntries` - Specifies the default limit of Address Book entries displayed to the user if no limit is specified in the request. The default value is **10**.
- `AddressBook.Limit.MaxDisplayEntries` - Specifies the maximum number of Address Book entries returned by the server in a single response. The default value is **100**.
- `AddressBook.Limit.MaxDisplayPages` - Specifies the maximum number of Address Book pages in a single request. The maximum number of entries returned by an Address Book source is the multiplication of this value and `AddressBook.Limit.MaxDisplayEntries`. The default value is **100**.
- `AddressBook.Search.Min.Length` - Specifies the minimum search string length when performing a wildcard search for Address Book entries. The default value is **3**.

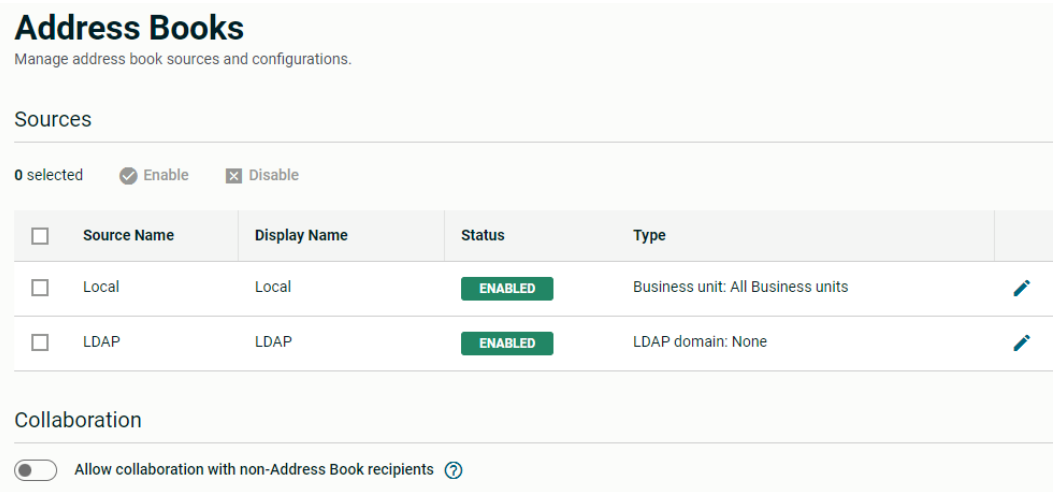
Address Book global sources list configuration

You can configure (enable and disable) Address Book data sources and specify if collaboration is allowed for the whole system. You cannot add or remove address books from the Address Book sources list on the *Address Books* page because of the address book provider auto-discovery registration process. This process works in the following way: in order to add a new Address Book source, the Address Book provider implementation package must be placed inside the `<FILEDRIVEHOME>/lib/jars` folder. Upon Transaction Manager startup, the provider is dynamically loaded and any new Address Book sources are registered in the SecureTransport system. For more information, see the *SecureTransport Developer Guide*.

To view and configure the global Address Book sources list:

Navigate to **Setup > Address Books**.

Note The *Address Books* page is only displayed if the Address Book feature is enabled (the value of the `AddressBook.Enabled` configuration option is set to **true**).



Enable an Address Book source

1. Select the Address Book source to enable.
2. Click **Enable**.

Disable an Address Book source

1. Select the Address Book source to disable.
2. Click **Disable**.

Edit the Local source

1. Click the **Edit** icon ( ).
2. From the *Type* menu select either **All Business units** or **User's own Business unit**. If the local source type is **All Business units**, the user can send emails to all SecureTransport accounts that have a configured email address. If the local source type is **User's own Business unit**, the user can send emails only to accounts in the same business unit.
3. Click the **Save** icon ( ), or cancel the change ( ).

Edit the LDAP source

1. Click the **Edit** icon ( ).
2. From the *Type* menu select the **Domain**.
3. Click the **Save** icon ( ), or cancel the change ( ).

Edit a Custom source

1. Click the **Edit** icon ( ).
2. Edit the **Display Name**.
3. Click the **Save** icon ( ), or cancel the change ( ).

Allow collaboration with non-Address Book recipients

1. Select **Allow collaboration with non-Address Book recipients**.
   - When **enabled**, accounts that use the global Address Book policy are allowed to send email packages and share folders with users that do not exist in the defined Address Book.
   - When **disabled**, accounts that use the global Address Book policy are allowed to send email packages and share folders only with users that exist in the defined Address Book.

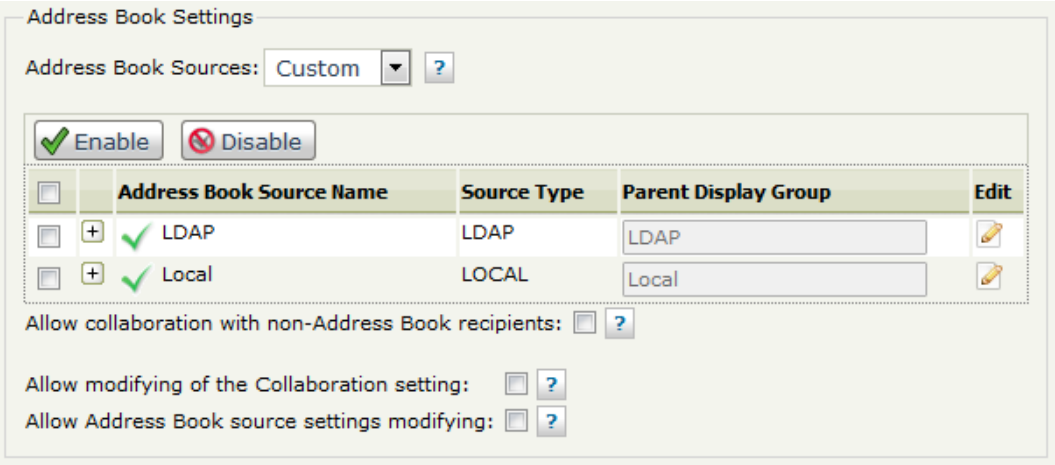
   This global setting can be overridden on business unit and account configuration levels.

Note Users will be able to share folders when **Allow collaboration with non-Address Book recipients** is enabled only if the `SharedFolders.AllowCollaboration` server

configuration parameter is set to **true**. For more information on changing server configuration parameters, refer to *View and change server configuration parameters on page 334*.

Address Book business unit level configuration

Navigate to **Accounts > Business Units** and select the relevant business unit. At the business level, SecureTransport administrators can configure the following settings:



- Address Book Sources: Select the Address Book policy that will apply for the current business unit. There are three options:
  - **Default** - When selected, the business unit inherits either its parent's Address Book policy, or the global Address Book policy if it is a top level business unit.
  - **Custom** - When selected, a custom Address Book policy configuration will be set for this business unit only. A list of all registered Address Books is displayed, and for each the administrator is able to:
    - Enable or disable Address Book sources for the business unit.
    - Specify the parent group for a given Address Book source.
    - Specify the domain for LDAP Address Book sources.
    - Specify **All Business units** or **User's own Business unit** for local and custom Address Book sources.
  - **Disabled** - When selected, the Address Book policy is set to disabled for this business unit.
- Allow collaboration with non-Address Book recipients
  - When **checked**, accounts that use the Address Book policy defined on the business unit level are allowed to send email packages and share folders with users that do not exist in the defined Address Book.
  - When **unchecked**, accounts that use the Address Book policy defined on the business unit level are allowed to send email packages and share folders only with users that exist in the defined Address Book.

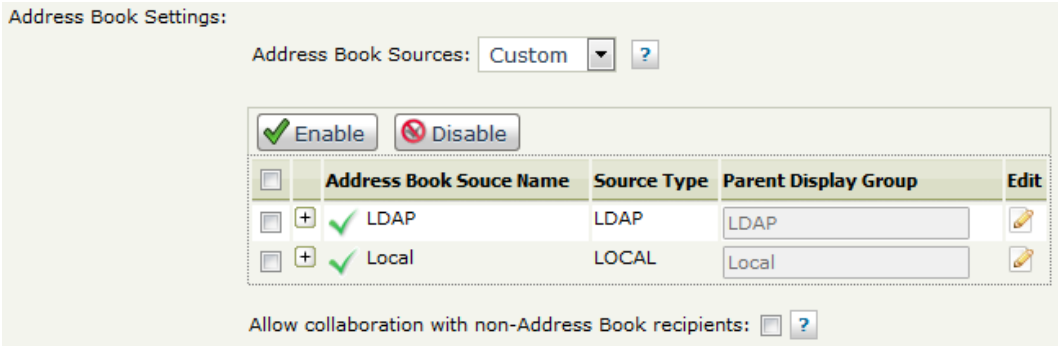
This business unit setting overrides the global Address Book policy setting for collaboration. This setting can be overridden on the account level if **Allow modifying of the Collaboration setting** is checked.

Users will be able to share folders when **Allow collaboration with non-Address Book recipients** is checked only if the `SharedFolders.AllowCollaboration` server configuration parameter is set to **true**. For more information on changing server configuration parameters, refer to *View and change server configuration parameters on page 334*. If the Address Book feature is disabled, this setting does not affect user collaboration.

- Allow modifying of the Collaboration setting
  - When **checked**, accounts that use the Address Book policy defined in this business unit are allowed to override the **Allow collaboration with non-Address Book recipients** setting specified here.
  - When **unchecked**, accounts that use the Address Book policy defined in this business unit are not allowed to override the **Allow collaboration with non-Address Book recipients** setting specified here.
- Allow Address Book source settings modifying
  - When **checked**, accounts that use the Address Book policy defined in this business unit are allowed to override the Address Book sources specified here.
  - When **unchecked**, accounts that use the Address Book policy defined in business unit are not allowed to override the Address Book sources specified here.

Address Book account level configuration

Navigate to **Accounts > User Accounts** and select the relevant account. At this level, SecureTransport administrators can configure the following settings:



- Address Book Sources: Select the Address Book policy that will apply for the current account. There are three options:
  - **Default** - When selected, the account inherits either its business unit Address Book policy, or the global Address Book policy.
  - **Custom** - When selected, a custom Address Book policy configuration will be set for this account only. A list of all registered Address Books is displayed, and for each the administrator is able to:

        - Enable or disable Address Book sources for the account.
        - Specify the parent group for a given Address Book source.
        - Specify the domain for LDAP Address Book sources.
        - Specify **All Business Units** or **User's own business unit** for local and custom Address Book sources.
    - **Disabled** - When selected, the Address Book policy is set to disabled for this account.
- Allow collaboration with non-Address Book recipients
    - When **checked**, the account is allowed to send email packages and share folders with users that do not exist in the defined Address Book.
    - When **unchecked**, the account is allowed to send email packages and share folders only with users that exist in the defined Address Book.

  This account setting overrides the business unit and/or global Address Book policy setting for collaboration.

  Users will be able to share folders when **Allow collaboration with non-Address Book recipients** is checked only if the `SharedFolders.AllowCollaboration` server configuration parameter is set to **true**. For more information on changing server configuration parameters, refer to *View and change server configuration parameters on page 334*. If the Address Book feature is disabled, this setting does not affect user collaboration.

Address Book use cases

This topic outlines different Address Book use cases. In all of the use cases, only Local and LDAP sources are enabled.

Local user sends an AdHoc package to recipients

Assuming that the global Address Book settings are taken into consideration and the user's Address Book setting is Default. At the global configuration level a LDAP source is enabled and **Allow collaboration with non-Address Book recipients** is selected. A user can send AdHoc packages to all entries available in their address book (the LDAP source) and to external users.

When the user creates a draft message, a list of the available Address Book entries that can be selected as recipients is displayed. In this case, the user can also type in email addresses that do not exist in their configured Address Book.

Note If the total number of recipients (resolved email addresses) exceeds the value of the `AddressBook.Limit.AdHoc.Recipients` configuration option, the send operation will fail and an error message will be displayed to the user stating the reason for the failure.

Note If **Allow collaboration with non-Address Book recipients** is not selected, the user is not allowed to type email addresses in recipient fields - only Address Book contacts can be selected.

For additional information, refer to the *SecureTransport Web Client User Guide*.

Local user in a Business Unit sends AdHoc package to recipients

Assuming that the business unit level Address Book settings are taken into consideration and the account level Address Book settings are Default. At the business unit level, the Address Book settings are **Custom** and a Local source is assigned. **Allow collaboration with non-Address Book recipients** is not enabled for the business unit. When the Local source type is **All Business units**, the user can send emails to all SecureTransport accounts that have a configured email address. When the Local source type is **User's own Business unit**, the user can send emails only to accounts in the same business unit. The list of available contacts will be displayed to the user. The user will not be able to type external emails as this type of collaboration is not allowed for their business unit.

Local user shares a folder

Apart from sending emails, an account can also share folders to a defined list of recipients. The server configuration option `SharedFolders.AllowCollaboration` must be set to **true**. This is valid for global and account levels. If an account is in a business unit, the business unit option **Allow collaboration with non-Address Book recipients** should be enabled. Then Address Book settings can regulate whether a folder will or will not be shared with the list of recipients.

When the user starts to type in shared folder collaborators, a list of Address Book contact names are displayed to the end user. Each of the listed contacts can be selected as collaborators.

If the **Allow collaboration with non-Address Book recipients** option is enabled for the user, the user will be able to type in email addresses and specify them as collaborators.

If the number of specified collaborators exceeds the `AddressBook.Limit.AdHoc.Recipients` option value, the share operation will fail and an error message will be displayed to the user stating the reason for the failure.

Note If the **Allow people to view collaborators** option is enabled, only collaborators' email addresses will be displayed, display names will not be shown even if collaborators are part of the address book.

For additional information, refer to the *SecureTransport Web Client User Guide*.

LDAP user shares a folder

For LDAP sources, the SecureTransport administrator can specify one of the already defined LDAP domains or the **Logged In (current domain)** option. When an LDAP user is logged in to SecureTransport and the **Logged In** domain is set as an Address Book source, all members of the same LDAP server will be displayed as Address Book entries.

Assuming that account settings are taken into consideration - LDAP users are mapped to an account template which has custom Address Book settings. An LDAP source **Logged In (current domain)** is assigned to the account template. The LDAP user which is mapped to the account template can share folders with all other users and groups defined in the same LDAP domain.

Note If the total number of resolved collaborators exceeds the value of the `AddressBook.Limit.AdHoc.Recipients` configuration option, the share operation will fail and an error message will be displayed to the user stating the reason for the failure.

Sending an AdHoc package through REST API

The end-user REST API is able to identify Address Book entries by their IDs. When sending a message through REST API recipients can be: user entry IDs, group entry IDs, and email addresses. Email addresses will be filtered out if they are not part of the Address Book and the user is not allowed collaboration with external users. The list of user's available contacts is displayed by making a REST call to `/addressBook`.

Create message draft:

```
POST/api/v1.4/mailbox/messages
POST DATA:
to=petya@sofia-pso.tumbleweed.com&cc=<group_entry_
id>&bcc=<user_entry_
id>&subject=from+api&message=sample+message&security=ANONYMOUS_
LINK&question=+What+is+the+name+of+your+best+friend+from+childh
ood%3F+&answer=&expiration=86400
RESPONSE:
http://10.232.3.211:8080/api/v1.4/mailbox/messages
200 OK
Headers
Response body
{
 "info": "Draft was successfully created",
 "messageId": "6a5467d4a084445687b1cdc0cd11f3a3"
}
```

Consider the following use case:

A user creates a draft and specifies an email address in the **To** recipient category, an Address Book entry ID in **CC** and an Address Book group ID in **BCC**. When the user sends the message the following happens:

- If the email address specified in **To** does not belong to any entry of the current user's Address Book, the message will be sent to that email address only if the **Allow collaboration** option is enabled for this user.
- The group ID specified in **CC** will be searched in user's Address Book and if it is found, it will be resolved to all users email addresses that belong to this group.
- The user ID specified in **BCC** will be searched in user's Address Book and if it is found, it will be resolved to corresponding entry email address.
- The message will be sent to all resolved email addresses.

Note If the total number of resolved collaborators exceeds the value of the `AddressBook.Limit.AdHoc.Recipients` configuration option, the share operation will fail and an error message will be displayed to the user stating the reason for the failure.

Note If **Allow Collaboration** is turned off and the user tries to send mail to an external user, the user will be removed from recipients list and a warning message will be displayed in the server log.

Sharing a folder through REST API

When sharing a folder through REST API collaborators can be user entry IDs, group entry IDS, and email addresses. Email addresses will be filtered out if they are not part of the Address Book and the user is not allowed collaboration with external users. The list of user's available contacts is displayed by making a REST call to `/addressBook`.

Example request for sharing folder (`sharedFolder`) with an Address Book user entry:

```
PUT/api/v1.4/shares/sharedFolder
DATA:
'{
 "users": [
  "<user_entry_id>"
 ],
 "shareRights": 7,
"notifications": 1,
"ownerNotifications": 1,
"showCollaboratorsToAll": true
}'
```

Note Only collaborators' email addresses will be listed when getting sharing metadata for the specified folder, display names will not be showed even if collaborators are part of the Address Book.

Note If the total number of resolved collaborators exceeds the value of the `AddressBook.Limit.AdHoc.Recipients` configuration option, the share operation will fail and an error message will be displayed to the user stating the reason for the failure.

Address Book REST API

This topic provides Address Book administrator and end user REST API examples and provides REST API support information for Address Book recipients.

Administrator REST API

The Address Book functionality introduces the following REST API endpoints which allow configuring and assigning address book data sources on all configuration levels:

- Address Book Global Registry REST API

  GET `/addressBookSources` - Lists all available Address Book data sources.

  GET `/addressBookSources/<ABSourceId>` - Gets an Address Book data source.

  POST `/addressBookSources/<ABSourceId>` - Modifies properties: enabled, group.

- Business Unit Address Book REST API

  GET `/businessUnit/<name>/addressBookSources` - Gets list of the assigned Address Book sources

  PUT `/businessUnit/<name>/addressBookSources/` - <set of Address Book entries in the body> - replaces all assigned to a business unit.

  GET `/businessUnit/<name>/addressBookSources/<ABSourceId>` - Gets an Address Book source assigned to a business unit.

  POST `/businessUnit/<name>/addressBookSources/<ABSourceId>` - Assigns and updates an Address Book source to a business unit. The body may contain the parent group.

  DELETE `/businessUnit/<name>/addressBookSources/<ABSourceId>` - Removes an Address Book source from those assigned to a business unit.

- Account Address Book REST API

  GET `/accounts/<name>/addressBookSources/` - Gets list of assigned Address Book sources to an account.

  PUT `/accounts/<name>/addressBookSources/` - <set of Address Book entries in the body> - replaces all assigned.

  GET `/accounts/<name>/addressBookSources/<ABSourceId>` - Gets an Address Book source assigned to an account.

  POST `/accounts/<name>/addressBookSources/<ABSourceId>` - Assigns or updates an Address Book source to an account. The body may contain the parent group property.

  DELETE `/accounts/<name>/addressBookSources/<ABSourceId>` - Removes an Address Book source from the list of assigned sources to an account.

  GET `/accounts/<accountName>/addressBook/contacts` - Get address book contacts of a specified account entity.

  POST `/accounts/<accountName>/addressBook/contacts` - Create address book contacts to a specified account entity.

  DELETE `/accounts/<accountName>/addressBook/contacts/<id>` - Delete an address book contact of an account entity.

  PUT `/accounts/<accountName>/addressBook/contacts/<id>` - Update an address book contact of an account entity.

- Address Book global settings

  GET `/addressBookSources/settings/` - Gets `AddressBook.AllowCollaboration` and `AddressBook.Enabled` server configurations.

POST `/addressBookSources/settings/<name>` - Updates `AddressBook.AllowCollaboration` server configuration.

End User REST API

The following endpoints are exposed for the end user REST API:

- **GET** `/addressBook` - Lists all available contacts for the given account.
- **GET** `/addressBook?displayName=test%K&mail=test@axway.com&group=Axway&type=user&orderBy=email&limit=10&offset=0` - Search for all entries matching all parameters together. The search is case sensitive for all sources except for LDAP sources. The logical operation between the parameters is AND.
- **GET** `/addressBook?searchFor=test&limit=10&offset=0` - Searches all entries which `displayName`, email or group contain the given value. The search is case insensitive. The search phrase is looked up anywhere in the fields `displayName`, email, and group (wildcard search), except for LDAP sources where it is looked up in the beginning of these fields only (wildcard is at the end of the search phrase). The logical operation between the fields is OR.

Parameter Name	Description	Type
`displayName`	Optional. Case sensitive exact search by `displayName` field.	String
`mail`	Optional. Case sensitive exact search by `mail` field.	String
`parentGroup`	Optional. Case sensitive, exact search by `parentGroup` field.	String
`type`	Optional. Case insensitive search by `type`. The `type` can be either a user or a group.	String
`orderBy`	Optional. Specify the sort order of the result. Could be `displayName` or `email`.	String
`limit`	Optional. Page size for pagination. If not specified a default value will be set.	Integer
`offset`	Optional. Starting index of the pagination. If not specified a default value will be set.	Integer
`searchFor`	Optional. Case insensitive, wildcard search between `group`, `displayName`, and `mail` parameters.	String

- **GET** `/addressBook/count` - Returns the count of all Address Book entries from all enabled Address Book sources assigned to the current user.

- **GET** `/addressBook/<id>` - Gets an address book entry by ID.
- **GET** `/addressBook/myself` - Returns map of Address Book settings with their corresponding values. This includes the following:
    - `addressBookEnabled` – A flag indicating if Address Book is enabled.
    - `addressBookCollaborationAllowed` – A flag indicating if collaboration with external recipients is allowed for the given user.
    - `addressBookMinSearchLength` – Specifies the minimum search string length when performing wildcard search for Address Book entries.
    - `addressBookMaxDisplayEntries` – Specifies the maximum limit of Address Book entries displayed in a single page.
    - `addressBookMaxDisplayPages` – Specifies the maximum limit of Address Book pages in a single request.

Address Book entry schema:

```
AddressBookEntry {
    id: addressBookSourceId:addressBookEntryId,
    displayName,
    mail,
    parentGroup,
    type,
    customAttr1,
    customAttr2,
    customAttr3
}
```

Note:

The following methods are also available for the `/addressBook` end user REST resource. They are not implemented in SecureTransport 5.3.6 and earlier.

- **POST** `/addressBook/` - Creates a new user specific entry. The Address Book provider to which entry will be created will generate `ABEntry` ID. The Address Book source is defined by `parentGroup` property mapping.
- **PUT** `/addressBook/<id>` - Updates the user specific entry.
- **DELETE** `/addressBook/<id>` - Deletes the user specific entry.

For each one of them the SecureTransport server returns a response status **403 Forbidden**.

Support for Address Book recipients

Whenever an Address Book user inputs a group or display name either as an email recipient or shared folder collaborator, SecureTransport is able to resolve the recipients email addresses. If the given group members count exceed a predefined value, an error message is shown to the end user and the

operation is aborted. A server configuration option is exposed for setting that value - `AddressBook.Limit.AdHoc.Recipients`.

Whenever an end user sends a mail or shares a folder with group or display name in its recipients list, the REST call includes the Address Book entry IDs which corresponds to the names. If the list of recipients does not include Address Book entries, the REST call contains the actual email addresses. This allows the SecureTransport back end to handle cases with recipients not in the Address Book as well as recipients defined in the Address Book, or to determine if the Address Book is enabled or not.

When the client sends a request with Address Book entry IDs and email addresses, all package recipients are stored in the `.stfs` attributes and later when a list operation is performed the response contains the stored recipients (for example, IDs as well as email addresses). If sharing a folder, the SecureTransport will always return the email addresses.

Operations 3

Operations is a top-level menu in SecureTransport and SecureTransport Edge.

Use the Operations menu of the Axway SecureTransport Administration Tool to initiate operator-driven actions, planned daily tasks, statistics, monitors, and responses to events.

Administration Tool server runs as an HTTPS server using port 444 by default. The Administration Tool server starts automatically each time a UNIX-based system starts. The installer includes a startup item in the system's `rc` directory tree. On a Windows system, the server runs as a service.

You can also manually start the Administration Tool server by executing the following commands:

- For UNIX, go to `<FILEDRIVEHOME>/bin` and run `./start_admin`.
- For Windows, go to `<FILEDRIVEHOME>\bin\` and run `start_admin.com`.

Note The Administration Tool server reserves ports 8004 and 8005 to be used by the Java-based component. These ports are used strictly for internal communication within the Administration Tool and cannot be used for other purposes.

Operations menu overview

When you log in to the SecureTransport Administration Tool, the *Dashboard* tab is displayed. From the main navigation menu, click *Operations* to access the following pages:

- **Server Control** – Used to manage your FTP, HTTP, AS2, SSH, PeSIT, Transaction Manager (TM), and Monitor servers. For details, see *Server control* on page 259.
- **Cluster Management** – Used to view and maintain SecureTransport Servers in cluster. For details, see *Standard Cluster* on page 351 and *Enterprise Cluster* on page 368.
- **Server Usage Monitor** – Used to monitor by user class the FTP, SSH and HTTP, as well as the protocol bandwidth consumed. For details, see *Server usage monitor* on page 299.
- **File Tracking** – Displays a log of the status and attributes of each file transfer. Also, used to display detailed information about a transfer and to cancel or resubmit a transfer. SecureTransport Server only. For details, see *File Tracking* on page 302.
- **Server Log** – Used to view, search, and filter the logs from the SecureTransport Server. For details, see *Server log* on page 322.
- **Audit Log** – Used to view, compare, and export log entries that SecureTransport records when any change is made to the SecureTransport configuration. For details, see *Audit log* on page 326.
- **Server Configuration** – Used to view, change, export, and import server configuration settings. For details, see *Server configuration* on page 333.

- **Support Tool** – Used to collect information about SecureTransport and its host operating system and save it in a support information file that you can send to Axway Global Support. For details, see *Support tool* on page 345.

Server control

***Operations* on page 258** > **Server Control**

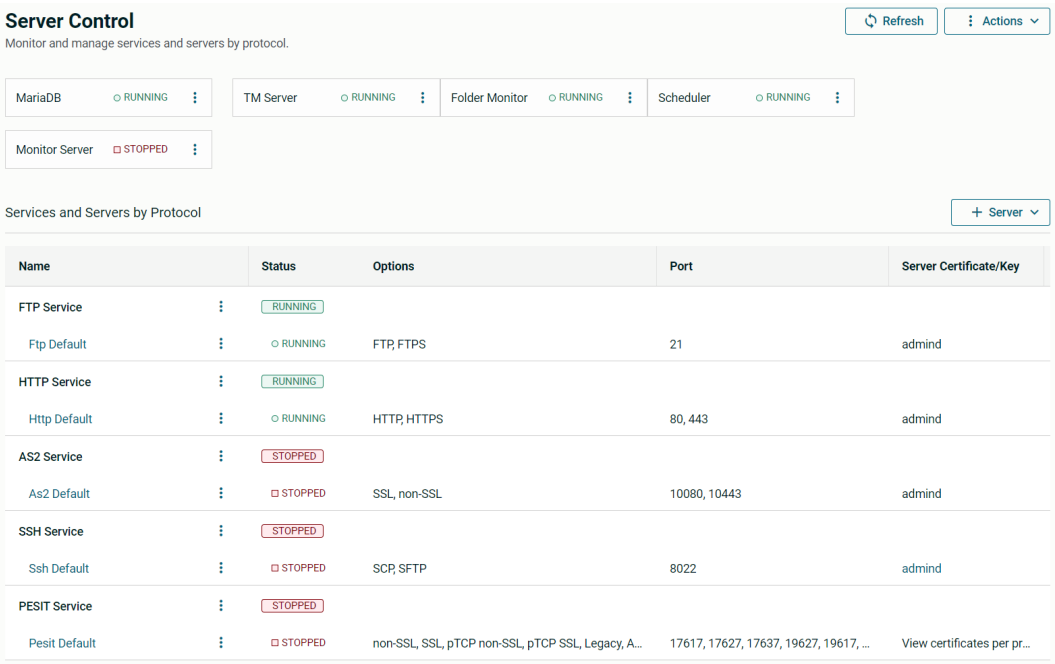
The *Server Control* page is the entry point for SecureTransport administrators upon login. Here you view and manage all protocol servers, the Transaction Manager server, Folder Monitor, Scheduler, and the Monitor server of your system.

Server control is available on both SecureTransport Servers and SecureTransport Edge Servers.

The following sections provide detailed server control information:

- *Server Control: Protocol servers* on page 260
- *Server Control: Database* on page 261
- *Server Control: Transaction Manager server* on page 261
- *Server Control: Folder Monitor* on page 261
- *Server Control: Scheduler* on page 261
- *Server Control: Monitor server* on page 262
- *Service status indicators* on page 262
- *Select a preferred cryptographic service provider* on page 263
- *Server Control on SecureTransport Edge* on page 263

The following image presents the *Server Control* page on a SecureTransport Server.



Server Control: Protocol servers

On the *Server Control* page, you can manage the following protocol services (also called daemons): FTP, HTTP, AS2, SSH and PeSIT. You can add as many servers (also called listeners) as you need for these services. In the *Services and Servers by Protocol* table, each protocol service is presented in a dedicated section along with its servers and their status, basic configuration options, listener ports, and key alias.

You can manage the protocol services and servers on three different levels.

Top-level Actions menu

Click on the **Actions** menu in the upper right corner to start and stop all services, including all protocol servers, the Transaction Manager, Folder Monitor, Scheduler and Monitor Server. You can also perform graceful shutdown of the SecureTransport Server or Edge node.

Service menu

Click on ⋮ (*more* icon) next to a service to open the contextual menu. You can add a protocol server, start, stop, and perform graceful shutdown of all servers for this service.

Server menu

Click on ⋮ (*more* icon) next to a server to open the contextual menu. You can start and stop an individual server, open its settings, and delete it. A server can be deleted only after it is stopped.

The following topics provide more details on adding and editing servers:

- *Add an FTP Server on page 264*
- *Add an SSH server on page 277*
- *Add an HTTP server on page 267*
- *Manage an AS2 server on page 272*
- *Add a PeSIT server on page 280*
- *Advanced service configuration and memory allocation on page 287*

Disabling protocol servers

If you don't want a server to start together with all other services, you can disable it. When you disable any of the servers through the Administration Tool, the script used to run the server from the command line is renamed with the `.disable` extension. To disable a server, open its settings and deselect all *Enable* checkboxes. The original script name is restored when the server is re-enabled.

For example, if you disable the FTP server, the script `start_ftpd` is renamed to `start_ftpd.disable`. Once you re-enable and start the FTP server through the Administration Tool, the script name is restored to `start_ftpd`. All utility scripts are located in the `<FILEDRIVEHOME>/bin` directory.

Server Control: Database

The Database tile on the *Server Control* page displays the database status (running or stopped), and allows easy access to the database settings via ⋮ (*more* icon).

Server Control: Transaction Manager server

The Transaction Manager (TM) server connects to the ports that are specified in the network zones. The dedicated tile displays its status and allows you to start or stop it, or to shut it down gracefully via ⋮ (*more* icon). There is no Transaction Manager tile on SecureTransport Edge Servers.

For more information, see *Communication across Transaction Manager, protocol, and proxy servers* on page 227.

Server Control: Folder Monitor

The Folder Monitor service is available only on SecureTransport Servers and is dependent on the value of the `FolderMonitor.enable` server configuration option. The dedicated tile displays its status and allows you to start or stop it via ⋮ (*more* icon).

For more information, see *Folder Monitor transfer sites* on page 557.

Note The `FolderMonitor.enable` server configuration option does not start or stop the service but only enables and disables it. When the Transaction Manager is running and the Folder Monitor service is disabled, the service runs in the background but does not pull any files. The Folder Monitor service is functional only when it is enabled and started, and the Transaction Manager is running.

Server Control: Scheduler

The Scheduler service is available only on SecureTransport Servers and is dependent on the value of the `Scheduler.enable` server configuration option. The dedicated tile displays its status and allows you to start or stop it via ⋮ (*more* icon).

For more information, see *Scheduled downloads and tasks* on page 674.

Note The `Scheduler.enable` server configuration option does not start or stop the service but only enables and disables it. When the Transaction Manager is running and the Scheduler service is disabled, the service runs in the background but does not schedule events. The Scheduler service is functional only when it is enabled and started, and the Transaction Manager is running.

Server Control: Monitor server

The Monitor server uses the `monitord` monitoring service to perform periodical checks and identify if the SecureTransport Servers are functional or not. The dedicated tile displays its status and allows you to start or stop it via ⋮ (*more* icon).

For more information, see *Monitor Server* on page 293.

Service status indicators

The following table lists the possible statuses that services can display on different occasions.

Status	Description	Used during:
Starting (displayed in green)	The service is in the process of starting.	Service startup
Running (displayed in green)	The service is running normally.	Normal operation
Stopping (displayed in red)	The service is in the process of stopping.	Service stop
Stopped (displayed in red)	The service is stopped.	Disabled or stopped services
Shutdown pending	An HTTP or FTP service suspension is pending.	Scheduled suspensions
Shutdown scheduled	The protocol service is scheduled to gracefully shut down in the near future. The service is running but does not allow transfers. Users can still connect and receive messages.	Graceful shutdown

Select a preferred cryptographic service provider

The default cryptographic provider in SecureTransport is BouncyCastle. This is determined by service-specific server configuration options, where the default value is `true`. The BouncyCastle cryptographic library is FIPS-certified and contains more algorithms and cipher suites than the Sun library. For maximum security, we recommend using the default settings.

In a case where you do not need FIPS, you can set the server configuration option for a particular service to `false` to speed up system performance. By doing so, Sun becomes the preferred provider, and BouncyCastle is used as a fallback. As Sun is not FIPS-compliant, FIPS mode must first be disabled in order to change the preferred provider from BouncyCastle to Sun.

The following options control the preferred cryptographic service provider, per service:

Service	Configuration option
FTP	`Ftp.preferBouncyCastleProvider`
HTTP	`Http.preferBouncyCastleProvider`
AS2	`As2.preferBouncyCastleProvider`
SSH	`Ssh.preferBouncyCastleProvider` Alternatively, use the **Prefer BouncyCastle Crypto provider** checkbox in the SSH Daemon settings.
PeSIT	`Pesit.preferBouncyCastleProvider`
TM	`TM.preferBouncyCastleProvider`

Server Control on SecureTransport Edge

On the *Server Control* page on a SecureTransport Edge Server there is no *TM Server* tile because the Transaction Manager does not run on Edge Servers. However, you can configure the port for the Edge proxy server. The proxy port is used by the SecureTransport Server to handle outgoing connections passed through the Edge. The default port number is 1080.

1. Click ⋮ (*more* icon) on the **Proxy Server** tile to open the settings.
2. Enter the port number in the **Port** field.
3. Click **Save**.

Note By default, SecureTransport Edge uses a cipher strength of `AES 256` for communication between its protocol servers and the TM server on the SecureTransport Servers. To change the enabled cipher suites, edit the

`TransactionManager.Listeners.Ssl.enabledCipherSuites` server configuration option. For valid values, see the cipher suites listed for transfers using AS2, FTPS, HTTPS, and PeSIT over Secure Socket in Supported ciphers and cipher suites (login required).

Add an FTP Server

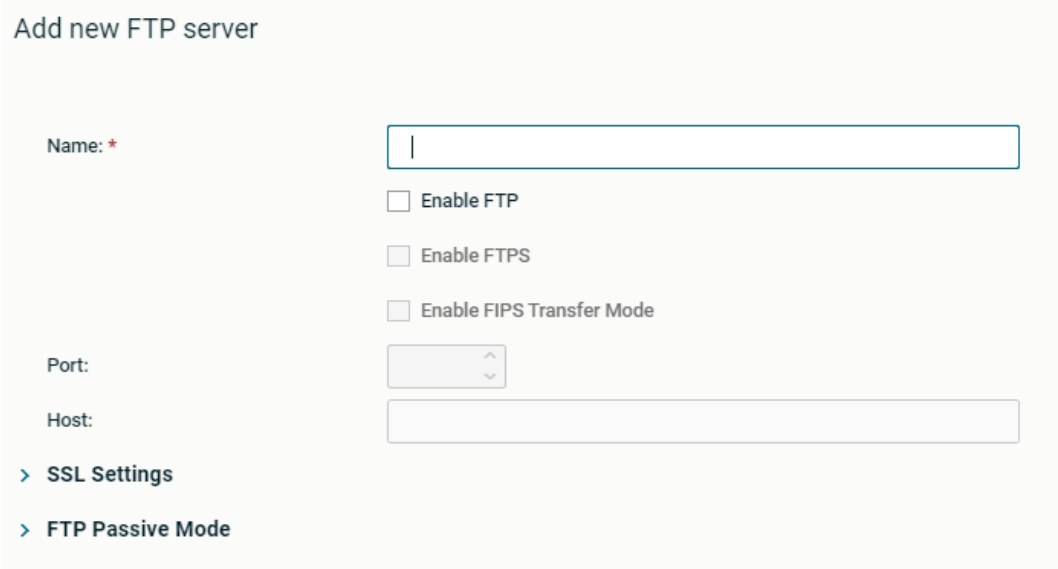
To add a new FTP server:

1. Click **Operations > Server Control**.
2. On the *Server Control* page, click the **+ Server** button and select **FTP Server** from the dropdown list.

   Alternatively, click ⋮ (*more* icon) next to *FTP Service*, and click **Add FTP server**.
3. On the *Add new FTP server* pane, configure the server settings. For detailed descriptions of all configuration settings, see *FTP Server Settings* on page 264.
4. Click **Save** to create the server.
   Clicking **Cancel** discards all changes and returns you to the *Server Control* page.

FTP Server Settings

The configuration settings for an FTP server are separated into three sections: General, SSL Settings, and FTP Passive mode.



General Settings

Start by configuring the general connection properties for your new FTP server.

Field	Description
Server Name	Enter a unique name for your server.
Enable FTP	Select this checkbox to enable unencrypted FTP connections. The checkbox must be selected in order to enable FTPS (FTP over SSL).
Enable FTPS	Select this checkbox to enable FTPS. **Caution** When uploading files to SecureTransport Server via FTPS, the FTP client is required to indicate that the transfer is complete by sending a `close_notify` message. If a `close_notify` message is not sent by the client, the file transfer will fail. You can prevent transfer failure by setting the configuration option `Ftp.Ssl.requireCloseNotify` to `false` but this would make the server susceptible to TLS truncation attacks.
Enable FIPS	Enabling FIPS limits the FTP server to only use FIPS 140-2 compliant ciphers for encrypted connections. For more information, see *FIPS transfer mode* on page 1065. By selecting this option, the **Enabled FIPS Ciphers** field becomes editable.
Port	Enter a port number for connection to FTP or FTPS server.
Host	Enter the IP address of your external FTP (or FTPS) host server. Leave this option blank if you do not need an external host.

SSL Settings

Click to expand the section and configure all aspects of your FTP server SSL/TLS security: choose a certificate to use, the key and trust manager factory algorithms, allowed TLS versions and cipher strings, etc. The SSL settings become editable after you select the **Enable FTP** and **Enable FTPS** checkboxes.

Field	Description
SSL Key Alias	Select an SSL Key Alias from the dropdown list.

Field	Description
Enabled Protocols	Specify the allowed TLS versions. Use comma to separate different versions. Default value for newly created FTPS servers after updating to SecureTransport 5.5-20210930: `TLSv1.2, TLSv1.3`. Default value for existing FTPS servers: `TLSv1, TLSv1.1, TLSv1.2`. For instructions on how to enable TLSv1.3 protocol support, refer to the *SecureTransport5.5 Security guide*. **Note** `TLS v1.3` no longer supports DSA certificates. If a server is configured to use DSA certificates and `TLS v1.3` is enabled on both the client and the server, the handshake will fail.
SSL Key Algorithm	Specify the key manager algorithm to use. The default value is `SunX509`.
SSL Protocol	Enter the used SSL protocol group: `SSL` or `TLS`. The default value is `TLS`.
SSL Trust Algorithm	Enter the SSL Trust Algorithm. The default value is `SunX509`.
Client Certificate	Specify whether the server will require certificate authentication. Choose from the following options: • `Disabled` – no certificate authentication is required • `Required` – the client must authenticate using a certificate • `Optional` – the client can authenticate either using a certificate or a password
Enabled Ciphers	From the dropdown list, select the cipher suites to be used with your server. The default list is dictated by the `Ftp.Listeners.Ssl.enabledCipherSuites` configuration option. For more information, see Supported ciphers and cipher suites (login required).
Enabled FIPS Ciphers	**Note** This setting become editable after you enable FIPS transfer mode. From the dropdown list, select the cipher suites to be used with your FTPS server in FIPS mode. The default list of enabled FIPS-compliant cipher strings is determined by the `Ftp.FIPS.Listeners.Ssl.EnabledCipherSuites` configuration option. For more information, see FIPS-compliant ciphers and cipher suites (login required).

FTP Passive Mode

In FTP Passive mode, the client initiates the control connections to the server. The FTP server replies with an IP address and port number for the client to use to establish the data connection.

Field	Description
Base Port	Enter the passive mode base port. The default value is `0`, meaning that SecureTransport will use a random port for FTP passive mode transfers.
Number of Ports	The passive mode port range. For example, if **Base Port** is set to 5000 and **Number of Ports** is set to 100, the passive connection will be made on a port between 5000 and 5100.

Edit FTP Server Settings

You can change any of the FTP server property values. The are only few specifics:

- You cannot change the name of the "Ftp Default" server from the Administration Tool.
- The server name can only be changed when the server is stopped.

To update an FTP server configuration:

1. On the *Server Control* page, click on the name of the FTP server you want to edit.

   A pane with the your server settings appears on the right side.
2. Make your changes.
3. Click **Save**.

Add an HTTP server

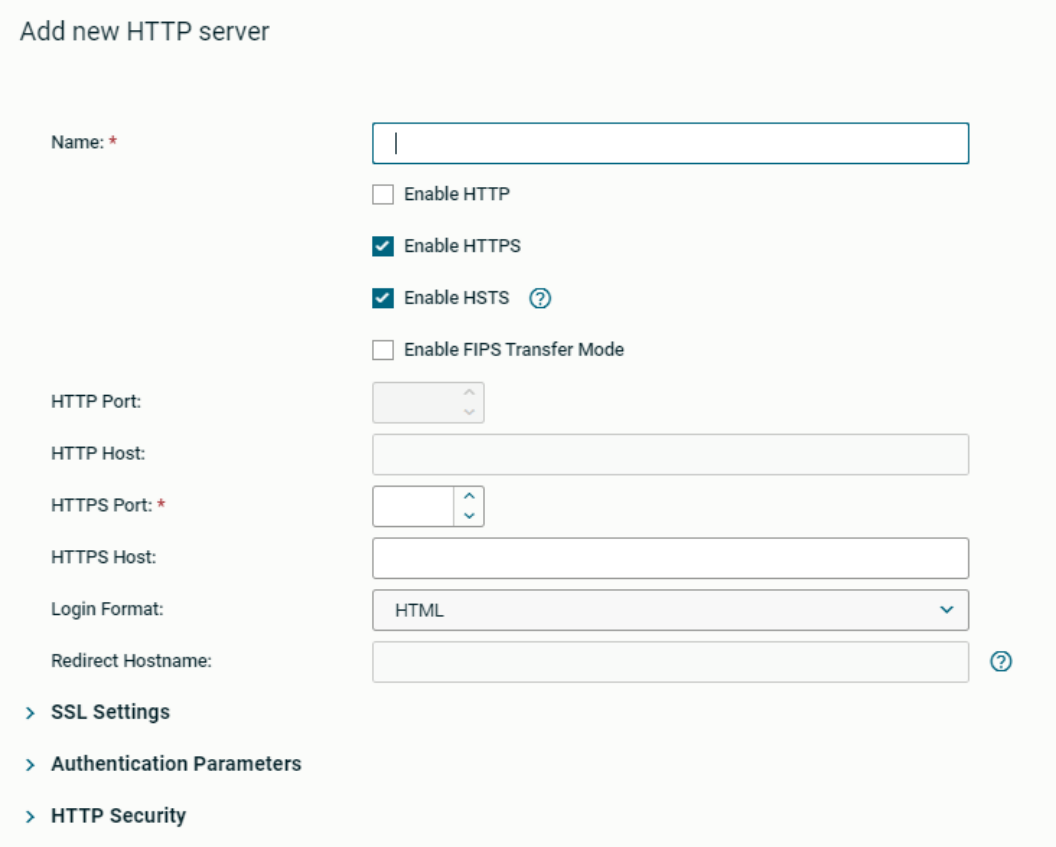
To add a new HTTP server:

1. Click **Operations > Server Control**.
2. On the *Server Control* page, click the **+ Server** button and select **HTTP Server** from the dropdown list.

   Alternatively, click ⋮ (*more* icon) next to *HTTP Service*, and click **Add HTTP server**.
3. On the *Add new HTTP server* pane, configure the server settings. For detailed descriptions of all configuration settings, see *HTTP Server Settings* on page 268.
4. Click **Save** to create the server.
   Clicking **Cancel** discards all changes and returns you to the *Server Control* page.

HTTP Server Settings

The configuration settings for an HTTP server are separated into four sections: General, SSL Settings, Authentication Parameters, and HTTP Security.



General Settings

Start by configuring the general connection properties for your new HTTP server.

Field	Description
Server Name	Enter a unique name for your server.
Enable HTTP	Select to enable HTTP transfers.
Enable HTTPS	Select to enable HTTPS transfers.

Field	Description
Enable HSTS	Select to enable HSTS to always send the "`Strict-Transport-Security`" HTTPS response header to redirect plain HTTP connections to HTTPS. With this functionality, two dedicated server configuration options for HSTS are added: • `Http.Security.Hsts.enabled` - Enable or disable HSTS for the HTTP server. Serves the same purpose as the checkbox. Possible values are `true` or `false` with `true` being the default. It is only editable from the *Server Configuration* page. • `Http.Security.Hsts.max-age` - HSTS header maximum age attribute value for the HTTP server measured in seconds. The default value is 6 months which is the equivalent of `15768000 seconds`.
Enable FIPS	Select to enable FIPS transfer mode for HTTPS connections. By selecting this option, the **Enabled FIPS Ciphers** field becomes editable.
HTTP Port	Enter a port number for your HTTP listener.
HTTPS Port	Enter the port number of your HTTPS listener.
Login Format	Select the authentication format for end-user login: • HTML – for user login using the ST Web Client login form • BA – basic authentication • ERR – must use config/auth agents • PREAUTH – config/auth agents + HTML login page in case of failed login
Redirect hostname	Enter a redirect host name or IP address. When you set this value, all requests to the ST Web Client, subsequent to the first one, will be bound to that hostname. Use this option in the case where a DNS switch occurs to avoid requests getting split across different nodes.

SSL Settings

Click to expand the section and configure all aspects of your HTTP server SSL/TLS security: choose a certificate to use, allowed TLS versions, allowed cipher strings in FIPS and non-FIPS mode, etc. The SSL settings become editable after you select the **Enable HTTP** and **Enable HTTPS** checkboxes.

Field	Description
Client Certificate	Specify whether the server will require certificate authentication. Choose from the following options: • `Disabled` – no certificate authentication is required • `Required` – the client must authenticate using a certificate • `Optional` – the client can authenticate either using a certificate or a password
SSL Key Alias	Select an SSL Key Alias from the dropdown list, for example, `HTTPd`.
SSL Protocol	Enter the used SSL protocol group: `SSL` or `TLS`. The default value is `TLS`.
Enabled SSL Protocols	Specify the allowed TLS versions. The default value for newly created HTTPS servers after updating to SecureTransport 5.5-20210930: `TLSv1.2, TLSv1.3`. Default value for existing HTTPS servers: `TLSv1, TLSv1.1, TLSv1.2`. For instructions on how to enable TLSv1.3 protocol support, refer to the *SecureTransport5.5 Security guide*.
Enabled Ciphers	From the dropdown list, select the cipher suites to be used with your server. The default list is dictated by the `Http.Ssl.EnabledCipherSuites` configuration option. For more information, see Supported ciphers and cipher suites (login required).
Enabled FIPS Ciphers	**Note** This setting become editable after you enable FIPS transfer mode. From the dropdown list, select the cipher suites to be used with your HTTP server in FIPS mode. The default list of enabled FIPS-compliant cipher strings is determined by the `Http.FIPS.Ssl.EnabledCipherSuites` configuration option. For more information, see FIPS-compliant ciphers and cipher suites (login required).

Authentication Parameters

In this section you specify the allowed authentication parameters and their maximum size.

Field	Description
Allowed Authentication Parameters	Enter the allowed HTTP Authentication parameters, separated by a semi-colon (;).
Allowed Authentication Parameters Max Size	Enter the allowed HTTP Authentication parameters maximum size in bytes.

HTTP Security

Setting up HTTP security headers in the SecureTransport server configuration protects you against many common attacks, including cross-site request forgery, cross-site scripting, clickjacking, etc. They can be set at the global level via server configuration options and customized for each HTTP server via the following fields:

Field	Description
Content Security Policy	Sets the `Content-Security-Policy` header.
XSS Protection	Sets the `X-XSS-Protection` header.
Content Type Options	Sets the `X-Content-Type-Options` header.
Referrer Policy	Sets the `Referrer-Policy` header. Accepted values: `no-referrer, no-referrer-when-downgrade, origin, origin-when-cross-origin, same-origin, strict-origin, strict-origin-when-cross-origin, unsafe-url`
Expect CT	Sets the `Expect-CT` (certificate transparency) header. Accepted values: `max-age=<age>; enforce; report-uri=<uri>`. The `enforce` and `report-uri` directives are optional.
SameSite cookie attribute	Sets the SameSite attribute of the `Set-Cookie` HTTP response header.

Note For more information on the HTTP security headers and their defaults in SecureTransport, refer to the Security Guide.

Edit HTTP server settings

You can change any of the HTTP server property values following the steps below.

Caution The server name can only be changed when the server is stopped.

1. On the *Server Control* page, click on the name of the HTTP server you want to edit.

   A pane with the your server settings appears on the right side.
2. Make your changes.
3. Click **Save**.

Manage an AS2 server

AS2 (Applicability Statement 2) is a specification about how to transport data securely and reliably over the Internet. Security is achieved by using digital certificates and encryption. The AS2 specification describes how to exchange business data securely and reliably using HTTP as an underlying transport. The data is packaged using standard MIME content types so you can use XML, EDI, binary data, and any other data describable in MIME. Message security (authentication, confidentiality) is implemented using S/MIME. Message reliability is enabled through the use of MDNs. Nonrepudiation and Nonrepudiation of Receipt are business and legal concepts that build upon the security and reliability components in AS2.

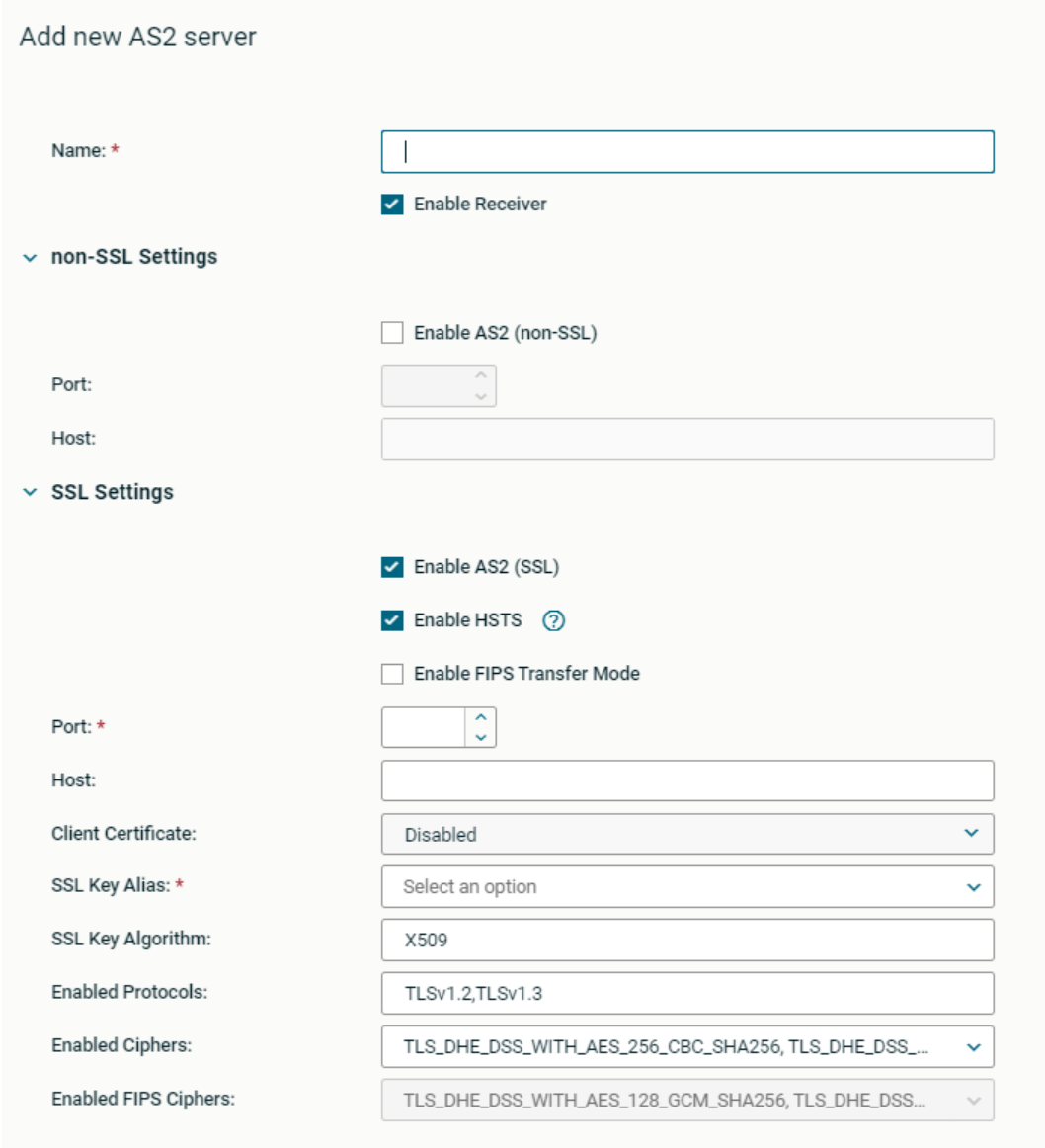
If an AS2 license is available, enable the AS2 server. In cluster setup, specify the AS2 settings on both SecureTransport Server and SecureTransport Edge.

Add an AS2 Server

To add a new AS2 server:

1. Click **Operations > Server Control**.
2. On the *Server Control* page, click the **+ Server** button and select **AS2 Server** from the dropdown list.

   Alternatively, click ⋮ (*more* icon) next to *AS2 Service*, and click **Add AS2 server**.
3. On the *Add new AS2 server* pane, configure the server settings. For detailed descriptions of all configuration settings, see *AS2 Server Settings* on page 273.
4. Click **Save** to create the server.
   Clicking **Cancel** discards all changes and returns you to the *Server Control* page.



AS2 Server Settings

The following table presents all parameters and expected values associated with your new AS2 server.

Field	Description
Server Name	Enter a unique name for your server.
Enable Receiver	Select to enable receiving of your current AS2 server.
non-SSL Settings	

Field	Description
Enable AS2 (non-SSL)	Select to enable insecure AS2 transfers with your current AS2 server. By selecting this option, the **non-SSL Port** and **non-SSL Host** options become editable. **Note** To enable AS2 without SSL, you must create an SSL encryption entry for a user class with SSL encryption optional. See *Manage SSL access* on page 781.
non-SSL Port	Enter the port number of your non-secure AS2 server.
non-SSL Host	Enter the host address of your non-secure AS2 server.
SSL Settings	
Enable AS2 (SSL)	Select to enable secure AS2 transfers with your current AS2 server. By selecting this option, the remaining options become editable.
Enable HSTS	Select to enable HSTS to always send the "`Strict-Transport-Security`" HTTPS response header to redirect plain HTTP connections to HTTPS. With this functionality, two dedicated Server Configuration options for HSTS are added: • `As2.Security.Hsts.enabled` - Enable or disable HSTS for the AS2 server. Serves the same purpose as the check-box. Possible values are: **true** or **false**. It is only editable from the *Server Configuration* page. The default value is **true**. • `As2.Security.Hsts.max-age` - HSTS header maximum age attribute value for the AS2 server measured in seconds. The default value is 6-months which is equivalent to **15768000 seconds**.
Enable FIPS	Select to enable *FIPS transfer mode* on page 1065 for AS2 connections. By selecting this option, the **Enabled FIPS Ciphers** field becomes editable.
SSL Port	Enter the port number of your AS2 server.
SSL Host	Enter the host address of your AS2 server.
Client Certificate	If you are using AS2 via SSL, this dropdown list presents the different client certificate enforcement options. • `Disabled` – no certificate authentication is required • `Optional` – the client can additionally authenticate using a certificate. However, if the client presents an incorrect certificate, authentication fails. • `Required` – the client must authenticate using a certificate

Field	Description
SSL Key Alias	Select an SSL Key Alias from the dropdown list, for example, `admind`.
Key Exchange Algorithms	Enter the Key Algorithm (the default is `SunX509`) .
Enabled SSL Protocols	Specify the allowed TLS versions. Use comma to separate different versions Default value for newly created AS2 servers after updating to SecureTransport 5.5-20210930: `TLSv1.2, TLSv1.3`. Default value for existing AS2 servers: `TLSv1, TLSv1.1, TLSv1.2`. For instructions on how to enable TLSv1.3 protocol support, refer to the *SecureTransport5.5 Security guide*. **Note** `TLS v1.3` no longer supports DSA certificates. If a server is configured to use DSA certificates and `TLS v1.3` is enabled on both the client and the server, the handshake will fail.
Enabled Ciphers	From the dropdown list, select the cipher suites to be used with your server. The default list is dictated by the `As2.Listeners.Ssl.enabledCipherSuites` configuration option. For more information, see Supported ciphers and cipher suites (login required).
Enabled FIPS Ciphers	**Note** This setting becomes editable after you enable FIPS transfer mode. From the dropdown list, select the cipher suites to be used with your server in FIPS mode. The default list of enabled FIPS-compliant cipher strings is determined by the `As2.FIPS.Listeners.Ssl.EnabledCipherSuites` configuration option. For more information, see FIPS-compliant ciphers and cipher suites (login required).

For information about more AS2 settings, see *Configure the AS2 server settings* on page 80.

Edit AS2 server settings

You can change any of the AS2 server property values following the steps below.

Caution The server name can only be changed when the server is stopped.

1. On the *Server Control* page, click on the name of the AS2 server you want to edit.

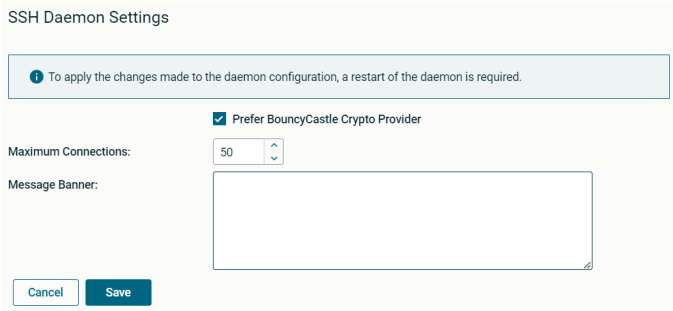
   A pane with the your server settings appears on the right side.
2. Make your changes.
3. Click **Save**.

Modify the SSH daemon configuration

Administrators with permissions to access to the *Server Control* and *SSH Settings* page (configurable through the Administrative role settings) can modify several settings to alter the behavior of the daemon. Those include:

- select the preferred cryptographic provider
- limit the maximum number of SSH connections
- set up a login message banner

To access those settings , click ⋮ (*more* icon) next to the **SSH Service** and then click on **Daemon settings**.
A pane with the *SSH Daemon Settings* appears on the right side.



Once you have finished editing the daemon settings, click **Save**. Then, restart the SSH service for the changes to take effect.

Note The options to edit the daemon configuration are also exposed as REST API endpoints: `/daemons/{name}`.

Select BouncyCastle as preferred provider

The default cryptographic provider is BouncyCastle, with Sun as the alternative. For more information, see *Select a preferred cryptographic service provider* on page 263.

Limit the Maximum Number of Connections

Type the maximum number of SSH clients that can simultaneously connect to the SSH server. If the maximum number of connections is reached, SecureTransport prevents any further connections before the user is authenticated. If you increase the value of **Maximum Connections**, you must also increase the value of `SSH_JAVA_MEM_MAX` in the `<FILEDRIVEHOME>/bin/start_sshd` script.

The script uses the `SSH_JAVA_MEM_MAX` value to set the maximum heap size for the Java Virtual Machine (JVM). The SSH server allocates memory in the heap for each connection (and frees it when the connection is closed). To avoid SSH service interruptions when no memory is available in the Java heap, you must configure `SSH_JAVA_MEM_MAX` to the desired number of **Maximum Connections** (as defined in the previous step) multiplied by 10,000 Kibibytes. However, in the script

file, you must convert this result to Megabytes (10,000 Kibibytes is equivalent to 10.24 Megabytes). For example, 500 concurrent connections would require a value of 500 * 10,000 KiB = 5,000,000 KiB. This is equivalent to 5120 Megabytes, so enter this in the script file as:

```
SSH_JAVA_MEM_MAX="5120M"
```

Use `M` to specify Megabytes. Do not insert a space between the number and the `M`.

Set up a Message Banner

The SSH daemon can be modified to present a custom banner to the users when they log in via SSH. The content of this banner is entered in the **Message Banner** text field. For example, you can enter a legal notice, a disclaimer, or a welcome message.

The following topics describe how to bind SSH and SSHD to the same port number and debug SSH issues:

- *Debug SSH issues* on page 1058

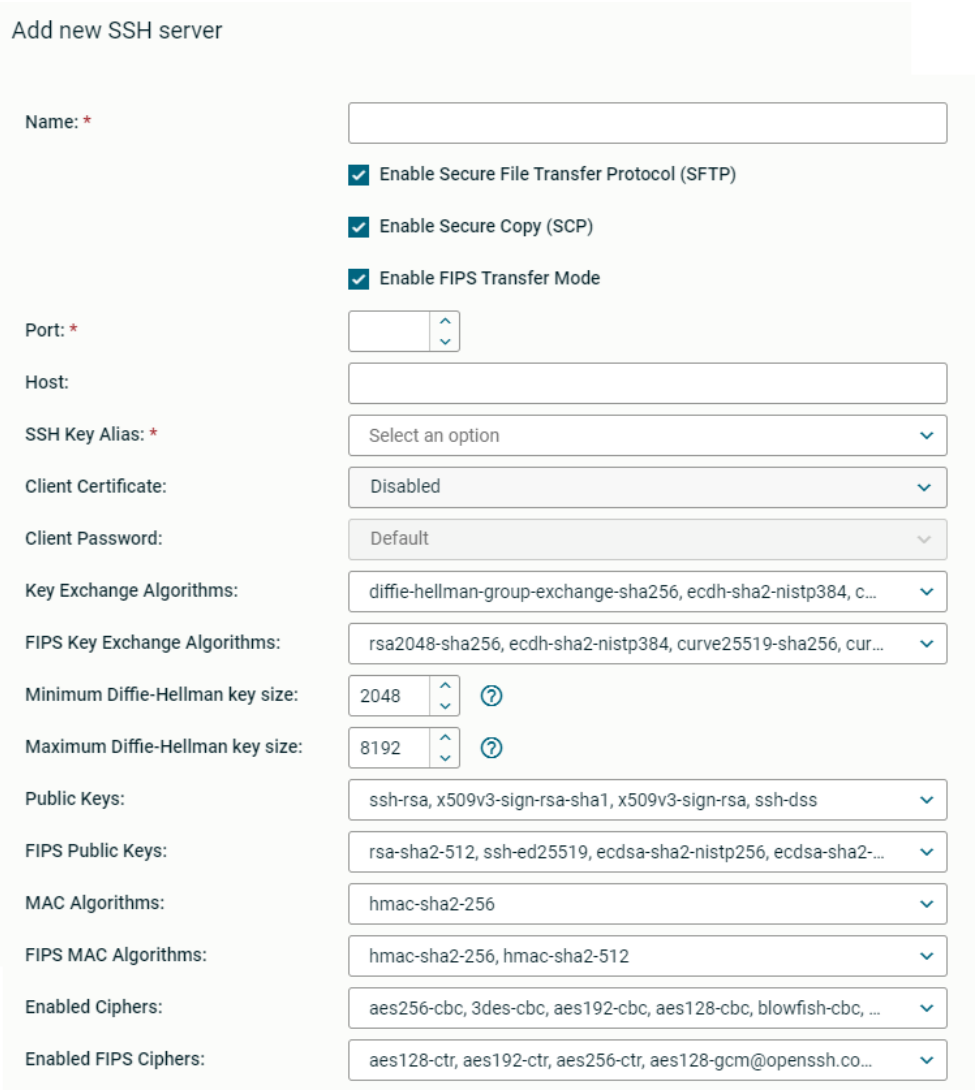
Add an SSH server

To add a new SSH server:

1. Click **Operations > Server Control**.
2. On the *Server Control* page, click the **+ Server** button and select **SSH Server** from the dropdown list.

   Alternatively, click ⋮ (*more* icon) next to *SSH Service*, and click **Add SSH server**.
3. On the *Add new SSH server* pane, configure the server settings. For detailed descriptions of all configuration settings, see *SSH Server Settings* on page 278.
4. Click **Save** to create the server.
   Clicking **Cancel** discards all changes and returns you to the *Server Control* page.

SSH Server Settings



The following table presents all parameters and expected values associated with your new SSH server.

Field	Description
Server Name	Enter a unique name for your server.
Enable SCP	Select to enable SCP (Secure Copy) support with transfers using your current SSH server.
Enable SFTP	Select to enable SFTP transfers using your current SSH server.

Field	Description
Enable FIPS	Select to enable FIPS transfer mode for SSH connections. By selecting this option, the following fields become editable: • FIPS Exchange Algorithms • FIPS Public Keys • FIPS MAC Algorithms • Enabled FIPS Ciphers
Port	Enter a port number for connection to your SSH server.
Host	Enter the host address of your SSH server.
SSH Key Alias	Select an SSH Key Alias from the dropdown list, for example, `admind`.
Client Certificate	Specify whether the server will require certificate authentication. Choose from the following options: • `Disabled` – no certificate authentication is required • `Required` – the client must authenticate using a certificate • `Optional` – the client can authenticate either using a certificate or a password
Client Password	When **Client Certificate** is set to `Required`, you can also specify the authentication methods that the SSH server offers to the SSH client: • `Default` – the server sends the authentication methods set in **Authentication > Login Settings > End-user login options** • `Disabled` – the server sends only "publickey"
Key Exchange Algorithms	From the dropdown list, select Diffie-Hellman exchange hashing algorithms to be used with your server, for example: `diffie-hellman-group14-sha1,diffie-hellman-group-exchange-sha256`.
FIPS Exchange Algorithms	From the dropdown list, select the KEX algorithms to be used with your server in FIPS mode. By default, this field is populated with all FIPS-compliant KEX algorithms supported by SecureTransport. For the complete list, see FIPS-compliant ciphers and cipher suites (login required).
Minimum key size for Diffie-Hellman exchange algorithms group	Enter the minimum exchange key size in bits. The default is `2048`.

Field	Description
Maximum key size for Diffie-Hellman exchange algorithms group	Enter the maximum exchange key size in bits. The default value is `8192`. See Key Exchange Algorithms (login required).
Public Keys	From the dropdown list, select the public keys to be used with your server.
FIPS Public Keys	From the dropdown list, select the keys to be used with your server in FIPS mode. For more information, see FIPS-compliant ciphers and cipher suites (login required).
MAC Algorithms	From the dropdown list, select the MAC algorithm that warrants the integrity of the transfer using your current SSH server.
FIPS MAC Algorithms	From the dropdown list, select the MAC algorithms to be used with your SSH server in FIPS mode. For more information, see FIPS-compliant ciphers and cipher suites (login required).
Enabled Ciphers	From the dropdown list, select the cipher suites to be used with your server. For more information, see Supported ciphers and cipher suites (login required).
Enabled FIPS Ciphers	From the drop-down list, select the cipher suites to be used with your server in FIPS mode. For more information, see FIPS-compliant ciphers and cipher suites (login required).

Edit SSH Server Settings

You can change any of the SSH server property values following the steps below.

Caution The server name can only be changed when the server is stopped.

1. On the *Server Control* page, click on the name of the SSH server you want to edit.

   A pane with the your server settings appears on the right side.
2. Make your changes.
3. Click **Save**.

Add a PeSIT server

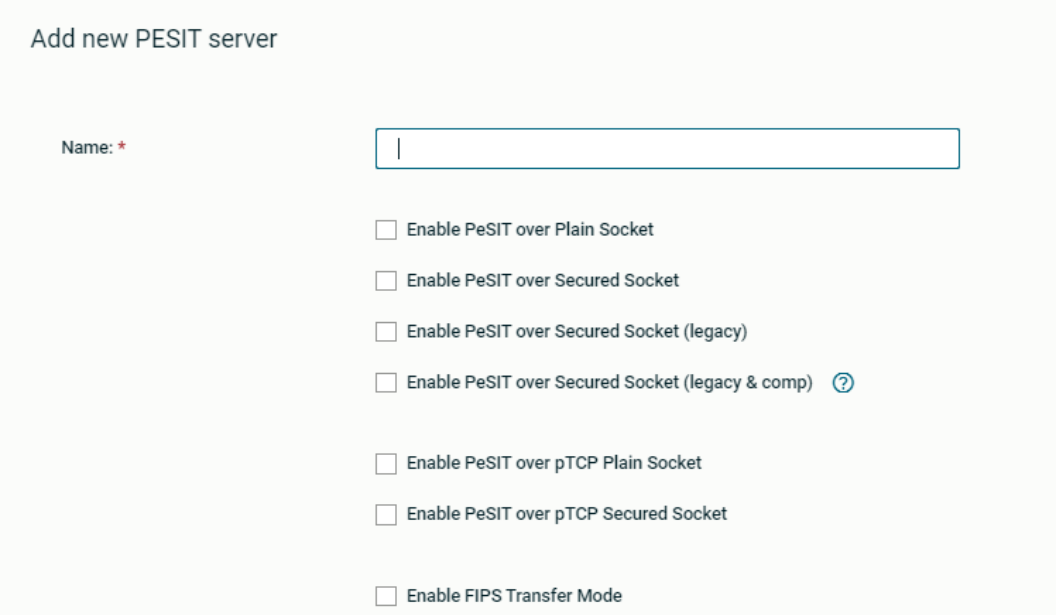
To add a new PeSIT server:

1. Click **Operations > Server Control**.
2. On the *Server Control* page, click the **+ Server** button and select **PeSIT Server** from the dropdown list.

   Alternatively, click ⋮ (*more* icon) next to *PeSIT Service*, and click **Add PeSIT server**.
3. On the *Add new PeSIT server* pane, configure the server settings. For detailed descriptions of all configuration settings, see *PeSIT Server Settings* on page 281.
4. Click **Save** to create the server.
   Clicking **Cancel** discards all changes and returns you to the *Server Control* page.

PeSIT Server Settings

The configuration settings for a PeSIT server are separated into several sections. You need to specify only those related to the desired connection layer.

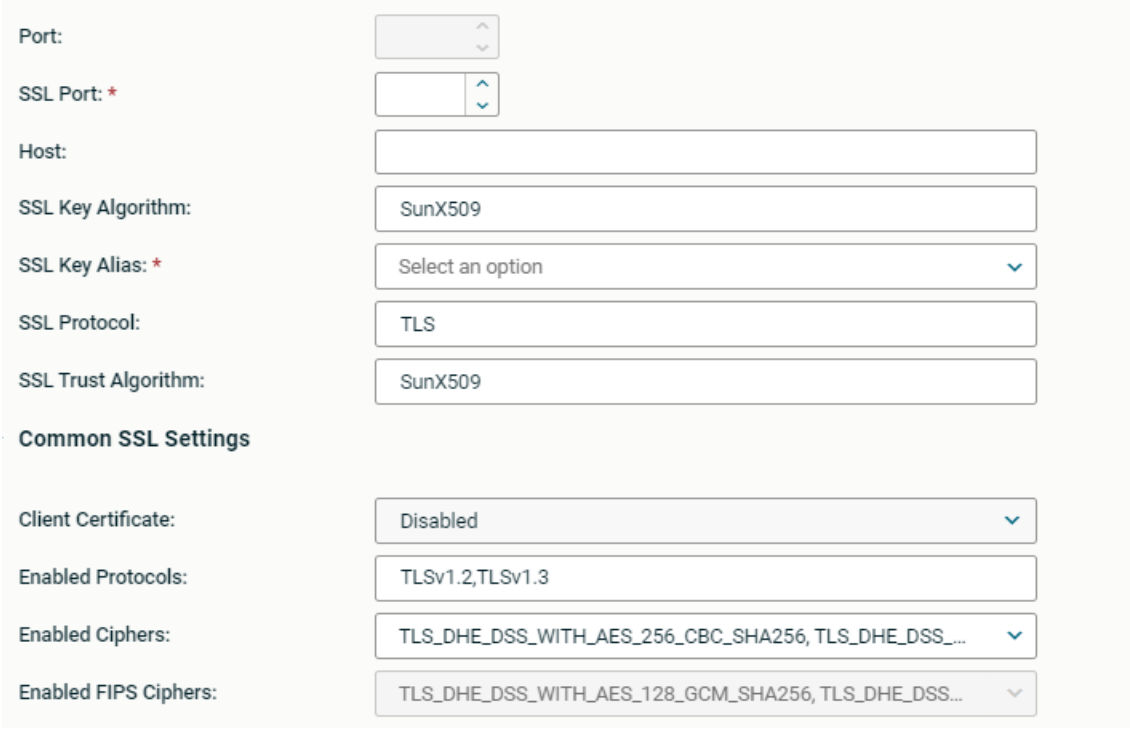


Enable PeSIT over Plain Socket

Selecting this checkbox will enable non-secure PeSIT transfers over TCP/IP. This requires that you specify a port and a host.

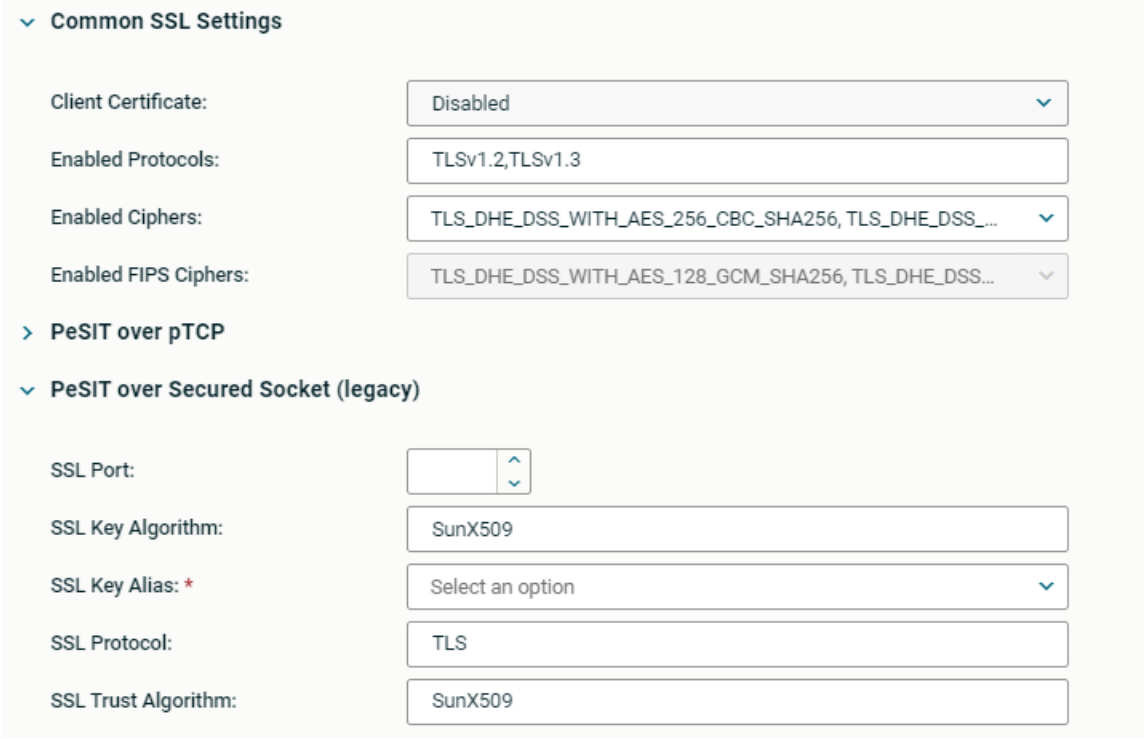
Enable PeSIT over Secured Socket

Selecting this checkbox will enable secure PeSIT transfers over TCP/IP SSL/TLS. This requires that you specify a port and an SSL Key Alias. The rest of the SSL settings are optional and include specifying the allowed SSL protocol, its version and cipher suites to use, as well as the key and trust algorithms. For more information, see *Common Fields and Settings* on page 285.



Enable PeSIT over Secured Socket (legacy)

Select this checkbox to enable transfers with remote partners using SSL Legacy. This option is mostly used for CIT transfers from Transfer CFT that runs on version 3.2 or older. In this case, the connection settings are located under **PeSIT over Secured Socket (legacy)**. You can also change the allowed TLS version and cipher strings: go to the *Common SSL Settings* section and set **Enabled Protocols** and **Enabled Ciphers**. For detailed information, see *Common Fields and Settings* on page 285.



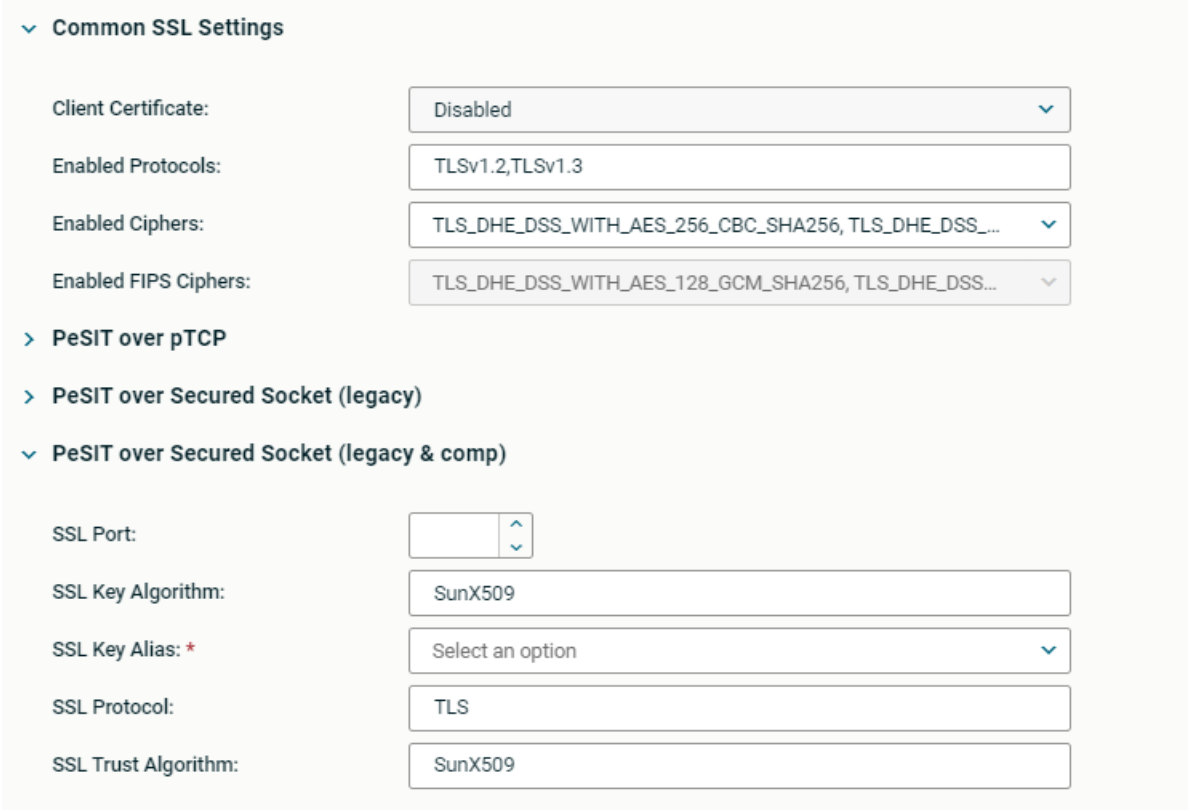
Enable PeSIT over Secured Socket (legacy § comp)

Just like the option above, this one is primarily used when the client is Transfer CFT. By selecting the checkbox, you enable the automatic detection of the used SSL/TLS mode (Legacy or Comp) when SecureTransport acts as a server. Information about the detected mode is logged in the server log under **Level > Debug**.

The PeSIT listener used for communication with partners in both TLS Comp and TLS Legacy modes is configured using the following server configuration parameters:

- `Pesit.Autodetect.Tls.Mode.Enabled` - enables or disables the listener
- `Pesit.Autodetect.Tls.Mode.Port` - specifies the port number of the listener
- `Pesit.Listeners.Autodetect.Tls.Mode.keyAlgorithm` - specifies the key algorithm
- `Pesit.Listeners.Autodetect.Tls.Mode.keyAlias` - specifies the key alias of the listener
- `Pesit.Listeners.Autodetect.Tls.Mode.protocol` - specifies the protocol of the listener
- `Pesit.Listeners.Autodetect.Tls.Mode.trustAlgorithm` - specifies the trust algorithm

All those settings can be customized at the server level using the settings under **PeSIT over Secured Socket (legacy & comp)** in the server configuration. Selecting an *SSL Key Alias* for the server is mandatory.

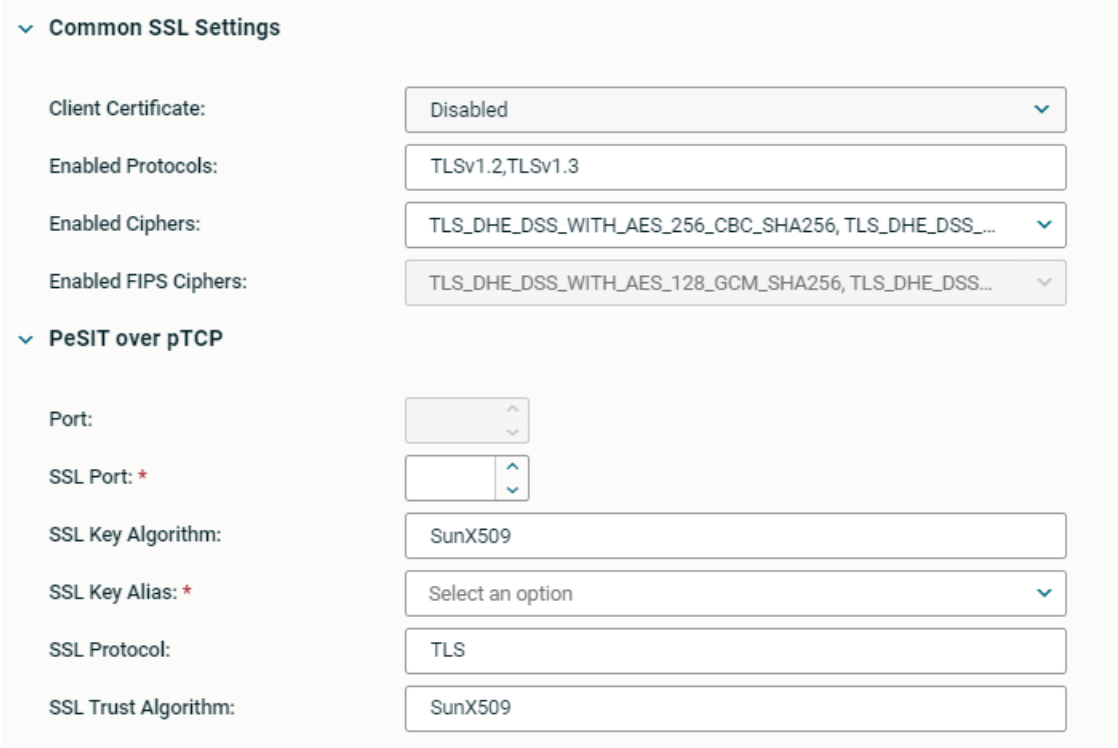


Enable PeSIT over pTCP Plain Socket

Select this checkbox to enable non-secure PeSIT transfers over pTCP. Then, type a port number for non-secure connection in the **Port** field under **PeSIT over pTCP**.

Enable PeSIT over pTCP Secure Socket

Select this checkbox to enable secure PeSIT transfers over pTCP SSL/TLS. In this case, you configure the SSL settings under **PeSIT over pTCP**.
To change the allowed TLS version and cipher strings: go to the *Common SSL Settings* section and set **Enabled Protocols** and **Enabled Ciphers**.



The delay and retries for the PeSIT handshake are configurable via the following server configuration options:

- `Pesit.pTCP.maxRetryCount` specifies the number of times SecureTransport can re-attempt the check that all of the configured PeSIT pTCP connections are open. It applies to all server-initiated transfers (inbound and outbound). The default value is 500 times.
- `Pesit.pTCP.retryDelayInterval` specifies the time interval in milliseconds between consecutive check attempts for open PeSIT pTCP connections. The default value is 20 milliseconds.

Enable FIPS Transfer Mode

Select this checkbox to enable FIPS transfer mode for Secure Socket connections. The allowed cipher suites can be selected from the **Common SSL Settings > Enabled FIPS Ciphers** dropdown list.

Common Fields and Settings

The following table describes the fields and settings that are used across the PeSIT server configuration.

Fields	Description
Port	Enter a port number for non-secure connection to the PeSIT server.

Fields	Description
SSL port	Enter the port number for secure connection to the PeSIT server.
Host	Enter the IP address of your external PeSIT host server. Leave this option blank if you do not need an external host.
SSL Key Algorithm	Specify the key manager algorithm to use. The default is `SunX509`.
SSL Key Alias	Select an SSL Key Alias from the dropdown list, for example, `PeSITd`.
SSL Protocol	Specify the used SSL protocol group: `SSL` or `TLS`. The default value is `TLS`.
SSL Trust algorithms	Specify the SSL Trust Algorithm. The default value is `SunX509`.
Client Certificate	
Enabled SSL Protocols	
Enabled Ciphers	From the dropdown list, select the cipher suites to be used with your server. The default list is dictated by the `Pesit.Listeners.Ssl.enabledCipherSuites` configuration option. For more information, see Supported ciphers and cipher suites (login required).
Enabled FIPS Ciphers	

Edit PeSIT Server Settings

You can change any of the PeSIT server property values following the steps below.

Caution The server name can only be changed when the server is stopped.

1. On the *Server Control* page, click on the name of the PeSIT server you want to edit.

   A pane with the your server settings appears on the right side.
2. Make your changes.
3. Click **Save**.

Advanced service configuration and memory allocation

SecureTransport 5.5 allows administrators to configure the Transaction Manager and protocol services using configuration files. The underlying concept is to supply unified daemon configuration by adding a dedicated start scripts configuration per server daemon.

Note The start scripts configuration overrides any other configuration.

Start scripts global configuration

The SecureTransport start scripts configuration is located in the *STStartScriptsConfig* file, which is created during installation in the `FILEDRIVEHOME/conf` directory. The location of this file is pre-configured in the `${FILEDRIVEHOME}/bin/common.sh` script and can be changed in two ways:

- by editing the `ST_START_SCRIPTS_CONF_PATH` parameter in *common.sh*;
- by setting `ST_START_SCRIPTS_CONF_PATH` as a permanent operating system environment variable. This method takes precedence over the setting in the *common.sh* script.

In the start scripts configuration file, you specify configuration settings using predefined properties, which are name/value pairs in the following format:

`[PROTOCOL_NAME]_[OPTION_NAME]=[value]`

Possible values for `PROTOCOL_NAME`: TM, SSH, FTP, HTTP, ADMIN, AS2, PESIT

Possible values for `OPTION_NAME`:

- `JAVA_MEM_MIN` - sets minimum memory of the `PROTOCOL_NAME` daemon's JVM heap size.
- `JAVA_MEM_MAX` - sets maximum memory of the `PROTOCOL_NAME` daemon's JVM heap size.
- `JAVA_OPTS` - sets java options to the `PROTOCOL_NAME` JVM.

Unlike the start_* scripts, the properties defined in the *STStartScriptsConfig* file must be provided one per line. The order of definitions does not matter. You cannot assign multiple values to the same variable in a single line.

An example of invalid syntax is:

```
TM_JAVA_OPTS="-DStreaming.numberOfConnections=10 -
DAdvancedRouting.maxRuntimes=32 $TM_JAVA_OPTS"
```

An example of valid syntax is:

```
TM_JAVA_OPTS="-DStreaming.numberOfConnections=10 $TM_JAVA_OPTS"
TM_JAVA_OPTS="-DAdvancedRouting.maxRuntimes=32 $TM_JAVA_OPTS"
```

Changing the start scripts configuration

You can change the start scripts configuration in two ways:

- Edit the existing file, *STStartScriptsConfig*. Always make a copy of any configuration file before editing.
- Create a new configuration file and set the `ST_START_SCRIPTS_CONF_PATH` to its location.

You must restart all SecureTransport services for the configuration changes to take effect.

The following is an example configuration for the protocol daemon servers. Read through it to see what could be specified.

```
# Example: Setting file encoding to UTF-8
# SSH_JAVA_OPTS="-Dfile.encoding=utf8"
# Default values

# SSH
SSH_JAVA_MEM_MIN=1024M
SSH_JAVA_MEM_MAX=2048M
SSH_JAVA_OPTS="-Dfile.encoding=utf8"

# HTTP
HTTP_JAVA_MEM_MIN=512M
HTTP_JAVA_MEM_MAX=1024M
HTTP_JAVA_OPTS="-Dfile.encoding=utf8"

# FTP
FTP_JAVA_MEM_MIN=64M
FTP_JAVA_MEM_MAX=512M
FTP_JAVA_OPTS="-Dfile.encoding=utf8"

# AS2
AS2_JAVA_MEM_MIN=64M
AS2_JAVA_MEM_MAX=512M
AS2_JAVA_OPTS="-Dfile.encoding=utf8"

# Admin
ADMIN_JAVA_MEM_MIN=512M
ADMIN_JAVA_MEM_MAX=1024M
ADMIN_JAVA_OPTS="-Dfile.encoding=utf8"

# PeSIT
PESIT_JAVA_MEM_MIN=64M
PESIT_JAVA_MEM_MAX=512M
PESIT_JAVA_OPTS="-Dfile.encoding=utf8"

# monitord service
```

```
MONITORD_JAVA_MEM_MIN=256M
MONITORD_JAVA_MEM_MAX=512M
MONITORD_JAVA_OPTS="-Dfile.encoding=utf8"

# xml_import and xml_export scripts
XML_JAVA_MEM_MIN=64M
XML_JAVA_MEM_MAX=256M
XML_JAVA_OPTS="-Dfile.encoding=utf8"

# TM
TM_JAVA_MEM_MIN=2048M
TM_JAVA_MEM_MAX=4096M
TM_JAVA_OPTS="-DStreaming.numberOfConnections=10"
```

Note You can add different shell script commands to the start script. Act with caution as your input will be executed each time the start script runs.

For more configuration options, refer to the "Additional SecureTransport configuration" topic in the *SecureTransport Capacity Planning Guide*. This document is available on the Axway Documentation portal to logged in users only.

TM options

In addition, a few TM options available in the `start_tm_console` are configurable through the `STStartScriptsConfig` file. The following code snippet shows them with example values:

```
# TM-specific options
disableHeapDumpOnOutOfMemoryError=true
generate_heap_dump=true
GC_LOGGING=true
NumberOfGCLogFiles=30
GCLogFileSize=5000K
TM_JAVA_OPTS="-XX:MaxDirectMemorySize=512M $TM_JAVA_OPTS"
TM_JAVA_OPTS="-DAdvancedRouting.isCamelEnpointSingleton=true $TM_JAVA_OPTS"
TM_JAVA_OPTS="-DAdvancedRouting.camelRuntimesPerCpuCore=2 $TM_JAVA_OPTS"
TM_JAVA_OPTS="-DAdvancedRouting.minRuntimes=16 $TM_JAVA_OPTS"
TM_JAVA_OPTS="-DAdvancedRouting.maxRuntimes=32 $TM_JAVA_OPTS"
```

Certificates for daemon configuration

You can use the following operating system environment variables to import certificates and use them later in SecureTransport for different purposes, including daemon configuration.

- `ST_CA_PATH` - Path to the Certificate Authority
- `ST_CA_ALIAS` - Certificate Authority alias
- `ST_CERT_PATH` - Local certificate path

- `ST_CERT_PASS` - Password for the local certificate
- `ST_CERT_ALIAS` - Local certificate alias

When ready with the configuration, restart the Admin service. Certificates will be imported on admin startup.

Graceful shutdown

This topic describes the concept and specifics of performing graceful shutdown of the different SecureTransport components.

- *Graceful shutdown of protocol services* on page 290
- *Graceful shutdown of the Transaction Manager* on page 292
- *Graceful shutdown of a SecureTransport Server node* on page 292
- *Graceful shutdown of a SecureTransport Server node* on page 292

Graceful shutdown of protocol services

The current subtopic contains general information and instructions on how to perform graceful shutdown on any of the available protocol services, except where noted otherwise. The option to gracefully shut down protocol services is available on both SecureTransport Server and Edge.

Graceful shutdown is an option to initiate a shutdown of any or all protocol services without abrupt cancellation of the currently ongoing client-initiated transfer (CIT) sessions. Once initiated, the graceful shutdown waits for the specified timeout period before stopping the selected service. This period defines the time allowed for existing transfers to be completed before shutting down. You can set its value (in seconds) per protocol service via the dedicated server configuration option:

Protocol	Server configuration option	Default value
FTP	`Ftpd.GracefulShutdownTimeout`	`86400s`
HTTP	`Http.GracefulShutdownTimeout`	`30s` – if not previously modified (see note)
AS2	`As2.GracefulShutdownTimeout`	`86400s`
SSH	`Ssh.GracefulShutdownTimeout`	`86400s`
PeSIT	`Pesit.GracefulShutdownTimeout`	`86400s`

Note With HTTP, the `Http.GracefulShutdownTimeout` server configuration option was available prior to introducing the graceful shutdown functionality. In case its default value has been modified, the input value is stored.

Note that during the graceful shutdown period:

- Existing CITs are allowed to complete within the specified timeout period.
- Any new attempts for file operations are rejected. This includes not only file uploads and downloads but also directory listing, deleting or renaming files, as well as deleting or creating directories.

Perform protocol service graceful shutdown

Before you proceed with graceful shutdown initiation, you must stop the Monitor Server.

To initiate this process using the Administration Tool, navigate to **Operations > Server Control**, click ⋮ (*more* icon) on the respective protocol service, and select **Graceful Shutdown**.

You can also perform this action using a console command in the following format: `stop_ <protocol_service> -g -timeout <interval_in_seconds>`.

- the `-g` parameter can be used to stop the service gracefully while using the specific service default configuration timeout;
- the `-timeout <interval_in_seconds>` parameter is optional and can be used in conjunction with `-g` to offset graceful shutdown with the defined interval (different than the default one, as defined in the server configuration option for the protocol). If you omit this parameter, the graceful shutdown will be performed using the respective configuration option value.

For example, if you want to initiate graceful shutdown after 60 seconds using the console command for the respective protocol service, enter:

- `stop_ftpd -g -timeout 60` for the FTP service
- `stop_httpd -g -timeout 60` for the HTTP service
- `stop_as2d -g -timeout 60` for the AS2 service
- `stop_sshd -g -timeout 60` for the SSH service
- `stop_pesitd -g -timeout 60` for the PeSIT service

The option to gracefully shut down a protocol service is also exposed as a REST API resource. For more information, refer to the Swagger REST API documentation.

Note With HTTP, when you initiate a graceful shutdown during active file uploads with ST Web Client, these uploads will be processed until the current chunk upload is completed. The chunk size is defined via the `uploadChunkSize` parameter in the ST Web Client configuration. By default, its value is `100 MB`, which means that in all cases, uploads of files with sizes smaller than 100 MB will be completed during the graceful shutdown. With larger files, uploads might be completed as well; or could be stopped depending on the current chunk upload.

Graceful shutdown logging

The Server Log displays information about active connections during an initiated graceful shutdown. For better visibility, a dedicated server option is introduced: `GracefulShutdown.Logging.Interval`. By default, its value is `60s`, which means that active transfer information will be logged once every 60 seconds. Note that the graceful shutdown logging interval applies to all protocol services.

Graceful shutdown of the Transaction Manager

Before you proceed with graceful shutdown initiation, you must stop the Monitor Server.

Graceful shutdown is a feature that allows you to have a planned Transaction Manager stop without abrupt cancellation of:

- current server-initiated transfers (SITs)
- post-routing, post-transformation, and post-processing actions
- Advanced Routing actions (all routes and their respective route steps)

  For example, let's say you initiate graceful shutdown of the Transaction Manager during the execution of a route with a Publish To Account step to another subscription folder. When the Transaction Manager moves a file to the other account, the file is processed and the respective subscription actions are triggered because they are considered a continuation of the original Advanced Routing action, which started before initiating graceful shutdown.

Once initiated, the graceful shutdown waits for the specified timeout period before stopping the Transaction Manager. This period defines the time allowed for existing transfers to be completed before shutting down. You can set its value (in seconds) via the dedicated server configuration option `TransactionManager.GracefulShutdownTimeout`.

To perform graceful shutdown of the Transaction Manager, navigate to **Operations > Server Control**, click ⋮ (*more* icon) on the TM Server tile, and select **Graceful Shutdown**. The Transaction Manager will be stopped after the timeout expires.

You can also perform this action using a console command in the following format: `stop_tm -g -timeout <interval_in_seconds>`. For example, `stop_tm -g -timeout 60` will shut down the Transaction Manager after 60 seconds.

It is recommended that you configure the timeout period for no less than 30 seconds.

The option to gracefully shut down the Transaction Manager is also exposed as a REST API resource. For more information, refer to the Swagger REST API documentation.

Note In an active/active Standard Cluster deployment, some events may cause a delay in the graceful shutdown process. Use the *Event Queue* on page 344 page to track events in the current TM work queue.

Graceful shutdown of a SecureTransport Server node

Before you proceed with graceful shutdown initiation, you must stop the Monitor Server.

Navigate to **Operations > Server Control**, click on the **Actions** menu in the top right corner, and select **Shut down node gracefully** to initiate a graceful shutdown of the entire SecureTransport Server node.

This process undergoes three consecutive steps:

1. Stops the Folder Monitor and Scheduler.
2. Gracefully stops all protocol services.
3. Gracefully stops the Transaction Manager.

Each step will be executed after successful completion of the previous one. The *Server Control* page displays messages with the shutdown status of all node components.

After step 3, the current SecureTransport Server will not be processing any transfers until you manually restart all services.

The option to gracefully shut down the SecureTransport Server node is also exposed as a REST API resource. For more information, refer to the Swagger REST API documentation.

Caution When you initiate graceful shutdown of a SecureTransport Server node through the Administration Tool, you must **not** close or refresh the *Server Control* page until all three steps are completed. If you need to perform actions on other screens in the Administration Tool, it is recommended to open it in a new tab. Graceful shutdown of the Administration Tool is not needed as the currently ongoing client- or server-initiated transfers must not be canceled. At that stage, the Administration Tool is not part of any transfer operation, and it must run in order to get the statuses of the protocol services and the Transaction Manager.

Graceful shutdown of SecureTransport Edge node

Before you proceed with graceful shutdown initiation, you must stop the Monitor Server.

Graceful shutdown on Edge nodes is similar to that on Server nodes. Note that there is no Folder Monitor, Scheduler or Transaction Manager component on Edge, however there is the Proxy server.

Navigate to **Operations > Server Control**, click on the **Actions** menu in the top right corner, and select **Shut down node gracefully** to initiate a graceful shutdown of the entire SecureTransport Edge node.

This process undergoes two consecutive steps:

1. Gracefully stops all protocol services.
2. Stops the Proxy server.

Monitor Server

The Monitor Server uses the `monitord` monitoring service to perform periodical checks and identify if the SecureTransport Servers are functional or not. If one or more monitored servers are not

responding, the monitoring service automatically restarts them. There are few exceptions to this: `monitord` does not restart a server if the DB server is not running; or if the server is manually stopped by an administrator.

Before version 5.5, SecureTransport Server was using cron for monitoring purposes. With SecureTransport version 5.5 and later, the Monitor Server relies on `monitord` as a separate process that uses the Quartz Job Scheduling Library.

The Monitor Server and `monitord` monitoring service can run with SecureTransport Server or SecureTransport Edge.

Start Monitor Server and monitord service

In order to make use of this functionality, you must have both the Monitor Server and `monitord` service started and running.

- To start the Monitor Server, go to **Operations > Server Control**, and on the **Monitor Server** tile click ⋮ (*more* icon) and then **Start service**.
- To start the `monitord` monitoring service, use the following script: `<FILEDRIVEHOME>/bin/start_monitord`.

Monitoring is applicable to all SecureTransport services. Whichever service you start, it attempts to start the `monitord` service (unless it is already running). To monitor a service that requires mutual authentication (Client authentication is mandatory), you need a client certificate . In this case, it is mandatory to have a local certificate named *monitord* having a private key with an appropriate Extended Key Usage configured.

Note The `monitord` service must use a non-SSL connection to check the PeSIT server, so you must check **Enable PeSIT over Plain Socket** on the *Server Control* page for monitoring. To prevent non-SSL PeSIT connections, restrict access to the non-SSL port you configure to the system where the PeSIT server and the Monitor server are running.

You can set up a monitoring schedule for each service by editing the `monitor.schedule.properties` file located in the `<FILEDRIVEHOME>/conf` folder. To exclude any of the servers from monitoring, you must remove its entry from the `monitor.schedule.properties` file.

You can configure number of times and intervals between consecutive attempts for `monitord` to restart each server by changing local configuration parameters.

The parameters apply to a respective server with its server name: `ADMIN`, `AS2D`, `DB` (embedded database only), `FTPD`, `HTTPD`, `SSHD`, and `TM` or `Proxy`, as follows:

- `Monitor.<server>.retryCount` – the number of times the Monitor Server tries to restart the server. The default value varies depending on the server.
- `Monitor.<server>.retryDelay` – the time in seconds that the Monitor Server waits between consecutive attempts to restart the server. The default value varies depending on the server.

These parameters are also editable in the `monitor.properties` file located in the `<FILEDRIVEHOME>/conf` folder. It is recommended to edit their values from the Administration Tool.

The monitoring service logs the status of each monitored service in a dedicated "out" file in the following location: `<FILEDRIVEHOME>/var/logs/`. The format of each file includes the service name in its filename, as follows: `monitor_<service>.out`

Besides manually starting `monitord`, the administrator can also stop and observe the server status by executing the respective script:

- `<FILEDRIVEHOME>/bin/stop_monitord` – stops the monitoring service

Note `monitord` is also stopped when executing the `<FILEDRIVEHOME>/bin/stop_all` script. The monitoring service is not stopped automatically when any of the other SecureTransport services are stopped, including the Admin service, as this would affect monitoring of the running services.

- `<FILEDRIVEHOME>/bin/status_monitord` – displays the status of the monitoring service

The log file of the `monitord` service is stored in the following location: `<FILEDRIVEHOME>/var/logs/monitord.log`.

When you make a change to any monitored service, you must stop and start the `monitord` service in order to reflect these changes.

During an upgrade, all SecureTransport cronjobs along with their schedules will be migrated to the `monitord` configuration and then deleted from cron. All non-SecureTransport related cronjobs will be preserved. This applies for all operating systems you install and run SecureTransport on.

Note With previous versions of SecureTransport on Windows server, the monitoring service was scheduled to run every 5 minutes. With version 5.5 onward, a fresh SecureTransport installation on Windows server is scheduled to run every 1 minute.

Health checks for services and cluster nodes

SecureTransport provides HTTP endpoints for monitoring the health of various system components, such as services or cluster nodes. They are primarily intended to relay operational status to a load balancer, enabling it to distribute traffic efficiently. Configuring a load balancer to utilize HTTP health endpoints allows continuous monitoring, ensuring only healthy servers receive traffic.

This section describes these endpoints and provides instructions on their configuration and usage.

SecureTransport supports two types of health checks: liveness and readiness.

- Liveness checks determine whether a service or server is operational (running).
- Readiness checks verify that a cluster node is ready to accept workloads.

Limitation: The health check only assesses the default listener!

The load balancer checks the liveness and readiness status by sending requests to particular HTTP endpoints. If it receives an HTTP response of 200, the server/service is deemed healthy. Otherwise, it is considered unhealthy and will stop processing traffic.

Health Check Configuration

The health check functionality can be configured on both backend servers and Edge nodes. You activate it by specifying a combination of three configuration options. These options are global, meaning they are shared between all backend servers in the cluster. However, these options are set separately for Edge nodes. Depending on whether Edge synchronization is enabled, they should be configured either collectively for all Edge nodes or individually on each Edge.

To activate and configure health check functionality, set the following options:

- `StatusChecker.enabled` must be set to `true` to allow reporting of the node/service status over HTTP.
- `StatusChecker.port` defines the HTTP port used when requesting a health check.
- `StatusChecker.heartbeatInterval` specifies how often, in seconds, the daemons update their status in the database. This frequency is important for obtaining accurate status results from the health check. The default value is 30 seconds.

For changes to take effect, restart the *monitord* service by navigating to the *<FILEDRIVEHOME>/bin* directory and executing the following commands:

```
./stop_monitord
./start_monitord
```

Liveness Check

Liveness status can be requested individually for each protocol daemon, the Admin or SOCKS service:

`GET {NodeIP}:{StatusChecker.port}/healthCheck?daemon=<daemon>` (e.g., ADMIN, SOCKS, HTTPD, FTPD)

Or, for the entire node:
`GET {NodeIP}:{StatusChecker.port}/liveness`

Node Liveness criteria:

For backends: Protocol daemons must be alive (in *Running*, *Stopping*, or *Stopped* status). The services intentionally stopped by the administrator are not considered unhealthy.

For Egde, there is no differentiation between the disabled and stopped state - both are disregarded. This means that if the Admin or the SOCKS service is stopped, it won't be considered for the liveness check. In case the Admin or the SOCKS is reported as running, but are not actually running upon a check, the node readiness status will be reported as unhealthy (503 Service Unavailable).

Status codes

HTTP Status Code	Status Name	Meaning
200	OK	Indicates that the service/node is healthy: all enabled services are running with at least one functional streaming connection, or stopped by admin.
503	Service Unavailable	Indicates that the service/node is NOT healthy: a service unexpectedly stopped (not stopped by an administrator) or no streaming connection has been established.
404	Not Found	Indicates that the server cannot find the requested resource.
N/A	Connection Refused	Returned when the health check service is not configured or enabled.

Readiness Check

Readiness can be requested individually for each cluster node via the request:

```
GET HTTP {NodeIP}:{StatusChecker.port}/readiness
```

Node Readiness Criteria

For a node to be deemed "Ready":

- It must be healthy (or "alive") - operational and responsive to requests.
- All configured protocol daemons must be running, with at least one streaming connection established to the Transaction Manager. Unlike the liveness check, the readiness check considers services intentionally stopped by the administrator as unhealthy.

  For Egde servers: there is no differentiation between the disabled and stopped states - both are disregarded. This means that if the Admin or the SOCKS service is stopped, it won't be considered for the readiness status. In case Admin or the SOCKS service is reported as Running, but is not actually running upon a check - node readiness status will be reported as unhealthy ("503 Service Unavailable").
  For backends: in case all protocol daemons are disabled, the condition for readiness is that the Transaction Manager and the Admin services are up and running.
- No shutdown process should be in progress.
- The `Zdu.Validate.Update` configuration option must be set to `true`. This option is set at a cluster level and propagated to all backend and Edge servers. It is used during zero downtime update (ZDU). The default value is `true` and must not be changed unless you are in the process of a ZDU.

Status Codes

This table provides an overview of each HTTP status code and its meaning:

HTTP Status Code	Status Name	Meaning
200	OK	Indicates that the cluster node is healthy; all readiness requirements are met.
503	Service Unavailable	Indicates that the node is NOT healthy. Possible reasons: a service is in a state different than RUNNING, or there's no active streaming connection , or `Zdu.Validate.Update` is set to `false`. It also occurs when some of the enabled listeners are stopped intentionally by the administrator.
404	Not Found	Indicates that the requested resource cannot be found.
N/A	Connection Refused	Returned when the health check service is not configured or enabled.

Troubleshooting

Health check service logs are stored in the log file of the *monitord* service at *<FILEDRIVEHOME>/var/logs/monitord.log*.

Use the operating system to monitor SecureTransport processes

As an alternative to the Monitor server, you can monitor the state of specific processes using operating system tools.

There are two categories of processes, those that implement file transfer and related functions and those that implement the administrative functions. You need to know their process IDs to monitor them.

The following topics describe the file transfer and admin processes:

- *File transfer processes* on page 299
- *Admin processes* on page 299

File transfer processes

For file transfer, there are six parent processes to monitor:

- java – AS2 proxy server
- java – FTP server
- java – HTTP server
- java – PeSIT server
- java – SSH server
- java – Transaction Manager server
- mysqld – Embedded database server, when used

The first five processes all interact with the Transaction Manager server.

The process IDs for each for these processes can be found in `<FILEDRIVEHOME>/var/run`. The files are called:

- `as2d.pid`
- `ftpd.pid`
- `httpd.pid`
- `pesitd.pid`
- `sshd.pid`
- `tm-java.pid`
- `db.pid`

Admin processes

For administration functions, there is one parent processes to monitor:

- java – Tomcat Admin server

It interacts with the file transfer processes by manipulating configuration and performing signaling.

The process ID for this process can be found in `<FILEDRIVEHOME>/var/run/admin/tomcat.pid`.

Server usage monitor

Operations > Server Usage Monitor

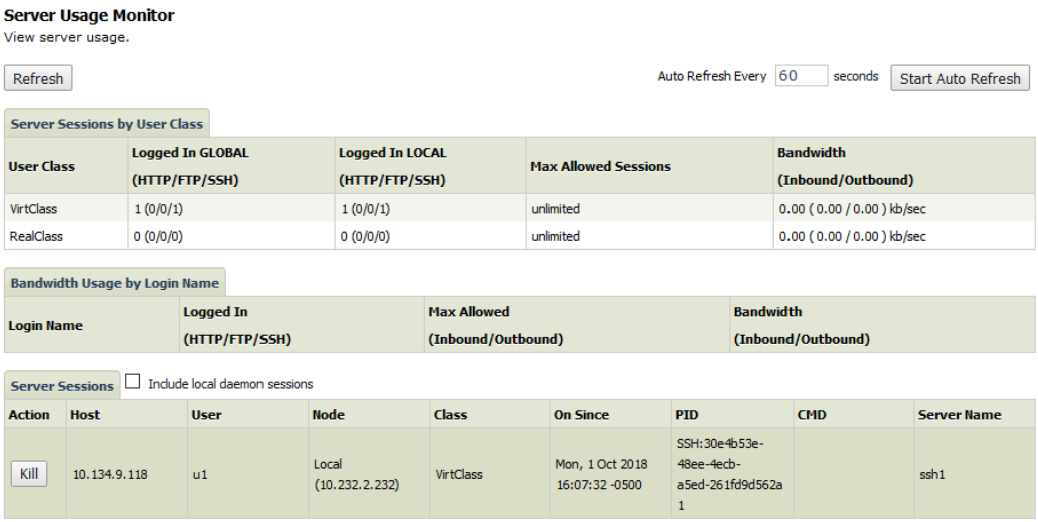
The *Server Usage Monitor* page presents info with current session and bandwidth usage, as follows:

- Server sessions by User Class - lists server sessions and bandwidth as consumed by User classes
- Bandwidth usage by login name - lists server bandwidth consumed by individual user accounts

- Server sessions - lists each connection session per user account

You also have the option to *Auto refresh Server Usage Monitor info* on page 302, as well as perform a manual refresh of the page.

Note that the *Server Usage Monitor* page shows FTP, HTTP(S) and SSH information only. Also, you can select what information to be monitored. For more information, see *Set usage monitor options* on page 200.



Note When in *cluster setup*, the information presented here includes all current sessions across all nodes.

Server sessions by User Class

This table allows you to monitor server sessions and bandwidth as consumed by User classes. The information is distributed into columns as follows:

- **User class**
- **Logged in GLOBAL (HTTP/FTP/SSH)** – the number of protocol (HTTP(S)/FTP/SSH) sessions that users in the respective user class are currently connected to globally.
- **Logged in LOCAL (HTTP/FTP/SSH)** – the number of protocol (HTTP(S)/FTP/SSH) sessions that users in the respective user class are currently connected to locally.
- **Max allowed sessions** – the upper limitation for open concurrent sessions per user class.
- **Bandwidth (Inbound/Outbound)** – the currently used bandwidth by the respective user class.

Note For more information on the format presented in the different columns of the *Server sessions by User Class* table, see the subtopic that follows: *Bandwidth usage by login name*.

Bandwidth usage by login name

This table allows you to monitor server bandwidth consumed by individual user accounts that are currently logged in. The information is distributed into columns as follows:

- **Login name** – the user account login name.
- **Logged In (HTTP/FTP/SSH)** – the number of protocol (HTTP(S)/FTP/SSH) sessions the user is currently connected to. The format used is [total sessions (HTTP/FTP/SSH)].
  For example, when the respective value for a selected user is (3 (1/1/1), this means that the user currently has 3 open sessions, 1 HTTP(S), 1 FTP and 1 SSH.
- **Max allowed (Inbound/Outbound)** – the maximum allowed traffic speed per the respective user. The format used is [Total allowed bandwidth (Max allowed Inbound bandwidth / Max allowed Outbound bandwidth].
  For example, when the respective value for a selected user is (768.0 (512.0 / 256.0), this means that the user's total traffic (both Inbound and Outbound) is limited to 768 kb/sec, of which 512.0 is the maximum allowed speed for inbound, and 256 is the maximum allowed speed for outbound traffic.
- **Bandwidth (Inbound/Outbound)** – the currently used bandwidth by the respective user. The format used is [Total allowed bandwidth (Max allowed Inbound bandwidth / Max allowed Outbound bandwidth].

Server sessions

This table lists each FTP, SSH, or HTTP(S) connection with the option to kill each one. The table presents the following options:

- **Action** – use the **Kill** button for each connection to terminate the respective session.
- **Host** – the IP address of the host location from which the user has connected. In streaming deployments with multiple NICs or multiple IP addresses on one NIC, for the desired IP address to be shown in the Server Usage Monitor and the SecureTransport logs in general, the IP address used for the streaming must be listed first in the hosts file. When there are multiple entries for the same host name, SecureTransport will always use the first address. Note that after editing the hosts file, you need to restart the daemons.
- **User** – the username of the respective user account.
- **Node** – the IP address to which the user is connected.
- **Class** – the user class the respective user belongs to.
- **On Since** – the Date and time the respective connectivity session was established.
- **PID** – the internal process ID of an FTP, SSH or HTTP(S) session within the server process.
- **CMD** – the transfer command.
- **Server Name** – the name of the server.

Select the **Include local daemon sessions** check-box to include the local sessions in the display results.

When using NICs, for the desired IP address to be shown in the logs, server usage monitor, etc, the system must be configured accordingly. For example, there must be an entry in the hosts file containing the IP from which connections from the edge to the backend are established(one specified in the network zone settings). This entry must be described first since a call to InetAddress.getLocalHost() returns the first result.

Auto refresh Server Usage Monitor info

Use the following procedure to enable automatic snapshot updates.

1. On the top right corner of *Server Usage Monitor* page, type the number of seconds between updates in the **Auto refresh every ____ seconds** box.
2. Click **Start Auto Refresh**. The button label changes to **Stop Auto Refresh** so you can later click it again to disable the automatic snapshot update.

Note Wile in Auto refresh mode, do not use **F5** on your keyboard or the **Refresh** button on the screen to refresh the browser window.

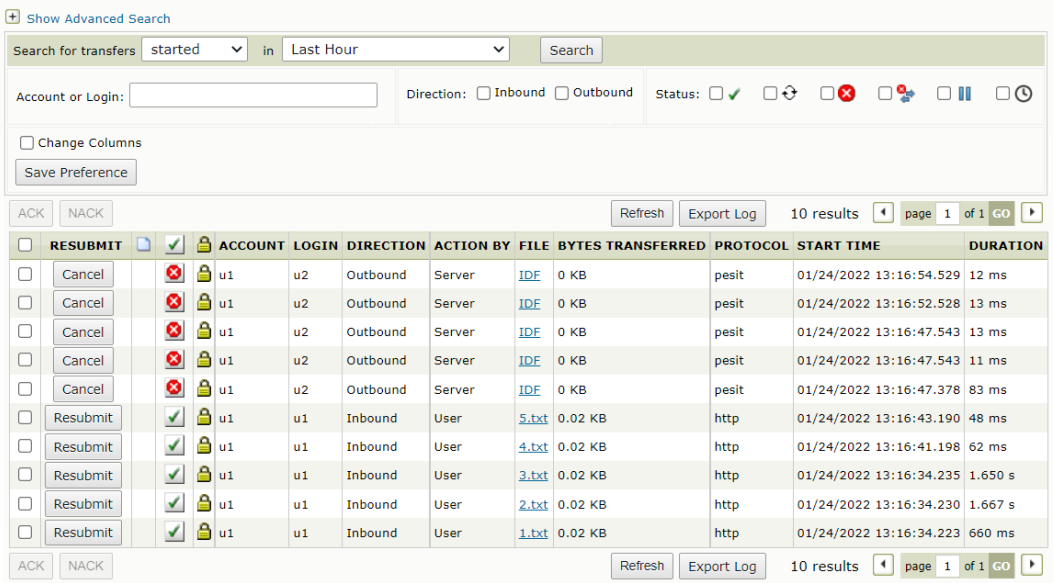
Disable automatic snapshot updates

To disable automatic snapshot updates, go to the *Server Usage Monitor* page and click **Stop Auto Refresh**. The button changes back its label to **Start Auto Refresh**.

File Tracking

SecureTransport records information about the file transfers that it processes in the so-called transfer log and displays it on the *File Tracking* page. This page is available on SecureTransport Server only and accessible by choosing **Operations > File Tracking**.

Every time you open the File Tracking page, SecureTransport automatically loads the transfer log entries for the past hour. This behavior is determined by the `FileTracking.InitialLoading.Enabled.Admin` server configuration option. It accepts boolean values, and the default value is `true`. To disable the initial auto-loading of log entries on the page, set the value to `false`.



Customize your view of the File Tracking table

You can view details about the file transfers in the *File Tracking* table. By default, it contains the following columns:

- Resubmit: When possible, SecureTransport shows a **Resubmit** button using which you can retry a failed transfer or resubmit a successful incoming one. See *Resubmitted and retried transfers on page 306*.
- MDN Receipt. See *View MDN receipt information about a transfer on page 309*.
- Transfer status: By clicking the transfer status icon, you can see more detailed status information, including the location of the file (Real File Location). You will be able to view more comprehensive information about the transfer if you are using Axway Sentinel. See *Transfer statuses on page 308*.
- Account and Login name

  Note For PeSIT transfers initiated by a partner, the *Login* column shows the name of the PeSIT transfer site that represents the partner.

- Direction of the transfer (inbound or outbound)
- Whether the transfer is initiated by the user or the server (*Action by* column)
- Details about the transferred file: name (*File*), *Bytes transferred*, and used *Protocol*

  Note On Windows environments, when SecureTransport receives a file with a name that contains Windows reserved characters, they are represented with their corresponding Unicode hexadecimal values.

  Note For all PeSIT transfers, the *File* column shows the name of the transfer profile used, and the name of the file is displayed in the *Local filename* column, which

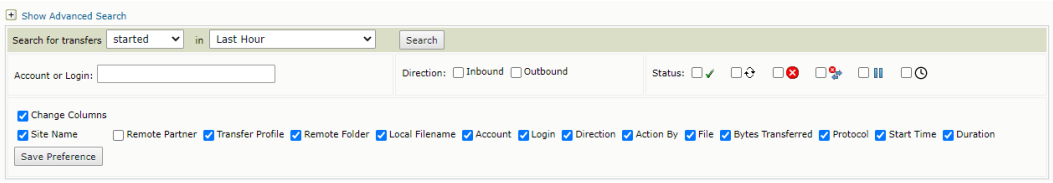
is not shown in the table by default.

- Start time and duration of the transfer

The File Tracking table can additionally show the following transfer details:

- Transfer site name. See *Transfer sites on page 540*.
- Remote partner: the remote account name (for pluggable transfer sites only)
- Transfer profile (PeSIT only). See *Transfer profiles on page 640*.
- Remote folder: For client-initiated transfers, this is the local account directory where the file is uploaded. For server-initiated transfers, this is the remote directory.
- Local name of the transferred file

To modify which columns are displayed in the File Tracking table, click **Change Columns**. Select or clear the checkboxes for the columns you want to show or hide. Then, click the **Save Preference** button to save your customized view. Preferences are saved for the currently logged in administrator. The changes take immediate effect.



Consolidated view of all information about a transfer

Instead of creating an individual log entry for each failed resubmit and retry of the same server-initiated transfer, SecureTransport can be configured to log all the records with the same CycleID in a single entry. This entry gets updated for each retry - the Transfer Log shows information for the most recent attempt, and the details for the retry and resubmit attempts are visible in the *Status Details* box only. Therefore, when the consolidated view is enabled, certain search filters, like *Remote Partner*, *Resubmit Status* and *Security*, return results based on the last transfer status only.

To enable this functionality, set the configuration option `FileTracking.ShowLatestStatusOnly.Enabled` to `true`.

With Consolidated view disabled:

RESUBMIT			ACCOUNT	LOGIN	DIRECTION	ACTION BY	FILE	BYTES TRANSFERRED	PROTOCOL	START TIME	DURATION
Resubmit			u2	U1	Inbound	User	PROF	0 KB	pesit	09/20/2022 13:19:39.599	212 ms
Resubmit			u1	u2	Outbound	Server	PROF	0 KB	pesit	09/20/2022 13:19:39.358	424 ms
Resubmit			u1	u2	Outbound	Server	prof	0 KB	pesit	09/20/2022 13:15:25.094	20 ms
			u1	u2	Outbound	Server	prof	0 KB	pesit	09/20/2022 13:15:08.044	34 ms
			u1	u2	Outbound	Server	prof	0 KB	pesit	09/20/2022 13:14:54.478	42 ms

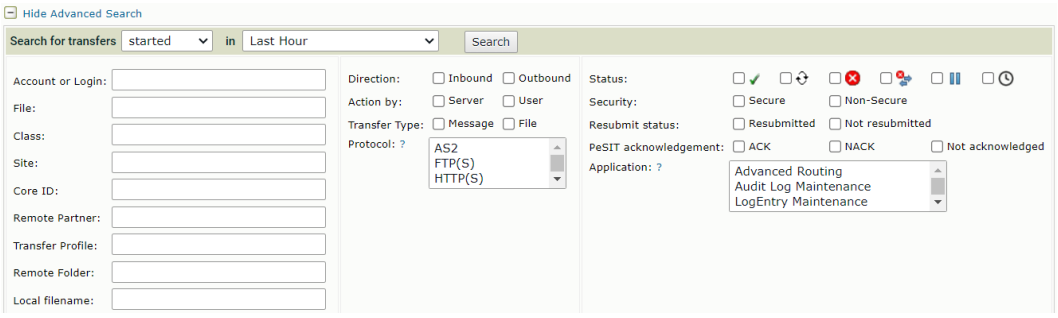
With Consolidated view enabled:

RESUBMIT			ACCOUNT	LOGIN	DIRECTION	ACTION BY	FILE	BYTES TRANSFERRED	PROTOCOL	START TIME	DURATION
Resubmit			u2	U1	Inbound	User	PROF	0 KB	pesit	09/20/2022 13:26:08.125	211 ms
Resubmit			u1	u2	Outbound	Server	PROF	0 KB	pesit	09/20/2022 13:26:08.034	279 ms

Search file tracking information

Use the following procedure to search for file tracking information.

1. Select **Operations > File Tracking**.
2. On the *File Tracking* page, specify the search criteria.



   a. In the *Search for transfers* pane, specify whether you want to search based on the time the transfer started or was completed and the time frame in which to search.
   b. (Optional) Specify the account or login associated with the transfer you are searching for.
   c. (Optional) Specify whether to search for inbound or outbound transfers.
   d. (Optional) Specify whether you want to include successful transfers, transfers currently in progress, paused transfers, failed transfers, or failed subtransmissions. You can specify more than one.
3. (Optional) Show the Advanced Search fields and specify additional search criteria.
   a. Specify the file name, class, site, Core ID, remote partner, transfer profile, remote folder, or local filename associated with the transfer you are searching for.
   b. Specify transfers initiated by the server or the user.
   c. Specify the transfer type: message or file.
   d. Specify the protocols used for the transfers. You can use **Control+click** and **Shift+click** to select more than one.
   e. Specify whether the transfer was secure or non-secure.
   f. Specify whether the transfer was resubmitted or not resubmitted. For more information, see *Resubmitted transfers on page 306*.
   g. Specify whether the PeSIT transfer was positively acknowledged (ACK), negatively acknowledged (NACK), or not acknowledged at all. If all three options are selected, the search displays all PeSIT transfers.
   h. Specify the application associated with the transfer. You can use **Control+click** and **Shift+click** to select more than one.
4. When ready, click **Search**.

File Tracking specifics

- In a Standard Cluster, the *File Tracking* page on the primary server in the cluster provides a consolidated view of the transfer data. See *Consolidated log data representation* on page 356.
- Due to protocol specifics, when a user pauses or resumes a client-initiated transfer, two entries are recorded on the *File Tracking* page for the interrupted transfer. The entries have the following characteristics:

  HTTP CIT: The log entry for the portion of the file uploaded before the transfer was paused is marked as failed and a **Resubmit** button is not displayed. The second entry - for the portion of the file uploaded after the transfer is resumed - is marked as successful and a **Resubmit** button is displayed.

  FTP/SSH CIT: The log entry for the portion of the file uploaded before the transfer was paused is marked as successful and a **Resubmit** button is displayed. The second log entry - for the portion of the file uploaded after the transfer is resumed - is marked as successful and a **Resubmit** button is displayed.
- By default, SecureTransport does not log SIT transfers if there are no files on the remote server to pull. To change that, set the `ZeroByteWildcardPullAllowed` configuration option to `true`, thereby making SecureTransport indicate such transfers by creating a file tracking entry for a zero-byte transfer. In this entry, the evaluated download pattern configured in the transfer site would be shown as a file name; however, there is no physical file to trigger an Advanced Routing subscription. `ZeroByteWildcardPullAllowed` is not applicable for Folder Monitor and pluggable transfer sites.
- The CycleId of the original transfer is preserved when retrying or resubmitting failed server-initiated transfers. For more information, see *CycleId* on page 162.

The following topics provide information on viewing and managing file transfer activity:

- *View file transfer information on page 309*
- *Manage file transfers from the File Tracking page on page 314*
- *Transfer Log Maintenance application on page 322*

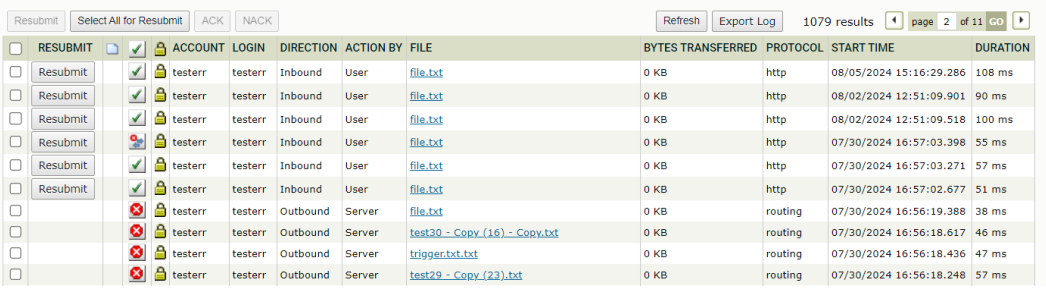
Resubmitted and retried transfers

This section explains the difference between retried and resubmitted transfers in the context of SecureTransport.

Resubmitted transfers

A *resubmitted* transfer is a transfer that has been submitted for subsequent execution manually from the *File Tracking* page by using the **Resubmit** button. An administrator can resubmit successful transfers and permanently failed transfers with the following exceptions: AdHoc mail, AdHoc attachment, CIT download, failed CIT upload, in-progress, retrying, PeSIT in a Paused state,

PeSIT message, Publish To Account Advanced Routing step. On the *File Tracking* page, SecureTransport indicates that a transfer can be resubmitted by displaying a **Resubmit** button next to it. No button is displayed when the transfer is of the above-mentioned exceptions.



Note The CycleId of the original transfer is preserved when resubmitting failed SITs, and a new CycleId is generated when resubmitting successful outbound transfers. For more information, see *CycleId* on page 162.

You can select and resubmit all eligible transfers on the current *File Tracking* page in bulk using the **Select All for Resubmit** and the **Resubmit** buttons. Both buttons are available at the top and bottom of the table for your convenience. Note that if there are related failed transfers (with the same cycleId and coreId), only the most recent one will be included in the selection and resubmitted.

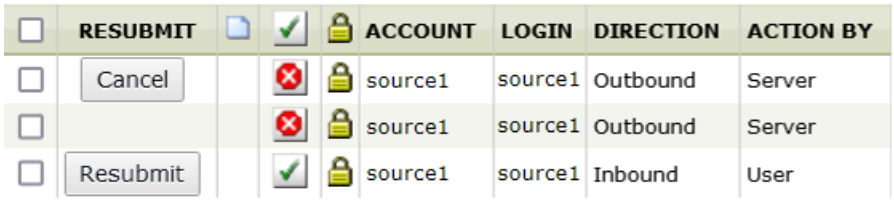
Resubmission specifics

When resubmitting a transfer, SecureTransport uses either an archived copy created when the server first received the file, or the original file if it is available. It always checks if an archived copy of the file is available before checking for the original file.

- If there is no archive for the transfer and the original file is not available, or if the Archive Folder path has been changed, the resubmit attempt fails. Archived files are removed from the server periodically. You can control these settings using the *Archive Maintenance application on page 826* and *File archiving global configuration on page 226*.
- If there is another file with the same name in the same location, the transfer is not resubmitted if the modification date and time do not match.
- If there is no active subscription for the transfer at the moment of resubmission, the subscription is reactivated. If a post-transmission action fails, it is restarted when you resubmit the transfer.
- Resubmitting a failed outbound SIT initiated by the Send to Partner step only resubmits the transfer itself and does not trigger the Advanced Routing flow. To re-initiate the Advanced Routing, the administrator must resubmit the inbound CIT. For more information, see *Send To Partner on page 948*.
- Resubmitting an inbound CIT or SIT does not initiate a new file transfer but triggers the file processing.

Retried transfers

A *retried* transfer is a temporarily failed SIT that is submitted for subsequent execution automatically by SecureTransport based on a predefined retry policy. For more information, see *Retry server-initiated transfers on page 764*. While a retry attempt is in progress, an administrator can cancel the transfer execution from the *File Tracking* page by clicking the **Cancel** button shown next to the transfer. Note that the **Cancel** button cannot be used for transfers initiated by the Advanced Routing application.



Transfer statuses

The transfer status is represented by an icon. The following table shows each icon and its meaning:

Icon	Status	Protocols supported
	MDN Receipt – Click to view or verify the MDN receipt for the transfer or to view the PeSIT acknowledgment for the transfer.	All
	Processed/Successful – The file was transferred successfully.	All
	In progress – The file is currently being transferred.	All
	Paused – The remote server has paused the PeSIT transfer.	PeSIT
	Failed – The file transfer failed for some reason. Transfers intentionally aborted by file transfer clients and end users are included in this category. The *File Tracking* page provides additional information to help distinguish transfers intentionally aborted from those that failed for other reasons.	All
	Failed Subtransmission – The file transfer was successful, but one or more Subtransmission actions requested failed for some reason. Subtransmission actions include post-transmission actions and data transformations such as encryption or decryption.	All

Click each transfer status icon to view detailed status information about the file transfer.

The padlock icon is displayed for transfers performed over a secure connection, such as FTPS, HTTPS, or SSH.

Note If a file is renamed immediately after a file transfer, the MDN receipt creation and verification may fail.

View file transfer information

Use the *File Tracking* page to search for file tracking information and view information about the file transfers.

- *View and export log statistics about transferred files* on page 309
- *View MDN receipt information about a transfer* on page 309
- *View detailed information about a file transfer* on page 310
- *View transfer history of a file* on page 312
- *View detailed information about an AR execution* on page 313
- *View detailed information about a PeSIT message transfer* on page 313

View and export log statistics about transferred files

Use the following procedure to view and export file transfer log statistics.

1. Select **Operations > File Tracking**.

   The *File Tracking* page is displayed.
2. Click **Export Log** to export the displayed file tracking data to a `.csv` file. You cannot export all the records from the database at one time.
3. Choose whether to save or open the file.
4. Click **OK**.

For each transfer, the export contains the following data: Status, Account, Login, UserClass, UserType, Application, Transfer Site, Direction, Action By, File, Remote Partner, Transfer Profile, Transfer Content Type, Remote Folder, Local Filename, Local Folder, ICAP Details, Local File, Size, Protocol, Secure, Mode, Start Time, End Time, Duration, Remote Host, Transfer ID, Session ID, Session Start Time, Operation Index, PeSIT Message, CoreID, Resubmitted, Additional info, X-Forwarded-For, Security Parameters, and Server Name.

View MDN receipt information about a transfer

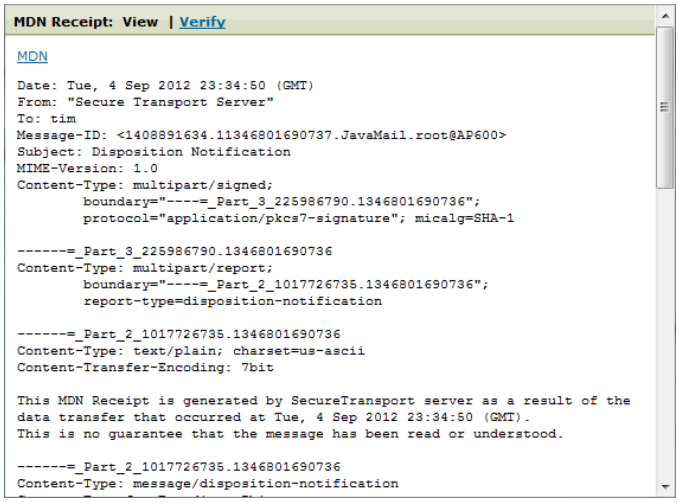
Use the following procedure to view MDN receipt information.

1. Select **Operations > File Tracking**.

   The *File Tracking* page is displayed.

2. Click the MDN receipt icon for a specific transfer. An MDN receipt icon is displayed only when a receipt for the transfer is available in the system and the transfer was successful. To enable MDN receipts, see *Certificates to generate during initial setup on page 63*.

   The *MDN Receipt* dialog box is displayed. You can either view the MDN receipt or click **Verify** to view the MDN signature and file integrity check results. Clicking **Close** returns you to the *File Tracking* page.



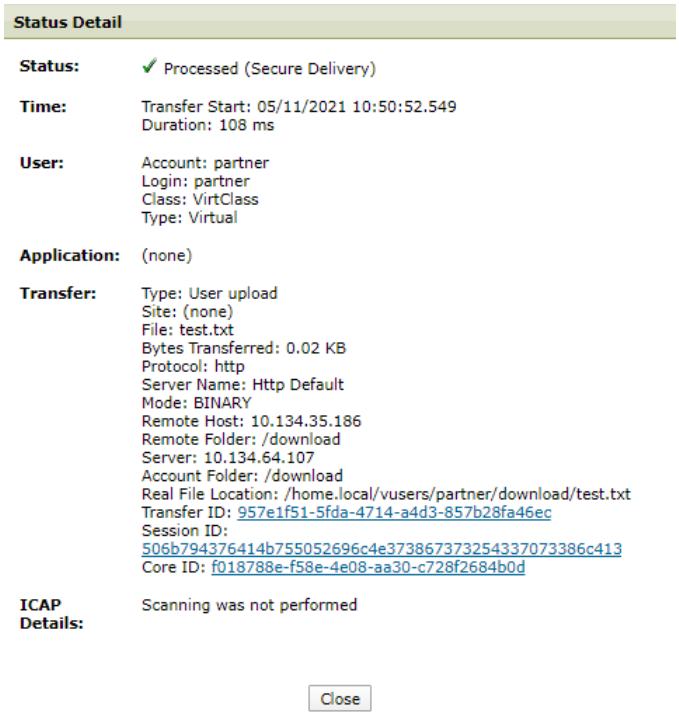
MDN receipts are not supported for PeSIT messages.

View detailed information about a file transfer

Use the following procedure to view detailed status information.

1. Select **Operations > File Tracking**.
2. Click the status icon or the file name for a specific transfer.

The *Status Detail* dialog box is displayed.



The *Status Detail* dialog box shows the following information:

Name	Description
Status	Displays the transfer status: Processed, In Progress, Failed, Aborted, or Failed Subtransmission.
Time	Displays the date and time the transfer was started and its duration in milliseconds.
User	Displays information about the account that performed the transfer: account name, login name, class type and user type.
Application	Displays the application instance name.

Name	Description
Transfer	Displays the transfer type, transfer site name, file name and size, transfer protocol, server name, transfer mode, remote host name, remote folder name, account folder name, real file location, success or failure details, and protocol messages. At the bottom, there are three ID links: • Transfer ID - a unique identifier of a transfer in SecureTransport. It is used to track what happened to a file during a single transfer (e.g upload, download, push, pull). When you click on the TransferID value, SecureTransport opens the *Server Log* page with the value pre-filled as a search criterion. • Session ID - a session identifier in SecureTransport. It is used to track the file(s) during one or multiple transfers that happened in the same session. When you click on the SessionID value, SecureTransport opens the *Server Log* page with the value pre-filled as a search criterion. • Core ID - a unique file identifier in SecureTransport. It is generated the first time the file arrives in SecureTransport, and is stable throughout the lifespan of the file, even if its name changes. Core ID is used to track the file across different transfers and sessions. When you click on the Core ID value, SecureTransport refreshes the *File Tracking* page with the Core ID value pre-filled as a search criterion.
Post Transmission Status	Displays the operation type, whether the operation succeeded or failed, the result of the operation, and comments that can provide additional information, such as if a transformation was performed. Multiple operations might be displayed.

Status Detail specifics:

- For server-initiated transfers, the *Status Detail* box also shows the protocol messages.
- For HTTP transfers, the headers are provided since there are no commands sent.
- For AS2 transfers, messages are also generated from the HTTP headers.
- For PeSIT transfers, the *Status Detail* box also shows the transfer profile and the transfer content type.

View transfer history of a file

Use the following procedure to view the transfer history of a file.

1. Select **Operations > File Tracking**.

   The *File Tracking* page is displayed.
2. Click the name of a file.

The *File History Details* dialog box is displayed. It lists out the file transfer status details starting from the last client-initiated upload or server-initiated download and ending with the first renaming or deletion of the file with the same name. If the client-initiated upload or server-initiated download is missing because the file tracking log was rotated, the dialog box shows the first event for that file name.

File History Details contains detailed information about tracked events. Those include uploads and downloads, server-initiated transfer protocol messages, PGP encryption, routing a file from one account to another, post-transmission actions, and file deletion or renaming from a client-initiated command.

Note When transfers are performed by users behind a proxy or a load balancer, an additional parameter is listed: `X-Forwarded-For`. This is a dedicated HTTP header which is commonly used to identify the originating IP address of the user account and is especially useful when the user is behind a proxy or a load balancer. In such case, the `Remote Host` displays the IP address of the proxy/load balancer, and the `X-Forwarded-For` parameter displays the user's original IP address. Note that when the user is not behind a proxy/load balancer, the original IP address is displayed with the `Remote Host` parameter, and `X-Forwarded-For` is hidden from view.

3. Click **Close** to return to the *File Tracking* page.

Note When a logged in account does not have permission to delete files and tries to delete them, the *File History Details* dialog box is not available.

View detailed information about an AR execution

Use the following procedure to view detailed status information about an advanced route execution:

1. Select **Operations > File Tracking**.

   The *File Tracking* page is displayed.

2. Click the status icon or the file name for a specific advanced route execution.

   The *Status Detail* window is displayed. It shows detailed information about the file transfer and the Route Status, including the operation type, whether the route succeeded or failed, the route package name, start and end times, duration, and the execution ID.

3. Click **Close** to return to the *File Tracking* page.

View detailed information about a PeSIT message transfer

Use the following procedure to view detailed status information.

1. Select **Operations > File Tracking**.
2. Click the status icon next to a PeSIT message transfer.

   The *Status Detail* dialog box shows the following information:

Name	Description
Status	Displays the transfer status: Processed, In Progress, Failed, Aborted, or Failed Subtransmission.
Time	Displays the date and time the transfer was started and its duration in milliseconds.
User	Displays information about the account that performed the transfer: account name, login name, class type and user type.
Application	Displays the application instance name.
Transfer	Displays the transfer type, transfer site name, transfer profile, transfer content type, bytes transferred, transfer protocol, server name, transfer mode, remote host name, remote folder name, account folder name, PeSIT message content. At the bottom, there are three ID links: • Transfer ID - a unique identifier of a transfer in SecureTransport. It is used to track what happened during a single transfer. When you click on the TransferID value, SecureTransport opens the *Server Log* page with the value pre-filled as a search criterion. • Session ID - a session identifier in SecureTransport. It is used to track one or multiple transfers that happened in the same session. When you click on the SessionID value, SecureTransport opens the *Server Log* page with the value pre-filled as a search criterion. • Core ID - a unique identifier in SecureTransport. It is generated the first time the message arrives in SecureTransport, and is stable throughout the lifespan of the message. When you click on the Core ID value, SecureTransport refreshes the *File Tracking* page with the Core ID value pre-filled as a search criterion.
Post Transmission Status	Displays the operation type, whether the operation succeeded or failed, the result of the operation, and comments that can provide additional information. Multiple operations might be displayed.

Manage file transfers from the File Tracking page

The following topics provide how-to instructions for managing file transfers:

- *Resubmit a transfer* on page 315
- *Cancel a transfer* on page 315
- *Acknowledge a PeSIT transfer* on page 315

Resubmit a transfer

If a file transfer fails permanently, SecureTransport displays a **Resubmit** button in the *RESUBMIT* column.

1. Navigate to **Operations > File Tracking**.
2. Click **Resubmit** to the left of the failed transfer you want to resubmit.

   A new line is added to the *File Tracking* page showing the progress of the resubmitted transfer.

Resubmit transfers in bulk

You can select and resubmit failed transfers in bulk, rather than one by one. To do so:

1. On the *File Tracking* page, click the **Select All for Resubmit** button.
   This will select all eligible transfers on the current *File Tracking* page. To exclude an entry from the selection, uncheck the box next to it.

   Note If there are related failed transfers (with the same cycleId and coreId), only the most recent one will be included in the selection.

2. To resubmit the selected transfers, click the **Resubmit** button next to **Select All for Resubmit**.

Cancel a transfer

If a server-initiated transfer fails temporarily, SecureTransport can automatically retry the transfer a set number of times at set intervals. For more information, see *Retry server-initiated transfers on page 764*.

When a temporarily failed server-initiated transfer is scheduled for a retry, SecureTransport displays a **Cancel** button in the *RESUBMIT* column. To stop the retry attempt, click **Cancel**.

Note **Cancel** cannot be used to control automatic retries when using the Advanced Routing application.

Acknowledge a PeSIT transfer

Among other transfer information, the *File Tracking* page shows if a PeSIT transfer has been acknowledged or not. Via acknowledgments, SecureTransport signals whether application-level errors were detected. Acknowledgments are not supported for PeSIT messages.

SecureTransport supports two modes:

- Automatic acknowledgment on success
- Manual acknowledgment

Depending on the acknowledgment method used, SecureTransport indicates a problem either by not sending an acknowledgment, or by sending a negative one. A PeSIT transfer can have only one acknowledgment - positive or negative.

Automatic acknowledgment is configured per transfer profile; the message text is specified in the `Pesit.Transfer.Acknowledge` configuration option. In this mode, SecureTransport automatically acknowledges the transfer only if its processing completes successfully.

For inbound PeSIT transfers that have been completed but not acknowledged, you can send a positive (ACK) or a negative (NACK) acknowledgment manually using the dedicated buttons on the *File Tracking* page.

Resubmitted PeSIT transfers can be acknowledged manually or automatically. When a transfer is acknowledged, all related transfers with the same *Transfer ID* (PI13), if any, receive the same acknowledgment status. Transfers with unique PI13 values are acknowledged independently. For more information, see *PI13* on page 163*.

Example scenario:

A file is transferred between two accounts on the same server, user *u1* (sender) and user *u2* (receiver).
The server-initiated outbound transfer fails temporarily. It is retried the configured number of times, until a permanent failure occurs. The failed transfer is resubmitted, and two new log entries are recorded: one for the successful inbound transfer and one for the successful outbound transfer. All entries have the same PI13 value.

- When the inbound transfer is acknowledged, all transfers with the same PI13 value get acknowledged as well.

RESUBMIT	ACCOUNT	LOGIN	DIRECTION	ACTION BY	FILE	BYTES TRANSFERRED	PROTOCOL	START TIME	DURATION
Resubmit	u2	U1	Inbound	User	PROF	0 KB	pesit	09/20/2022 13:19:39.599	212 ms
Resubmit	u1	u2	Outbound	Server	PROF	0 KB	pesit	09/20/2022 13:19:39.358	424 ms
Resubmit	u1	u2	Outbound	Server	prof	0 KB	pesit	09/20/2022 13:15:25.094	20 ms
	u1	u2	Outbound	Server	prof	0 KB	pesit	09/20/2022 13:15:08.044	34 ms
	u1	u2	Outbound	Server	prof	0 KB	pesit	09/20/2022 13:14:54.478	42 ms

- If the successful outbound transfer is resubmitted, it is assigned a new PI13 value and should be acknowledged independently. In this example, we have sent a negative acknowledgment.

RESUBMIT	ACCOUNT	LOGIN	DIRECTION	ACTION BY	FILE	BYTES TRANSFERRED	PROTOCOL	START TIME	DURATION
Resubmit	u2	U1	Inbound	User	PROF	0 KB	pesit	09/20/2022 13:26:08.125	211 ms
Resubmit	u1	u2	Outbound	Server	PROF	0 KB	pesit	09/20/2022 13:26:08.034	279 ms
Resubmit	u2	U1	Inbound	User	PROF	0 KB	pesit	09/20/2022 13:19:39.599	212 ms
Resubmit	u1	u2	Outbound	Server	PROF	0 KB	pesit	09/20/2022 13:19:39.358	424 ms
Resubmit	u1	u2	Outbound	Server	prof	0 KB	pesit	09/20/2022 13:15:25.094	20 ms
	u1	u2	Outbound	Server	prof	0 KB	pesit	09/20/2022 13:15:08.044	34 ms
	u1	u2	Outbound	Server	prof	0 KB	pesit	09/20/2022 13:14:54.478	42 ms

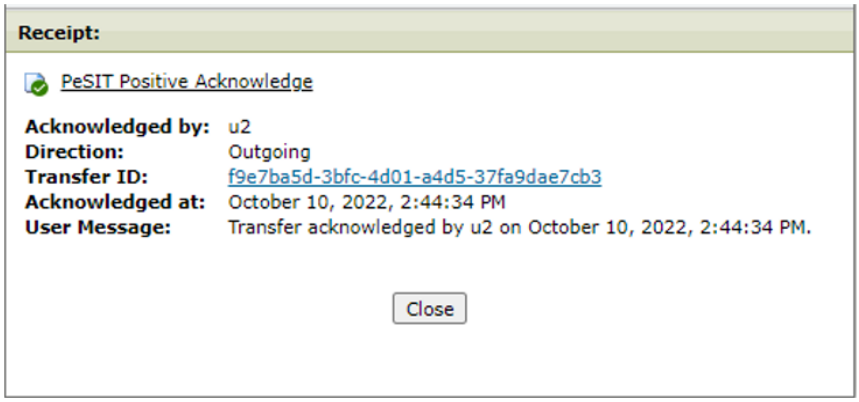
Note that SecureTransport can be configured to log all records with the same PI13 value in a single entry instead of creating individual ones. See also *Consolidated view of all information about a transfer on page 304*.

The following table summarizes when acknowledgments can be sent or received for client- and server-initiated transfers.

Mode		Client download	Client upload	Server pull	Server push
Sending acknowledgment (positive or negative)	Automatic	×	✓	✓	×
	Manual	×	✓	✓	×
Receiving acknowledgment (positive or negative)	Automatic	✓	×	×	✓
	Manual	✓	×	×	✓

Acknowledgment receipt and user messages

SecureTransport indicates that a PeSIT transfer has been acknowledged by displaying an icon next to it: for a positive acknowledgment, and for a negative acknowledgment. Clicking on the icon will open the acknowledgment receipt.



The acknowledgment receipt contains the following information:

- **Acknowledged by**: the user account that has acknowledged the transfer
- **Direction** of the transfer:
  - **outgoing** - if SecureTransport has sent the acknowledgment
  - **incoming** - if SecureTransport has received the acknowledgment
- **TransferID**: a unique identifier of a transfer in SecureTransport
- **Acknowledged at**: the date and time when the acknowledgment was sent or received
- **User Message**: the PeSIT PI91 value sent or received when a transfer is acknowledged. The default user message is specified via the `Pesit.Transfer.Acknowledge` configuration option.

  For manual acknowledgments, it is possible to customize the user message via the REST API resource `POST /logs/transfers/{id}/operations` using the `ackMessage` parameter. The maximum size of a custom message is 4096 bytes. If no

custom message is specified, SecureTransport uses the default user message.

The user message can be defined using the PeSIT variables listed in the *PeSIT expressions* on page 1110 section.

CFT PeSIT extensions

A typical use of the PeSIT protocol is for file transfers with the Axway Transfer CFT product. The latter makes use of a custom variant of the PeSIT E protocol that allows supplementary information, like parameters and metadata, to be conveyed from one Transfer CFT to another in the PI 99 of the PeSIT service primitives. This extra information is often referred to as "CFT PeSIT extensions" in Axway documentation, and the communication mode is known as PeSIT CFT.

As of Update 5.5-20230330, SecureTransport complies with PeSIT CFT extensions and can handle this PI 99 usage. This proves particularly useful when SecureTransport serves as an intermediate partner (relay) in a PeSIT flow between two Transfer CFTs as it maintains the integrity of the extended PI 99 message from the sender to the receiver.

Configuration

SecureTransport Update 5.5-20230330 and above include a server configuration option, `Pesit.CftExtensions.Enabled`, which enables the usage of PeSIT extensions. With this option set to `true`, which is the default value, SecureTransport mimics the behavior of a Transfer CFT, structuring the User Message (PI 99) field in the data units (FPDUs) CONNECT, ACK CONNECT and CREATE of the PeSIT protocol in the CFT format. However, in order to keep the CFT format, there are also two requirements: the User Message fields in both the PeSIT transfer site in use and the AR Send To Partner step (if present) must be left empty. Otherwise, the User Message defined in the Send To Partner overrides the transfer site setting, which in turn has priority over the configuration option.

Using PeSIT with Axway Transfer CFT

In a PeSIT flow between SecureTransport and Transfer CFT, the PeSIT CFT communication mode is negotiated in the following way: the initiator sends a protocol connection request (FPDU CONNECT) containing a PI 99 in a specific CFT format. If the responder replies with an ACK CONNECT with the same structured PI 99, the two partners start communicating in PeSIT CFT mode.

The PI 99 adheres to the following format:

```
User Message (PI_99) = "CFT Y=<partner_
os>,D=<timestamp>,V=<product_version>,Z=<build_
version>,K=<license_key>,C=<seal>"
```

It starts with "CFT " (4 characters) and ends with a seal – a checksum to verify that the PI 99 is in the CFT-specific format. The rest of the content is identification parameters in the form of key-value pairs.

For example, the following log excerpt shows SecureTransport 5.5 and Transfer CFT 3.6 negotiating the PeSIT CFT communication mode.

```
>>> FPDU_CONNECT (…User Message (PI_99) = "CFT
D=20230314124755729,V=ST55,Y=S,C=224460830")
<<< FPDU_ACONNECT (…User Message (PI_99) = "CFT
Y=S,D=2023031412215736,V=3.6,Z=--,K= ,C=438763577,1=NACK")
```

Note the V parameter. It is mandatory; for SecureTransport, its value is always `ST55`.

The PeSIT PI 99 field sent during the file creation process (FPDU CONNECT) includes much more detailed information about the file transfer, including

- The partners on behalf of which the transfer is made (the initial sender and final recipient of the file)
- Parameters, which have to be conveyed during the successive connections required to transfer the file through the various nodes of the network.

In PeSIT CFT mode, Transfer CFT modifies the message specified in the PI_99 parameter by adding additional key-value pairs. For example, if you specify PI 99=Test, CFT sends `User Message (PI_99) = ”CFT key1=value1 key2=value2 … A=Test”`.

Here is an example of what Transfer CFT could send, showing all possible parameters:

```
CFT A=<PI_99>,D=<timestamp>,E=<Original_File_Name>,G=<Receiver_
Application_Name>,H=<Sender_Application_Name>,K=<License_
Key>,L=<Sender_User>,M=<Receiver_User>,Q=<File_
Version>,U=<User_Identification>,V=<Receiver_Product_
Version>,W=<Group_Id>,Y=<Operating_System>,Z=<Receive_Product_
Build_Version>,1=<NACK - if NACK was sent>
```

In SecureTransport, you can use the following expressions to access the contents of a PI99 sent by Transfer CFT:

- `${pesit.cftServiceParam}`, which holds the entire modified string from CFT; for example, `CFT key1=value1 key2=value2 … A=Test`
- `${pesit.serviceParam}`, which holds the extracted value from the A key; for example, `Test`.

This allows SecureTransport to use both the full extended parameter string and the specific value intended to be sent.

PeSIT-extension PI codes

This section describes the PI codes for PeSIT extensions. A PESIT extension is additional information added to the PI that is specific to Transfer CFT. These extensions comprise the following:

- PI codes that receive an extension relative to the standardized usage
- PI codes that have been specially created and hence convey additional information

Any PI not mentioned in this section is used according to the standardized version of PeSIT.

A file creation request may look like this:

```
>>> FPDU_CREATE (... File Type (PI_11) = 0, File Name (PI_12) = "BIN",
Transfer ID (PI_13) = 7412564, ....
Requestor ID (PI_3) = "BB", Sender ID (PI_3) = "2222 4444", Server ID (PI_4) =
"CC", Receiver ID (PI_4) = "1111 3333", ....,
Originator (PI_61) = "", Destination (PI_62) = "",
User Message (PI_99) = "CFT B=0,R=4096,O=C,S=1,I=C1412564,T=B,F=U,X=
,Y=S,G=1111,H=2222,L=4444,M=3333,P=128,U=guest,W=,0=pub/FTEST,J=0;0;0;0;3;0;5;
4;0;0")
```

Non-standard PI code usage

PI 3 and PI 4

These codes specify the names of the initial and final users and applications. They are defined by Transfer CFT using the SAPPL, RAPPL, SUSER, and RUSER parameters. For PeSIT E CFT/CFT, the maximum allowed length for RAPPL and SAPPL is 48 characters; for RUSER and SUSER, it is 28 characters. Those parameters are also sent through PI 99.
Here is how they are recorded in SecureTransport:

- In case no RAPPL/RUSER and/or SAPPL/SUSER are sent from CFT, PI 3 and PI 4 only represent the partners' names on CFT side.

  For example, *Requestor ID (PI_3) = "BB", Server ID (PI_4) = "CC"*
- If RAPPL/RUSER and SAPPL/SUSER are specified in the transfer command, PI 3 and PI 4 represent both the names and the values of those fields. For example:

  Requestor ID (PI_3) = "BB", Sender ID (PI_3) = "SAPPL suser", Server ID (PI_4) = "CC", Receiver ID (PI_4) = "RAPPL ruser"

PI 61 and PI 62

These codes specify the initial sender and final recipient of the file to be transferred. Here are some examples of how they are recorded in SecureTransport based on the flow and whether RAPPL and SAPPL are specified:

- In the flow CFT -> ST -> CFT with RAPPL/SAPPL not specified: *Originator (PI_61) = "", Destination (PI_62) = "CC"*
- In the flow CFT -> ST -> CFT with RAPPL/SAPPL: *Originator (PI_61) = "", Destination (PI_62) = "CC"*
- In a direct transfer ST -> CFT: *Originator (PI_61) = "", Destination (PI_62) = ""*

PI 99

When SecureTransport acts as a relay between two Transfer CFTs, PI 99 is saved into ST file attributes and can be forwarded to the other partner through the Store and Forward mode. The names of the initial and final users and applications are sent through PI 99 as G=RAPPL,H=SAPPL,L=SUSER,M=RUSER. The maximum size of data that can be transmitted in the PI 99 field is 512 characters.

PI 11, PI 12 and PI 13

The following parameters of the FPDU CREATE contribute to the unambiguous identification of the transfer: PI 11 (File Type), PI 12 (File Name), PI 13 (Transfer Identify).

Deactivate PeSIT CFT Extensions

To disable the PeSIT CFT Extensions functionality, set the server configuration option `Pesit.CftExtensions.Enabled` to `false`. The option is available on both SecureTransport EDGE and SecureTransport Server. Changing the value does not require a restart.

PeSIT message transfers

As of Update 5.5-20240530, SecureTransport can receive PeSIT messages (also known as PeSIT datagram transfers, corresponding to PI 91) coming from a PeSIT partner. In order to receive PeSIT messages, both a PeSIT transfer site and a transfer profile must be set up. If the transfer profile is not set as default, it must be added in the *IDM* field when sending a message in Transfer CFT.

Even though these messages have no actual payload attached, they are still visible on the *File Tracking* page. The content of a message can be viewed in the transfer's *Status Details* box. Some SecureTransport features that require a physical file, like File Archiving, are not supported because PeSIT messages are not stored as physical files on the file system. The following operations are currently not available for PeSIT messages: resubmit, retry, cancel, and manual and automatic acknowledgment. Sending PeSIT messages is not a supported feature at this time.

Receive PeSIT messages without triggering AR

SecureTransport can be configured to receive PeSIT messages without triggering any further processing.

1. Create a user account with the following elements configured:
    - PeSIT partnership with a user at the partner site
    - PeSIT transfer profile with Advanced Properties
2. Go to the transfer profile and perform the following:

a. Enable the **Receiving Message Parameters** checkbox.

b. Select the **Do not trigger processing** option.

SecureTransport will now use the configured transfer profile to receive PeSIT messages without triggering any subscriptions.

Receive PeSIT messages and trigger AR

SecureTransport can also be configured to use a received PeSIT message as a trigger for further Advanced Routing actions:

- *Configure AR flow without payload based on received PeSIT message* on page 981
- *Configure AR flow with payload based on received PeSIT message* on page 982

Transfer Log Maintenance application

The built-in SecureTransport application type, Transfer Log Maintenance, maintains the SecureTransport transfer log by exporting and cleaning up transfer log entries on a regular basis, following a schedule you define.

A Transfer Log Maintenance type application has the following features:

- User-definable schedule for transfer log daily backup and export or both. For more information, see *Configure a schedule for a maintenance application* on page 1.
- Application-specific parameters regarding the processing of transfer log entries include:
  - Expiration period for log entries until export and deletion or both
  - A condition to export the entries before deleting them or not
  - Delete exported files
  - Number of records per file
- Support for a dedicated export folder. By default the exported entries are stored in the following location:

  `<FILEDRIVEHOME>/var/db/hist/transfer-log`

For more information, see *Transfer Log Maintenance application* on page 856.

Server log

Use the *Server Log* page to view the contents of the SecureTransport log messages from the following SecureTransport components: Transaction Manager (TM), processes that implement the AS2, FTP, HTTP, SSH (SFTP), and SOCKS5 protocols, Administration Tool interface (ADMIN), and auditing (AUDIT).

When you open the *Server Log* page, SecureTransport automatically loads the server log records for the past hour. This behavior is determined by the `ServerLog.InitialLoading.Enabled.Admin` server configuration option. It is available on SecureTransport Edge and SecureTransport Server and accepts boolean values with a default of `true`. To disable the initial auto-loading of log entries on the page, set the option to `false`. When dealing with a large number of logs, you can filter the entries based on one or more criteria to speed up reporting.

By default, the server log entries are stored in the SecureTransport database. In an Enterprise Cluster deployment using an Oracle database, you can store the server log data in a separate external database from the rest of the SecureTransport data. See *Direct log data to separate Oracle databases on page 97*.

Each transfer line in the log display includes the following information:

- **TIME** – The date and time the entry was logged. Click on the link to display more detailed information.
- **LEVEL** – The severity level of the entry.
- **COMPONENT** – The name of the SecureTransport component that produced the entry.
- **THREAD** – The ID of the SecureTransport execution thread that produced the entry.
- **MESSAGE** – The primary log information.
- **SESSIONID** - The identifier of the login session associated with the entry. Click on the link to copy the ID into the **Session ID** field in the search criteria. This parameter is available only on SecureTransport Server.

On SecureTransport Server but not on SecureTransport Edge, two separate log messages are recorded for each client- or server-initiated transfer: one for the transfer start, and one for the transfer end. Both records include the following information in JSON format:

- **MESSAGE** - Indicates the start or end of a transfer.
- **STATUS** - The transfer status: *active*, *OK* (for successful transfers), or *error* (for failed transfers).
- **FILENAME** - The name of the transferred file.
- **USERNAME** - The name of the user account that performed the transfer.
- **SERVERNAME** – The name of the SecureTransport protocol server that was used to perform the transfer. Note that this parameter is displayed as *null* for SITs.
- **INITIATOR** - The initiator of the transfer: *client* or *server*.
- **DIRECTION** - The direction of the transfer: *inbound* or *outbound*.
- **CLIENTHOSTNAME** - The hostname of the client.
- **EDGEHOSTNAME** - The hostname of the Edge server. Note that this parameter is displayed as *null* for all SITs.
- **SERVERHOST** - The hostname of the server.
- **COREID** - An identifier used to track the file across different transfers and sessions.
- **TRANSFERID** - An identifier used to track the file during the transfer process. Click on the link to copy the ID into the **Transfer ID** field in the search criteria.
- **TRANSFERTYPE** - Indicates the transfer type: *file* or *message*.

The filtered log entries can be exported as a `.csv` file.

The following topics describe viewing, searching, exporting, and managing server log content:

- *Search and view server log contents on page 324*
- *Export the results of a server log search on page 325*
- *Log Entry Maintenance application on page 326*

Search and view server log contents

Use the following procedure to search and view server log content.

1. Select **Operations > Server Log**.
2. In the *Search* pane, specify your search criteria for the log entries to display.
    a. In the **Time Interval** drop-down list, choose the time frame:
        - Last Hour (default)
        - Last 4 Hours
        - Last 8 Hours
        - Last 12 Hours
        - Last 24 Hours
        - Last 48 Hours
        - Last 1 Week
        - Last 2 Weeks
        - Specific Date/Time Range – Allows you to choose a specific period.
    b. (Optional) In the **Account or Login** field, type the name of the account or login associated with the log entries.
    c. (Optional) In the **Thread** field, type the name of the thread associated with the log entries.
    d. (Optional) Under **Level**, choose the levels of the log entries:
        - TRACE
        - DEBUG
        - INFO
        - NOTICE
        - WARN
        - ERROR
        - FATAL
    e. (Optional) Under **Component**, choose the SecureTransport Server component associated with the log entries:
        - TM
        - AS2D
        - SSHD
        - Socks
        - ADMIN
        - AUDIT

- FTPD
- HTTPD
- PESITD

f. (Optional) In a clustered deployment, select the **Cluster Node** associated with the log entries. The nodes shown are those listed on the *Cluster Management* page. Click **Select Multiple** to select more than one node.

3. (Optional) Click **Advanced Search** to display additional criteria and specify the following:
   a. (Optional) Using the **ST Activity** checkboxes, specify whether to display inbound or outbound SecureTransport activities.
   b. (Optional) In the **Message** field, enter a string contained in the messages to display.
   c. (Optional) Use the link in the **Session ID** column to paste a session identifier into the **Session ID** field.
   d. (Optional) Use the link in the **Transfer ID** column to paste a session identifier into the **Transfer ID** field.
   e. (Optional) In the **Client** IP address field, type a string contained in the host name or IP address of the client associated with the transfer when the message to display was generated.
   f. (Optional) In the **Edge** IP address field, type a string contained in the host name or IP address of the SecureTransport Edge associated with the transfer when the messages to display was generated.
   g. (Optional) In the **Server** IP address field, type a string contained in the host name or IP address of the SecureTransport Server associated with the transfer when the messages to display was generated.
4. Click **GO**.

   The filtered log is displayed.

   Each log entry includes a time stamp, the log level, the names of the component and thread that logged the entry, the node IP address for a node in a cluster, the log message, and session and transfer identifiers.

Note In some cases, the type of a new application reported in the log message is a different form of the type you selected when you created the application.

Export the results of a server log search

Use the following procedure to export the results of a server log search.

1. Search for server log file entries as described in *Search and view server log contents on page 324*.
2. Click **Export Log**.

   A dialog box is displayed asking whether you want to open the file or save it to disk.
3. Specify whether to save the file or open the file, and then click **OK**.

Log Entry Maintenance application

The built-in SecureTransport application type, Log Entry Maintenance, performs the exclusive function of rotating server log files.

The main characteristic features of the Log Entry Maintenance application type are:

- User-definable schedule for transfer log daily backup or export. For more information, see *Scheduled downloads and tasks on page 674*.
- Application-specific parameters regarding the processing of server log entries include:
  - Expiration period for log entries until export or deletion.
  - Number of records per file.
- Support for a dedicated export folder. By default the exported entries are stored in the following location:

  `<FILEDRIVEHOME>/var/db/hist/log-entry`

For more information, see *Log Entry Maintenance application on page 839*.

Audit log

Use the *Audit Log* page to view, compare, and export log entries that SecureTransport records when any change is made to the SecureTransport configuration. The audit log entries record:

- Changes made using the Administration Tool
- Changes made using the administration REST API
- Changes due to user actions, such as new user enrollment and password change
- Changes that result from a change on another SecureTransport Server in a cluster or on another synchronized SecureTransport Edge
- Changes to configuration objects, such as accounts, business units, and network zones
- Changes to server configuration parameters, whether they are made on the *Server Configuration* page or on other Administration Tool pages

In an active/active Standard Cluster (SC), SecureTransport forwards audit log updates from the secondary servers to the primary server, so the audit log on the secondary servers includes local changes only and the audit log on the primary server includes changes to all servers in the cluster.

Many Administration Tool pages include a **Last Modified** link that you can use to display in the audit log the entry that records the last change for that page. From the audit log, you can compare that entry with the last one to see what changes were made or with any previous entry for that object or parameter.

You can filter the log using one or more of eight criteria to find an entry.

Each line in the log display includes the following information:

- Time - The date and time the entry was logged. This column includes a drop-down menu indicated by an inverted caret that you can use to display more detailed information or compare log entries.
- User Name - The name of the administrator or user who made the change.
- Remote Address - The IP address of the client that made the configuration change – either a browser running the Administration Tool or a program using the administrator resources of the REST API.
- Object Type - The type of the object changed. Possible values are:
    - `Account` – for user account, unlicensed user account, service account and account template
    - `Administrator`
    - `AdministrativeRole`
    - `Application`
    - `BusinessUnit`
    - `Certificate` – for certificate and trusted CA
    - `CertificateSigningRequests`
    - `ClusterNode`
    - `HolidaySchedule`
    - `LdapDomain`
    - `LdapHomeFolderPrefix`
    - `LdapUidRangeMapping`
    - `MailTemplate`
    - `NetworkZone`
    - `PasswordVault`
    - `Route`
    - `ServerConfigurationParameter` – for changes that are not represented internally as objects by SecureTransport
    - `SiteTemplate`
    - `UserClass`
- Object ID - A unique identifier for the object changed. For most configuration objects, the object ID is an internal ID.

  For server configuration parameters, the object ID includes the server configuration parameter name, a node ID, and a profile ID. In a cluster, the node ID (`mNode`) identifies the server where a local server configuration parameters is changed. Otherwise, the value is `UNSPECIFIED`. SecureTransport 5.5 does not implement the feature that uses the profile ID (`mProfile`), so the value is always `Default`.

- Object Name - The name of the object changed.
- Operation - The type of operation that changed the object. The value can be `Create`, `Delete`, `Overwrite`, or `Update`.

  When you add a user class, it is added as the first user class as shown on the *User Classes* page. The audit log shows an update for each user classes, because adding a class changes the `Order` attribute for every class.
- Comment - Additional information added by SecureTransport or the administrator who made the change.
- All Audit Log actions are reported into the Server Log as audit information messages and administrators can configure the audit messages to be logged into the SecureTransport database or a flat file using standard `log4j` configurations. The audit information messages in the Server Log can be machine read and analyzed, while the messages on the *Audit Log* page require interaction to identify what property has changed. The audit messages in the Server Log can also be filtered by the AUDIT component filter. For additional Server Log information, refer to *Server log* on page 322.

The audit information messages are based on the `auditLogEntry` object data. The audit information messages are in the following format:

```
<username> <operation> <objectType> <objectName> <description>
[<list of properties>]
```

Where:

- `<username>` - The name of the user which performed the audit operation.
- `<operation>` - The name of the operation performed. It could be one of the following – create, update, delete, or create_or_update.
- `<objectType>`- The object type being audited.
- `<objectName>` - The name of the object being audited.
- `<description>` (optional) - The description defined for the given `auditLogEntry`. If a description is defined, it will be displayed.
- `[list of properties]` - Contains the mapping between the property name and the old and new value. Only changed properties will be visible in this map. If there is nothing to compare with, all new properties will be visible.

The following topics describe managing the audit logs:

- *Search and view audit log contents* on page 329 - Describes viewing and searching the audit log contents.
- *Enable or disable audit logging* on page 330 - Provides how-to instructions on enabling and disabling audit logging.
- *Export the results of an audit log search* on page 330 - Provides how-to instructions on exporting the results of an audit log search.
- *Add or edit an audit log entry comment* on page 330 - Provides how-to introductions on adding or editing an audit log entry comment.

- *Display audit log entry details* on page 330 - Provides how-to instructions on displaying audit log entry details.
- *Compare audit log entries* on page 331 - Provides how-to instructions on comparing audit log entries.
- *Link to the audit log* on page 331 - Describes the SecureTransport Administration Tool links to the audit log.
- *Audit Log Maintenance application* on page 332 - Describes the Audit Log Maintenance application.

Search and view audit log contents

Use the following procedure to search and view the audit log contents.

1. Select **Operations > Audit Log**.

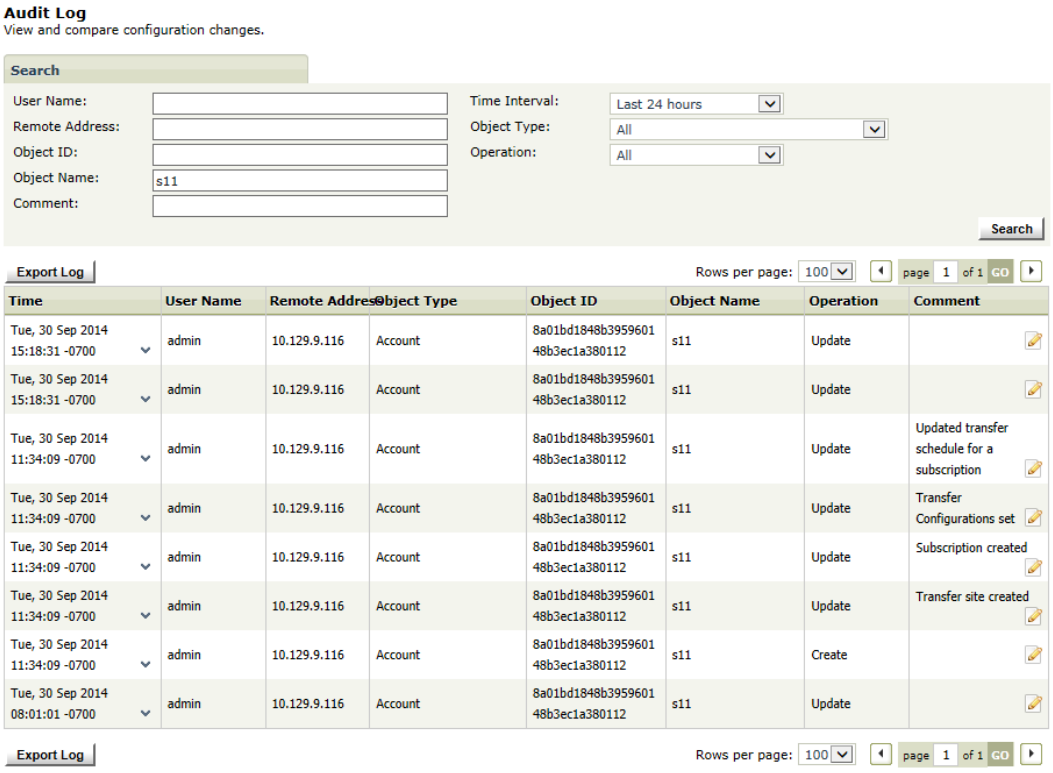
   The *Audit Log* page is displayed.

2. In the *Search* pane, specify the search criteria.

   The search fields include all the information displays in the audit log.

3. Click **Search**.

   The filtered log is displayed.



Enable or disable audit logging

Audit logging is enabled by default. You can disable or enable it by setting the following server configuration parameters:

- To disable logging of changes made by the Administration Tool, set the `AuditLog.Enabled.Admin` server configuration parameter to `false`. The default value is `true`.
- To enable logging of changes made by the Transaction Manager, set the `AuditLog.Enabled.TM` server configuration parameter to `true`. The default value is `false`.

To change these parameters, see *View and change server configuration parameters* on page 334.

Export the results of an audit log search

Use the following procedure to export the results of an audit log search.

1. Search the audit log described in *Search and view audit log contents* on page 329.
2. Click **Export Log**.

   A dialog box is displayed asking whether you want to open the file or save it to disk.
3. Specify whether to save the file or to open the file, and then click **OK**.

In addition to the fields displayed on the *Audit Log* page, the exported audit log contains the User-Agent string of the client that made the change.

Add or edit an audit log entry comment

Only the administrator that made the change represented by the audit log entry can add or edit a comment.

1. Click the Edit icon () in the **Comment** column.
2. Type or update the comment.
3. Click the Save icon ().

Display audit log entry details

Use the following procedure to display the details of an audit log entry.

- From the menu in the **Time** column, select **Details**.

  SecureTransport opens an *Object Detail* window that lists all attributes of the object at the time of the audit log entry with the attribute values.

Compare audit log entries

Use the following procedure to compare audit log entries.

1. From the menu in the **Time** column, select **Compare with** or **Compare with previous**.
2. If you selected **Compare with**, SecureTransport displays a *Compare with Another Entry* window.
3. If previous audit log entries are listed, select one and click **OK**.
4. If you selected **Compare with previous** and there is a previous audit log entry for that object, the result is the same as selecting the previous entry in the *Compare with Another Entry* window.

   If there is no previous audit log entry for the object, SecureTransport displays a message.
5. SecureTransport opens an *Audit Log Entry Comparison* window.

   The *Audit Log Entry Comparison* window lists all attributes and values of the object with one column for each audit log entry. Timestamps in the column headers identify the entries. The older entry is on the left. A colored background indicates attributes with different values.

Link to the audit log

The following Administration Tool pages have a **Last Modified** link that gives the date and time of the last update to any object listed:

- Operations menu:
  - Cluster Management – Enterprise Cluster only, indicates changes to the configuration of the cluster nodes only
  - Server Configuration
- Setup menu:
  - Local Certificates
  - Trusted CAs
  - Internal CA
  - Holiday Schedule
  - Mail Template Repository
  - File Archiving
  - Network Zone List
  - Edit Network Zone entry
- Authentication menu:
  - Login Settings
  - LDAP Domains

- LDAP Domain
- User Type Ranges
- Home Folders
  - Accounts menu:
    - User Account
    - Unlicensed User Account
    - Service Account
    - Administrators
    - Edit Administrator
    - Administrative Roles
    - Edit Administrative Role
    - Account Template
    - Passwords Files
    - Edit Password File
    - Business Units
    - Business Units Settings
  - Access menu:
    - User Classes
  - Application:
    - Applications
    - Application Details

If there has been no change to the object or objects since SecureTransport was installed or upgraded to a version that includes the audit log, the link indicates `No tracked change`.

SecureTransport stores configuration that you change on other pages of the Administration Tool in sever configuration parameters. Some pages save values that are not changed when you make a change, so the audit log might include entries for the corresponding server configuration parameters in addition to the ones for the fields you changed.

When you click a link on a page, SecureTransport opens the *Audit Log* page and displayed the entry for the update that the link represents.

Audit Log Maintenance application

An application of the built-in SecureTransport application type Audit Log Maintenance deletes old audit log entries periodically and can export them to a file. By default, SecureTransport has an Audit Log Maintenance application that runs at midnight on the first day of each month and deletes audit log entries that are six months old after saving them in a file in the `<FILEDRIVEHOME>/var/db/hist/audit-log` directory

For more information, see *Audit Log Maintenance application* on page 828.

Server configuration

SecureTransport server configuration consists of all the information the server and its components require to operate. You establish most of the server configuration by setting fields on the various pages of the Administration Tool. In a SecureTransport cluster, the servers store server configuration parameters in a shared database for an Enterprise Cluster (EC) and in synchronized embedded databases for a Standard Cluster (SC). The server configuration parameters include both shared, cluster-wide parameters and parameters for individual servers. When you change a shared parameter, all the servers in the cluster get the new value.

To support operation of the components that use configuration files, the server copies the configuration from the database into those files when it starts and when you change a parameter on any node of the cluster. It is not necessary to bounce the servers manually to propagate the change.

Use the *Server Configuration* page to view configuration parameters that are stored in the database and change those that are not set elsewhere in the Administration Tool. This page also includes access to pages you can use to view or update server configuration files on your local computer, synchronize configuration files, and export and import server configuration.

The following topics provide additional server configuration information:

- *Editable server configuration parameters on page 333* - Provides how-to instructions on setting editable server configuration parameters.
- *Local server configuration parameters on page 334* - Describes the local server configuration parameters.
- *View and change server configuration parameters on page 334* - Provides how-to instructions for viewing and changing server configuration parameters.
- *Update configuration files on page 336* - Provides how-to instructions on updating server configuration files.
- *Export and import server configuration on page 337* - Provides how-to instructions on exporting and importing server configurations.

Editable server configuration parameters

You set many server configuration parameters in fields elsewhere in the Administration Tool. You can view, but not change those parameters in the *Server Configuration* page.

Server configuration parameters that you can change have an Edit icon () in the **Edit** column of the list.

Note When you hover over the description of an editable server configuration parameter, addition description information is displayed. Before editing a server configuration parameter, refer its description for configuration parameters information.

Local server configuration parameters

Most of the parameters included in the *Server Configuration* page apply to all servers in the cluster. The parameters that apply to the local SecureTransport Server and are not copied to other servers are marked with a check in the **Local** column.

View and change server configuration parameters

Many SecureTransport server configuration parameters are stored in the database. In a cluster, any change you make to any shared parameter is automatically copied to all nodes in the cluster.

The *Server Configuration* page, accessed by selecting **Operations > Server Configuration**, shows all configuration parameters that are stored in the database. You can view the values of all parameters and you can edit the values of some of them. You set the values of parameters you cannot edit on the *Server Configuration* page using fields on other pages of the Administration Tool.

The following topics provide how-to instructions for searching, paging through, and changing parameters:

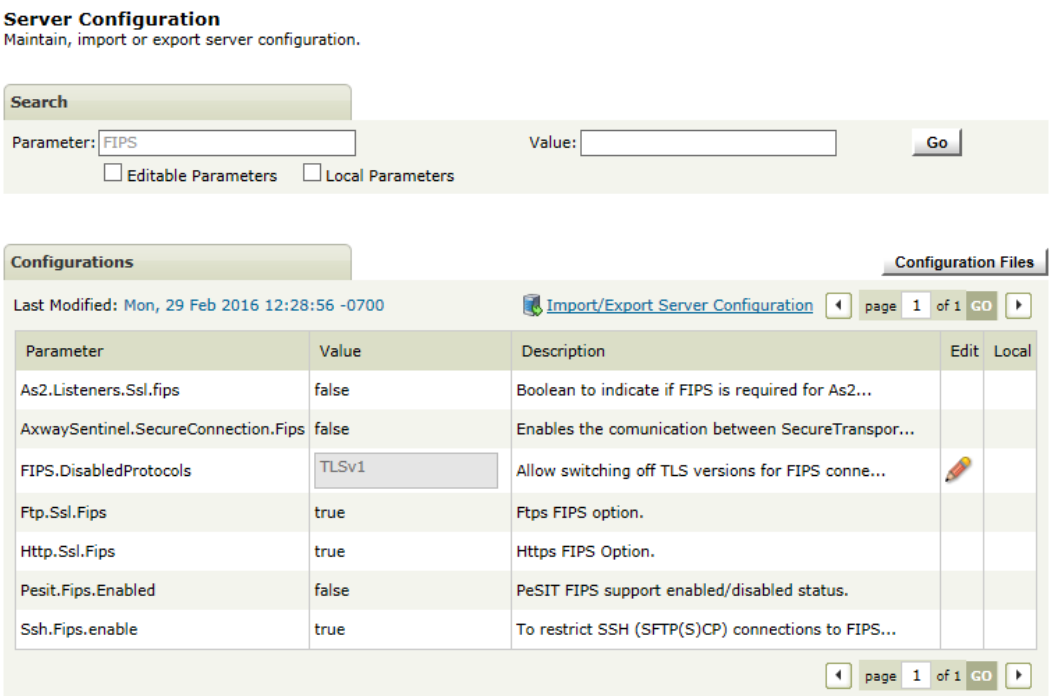
- *Search for a parameter* on page 334
- *Page through the parameter list* on page 335
- *Change a parameter value* on page 335

Search for a parameter

You can filter the list of parameters using fields in the *Search* pane.

1. To display only parameters that contain a string in their name, type that string in the **Parameter** field. The parameter name search is not case sensitive.
2. To display only parameters that contain a string in their value, type that string in the **Value** field. The parameter value search is case sensitive.
3. To display only parameters you can edit, select **Editable Parameters**.
4. To display only local parameters, select **Local Parameters**.
5. Click **Go**.

The filtered list is displayed.



Note Drag the resize handle of a field in the **Value** column to see or change all the text.

Page through the parameter list

If there are more than 100 parameters in the filtered list, they are displayed on pages.

1. To display to the next or previous page, click the forward or back arrow.
2. To display a page, type the page number in the **page** field and click **GO**.

Change a parameter value

If a parameter is editable, you can change its value. In many case, the valid values are include in the Description column.

1. In the Edit column, click the Edit icon ( ).
2. Type the new value in the **Value** column.
3. In the Edit column, click the Save icon ( ).

   SecureTransport saves the value to the database. If the parameter applies to all nodes in the cluster, SecureTransport copies it to the other nodes.
4. To cancel an edit, click **Go** in the *Search* pane.

Note If you edit the value of a second parameter before saving the value of the first, save each value in turn, waiting for each save operation to complete before saving the next.

Update configuration files

SecureTransport stores the following server configuration files in the database:

- `brules.xml`
- `FileServicesInterfaceRegistry.xml`
- `mime.types`
- `sentinel-returncode-translation.xml`
- `ssl.csr.conf`
- SSO Configuration Files - The Single Sign-On (SSO) related configuration files. For more information, refer to *Single Sign-On (SSO) and Single Logout (SLO)* on page 426.

You can use the *Server Configuration Files* page to make changes on SecureTransport Server to the files listed, save the changes to the database, propagate the changes to all servers in a cluster (if applicable), and configure SSO for end-users and administrators. For information about how to configure the SSO functionality in SecureTransport, see *Single Sign-On (SSO) and Single Logout (SLO)* on page 426.

1. On the *Server Configuration* page, select **Configuration Files**.

   The *Server Configuration Files* page is displayed.
2. Download the file to update. (For example, right-click the file name link in your browser and selected **Save Link As** or **Save Target As**.)
3. Update the file using a editor on your local computer.
4. Click **Browse** and select the file or type the path to the file in the **Selected File** column.
5. Select the checkbox that corresponds to the updated file(s).
6. Click **Upload**.
7. Restart the `admind` service in order to apply the configuration changes.

In cluster deployment, SecureTransport will upload the selected files to the respective Server and copy the files to all the other Servers. However, the Server nodes will load the updated configuration after a restart of the `admind` service on each. You can apply that same procedure to update configuration files across SecureTransport Edges.

Notes:

- SSO configuration files are replicated on Enterprise Clusters and Standard Clusters after they are uploaded in ZIP format.
- Alternately, you can edit these configuration files on the server and then click **Synchronize** on the *Server Configuration Files* page for that server. The server saves the file in the database and copies the files to all the other servers in the cluster, and the servers load the updated configuration immediately.
- For SSO configuration files, the **Synchronize** button will update the SSO-related configuration files in `<FILEDRIVEHOME/conf/sso>` directory with the database. This will only affect the current node.

- Only ZIP format is accepted for SSO configuration files import.
- Do not put the configuration files in a sub-directory inside the ZIP file.
- You can list all SSO-related configuration files, by clicking on the **Plus** () icon. You can hide the same files, by clicking the **Minus** () icon.
- You can download a single SSO-related configuration file, by clicking on the name of the file. You can also download all SSO files in ZIP format, by clicking on the **SSO Configuration Files** link.
- When importing SSO Configuration Files on a SecureTransport Edge, make sure that the ZIP file contains the `sso-admin.xml`, otherwise the import will not be successful.
- When importing the SSO Configuration Files in a Standard Cluster, make sure that the Transaction Manager service is running on the current node. For more information, see *Standard Cluster synchronization* on page 364.

Export and import server configuration

The objective of the Export/Import functionality is to allow for the restoration of an environment. It is not a supported way to migrate or upgrade an environment.

This feature allows you to export your environment configuration data to a compressed file which you can later use to return to a known good configuration or recover from a failure.

The export file includes the following data:

- bin, brules and conf subdirectories of the *<FILEDRIVEHOME>* directory
- XML files in the database that record the system configuration parameters, including those set on the *Server Configuration* page and elsewhere in the Administration Tool, plus the holiday schedule
- Local certificates, certificate signing requests, and trusted CAs.

The export files does not include:

- Database settings like port, password, and so forth.

When you import a server configuration file, all existing configuration on the environment is replaced by the configuration contained in the import file.

If an improper configuration file is imported (for example, a blank file or an export from another server), no error message is displayed and the configuration files are overwritten. Use system import with caution.

Exporting and Importing Cluster Configuration

Configuration files can be imported to the same SecureTransport environment where the configuration was exported.

Channel configurations are specific to environment in which they are created, so exporting and importing complete configurations as is won't work.

Importing a configuration file created on one Enterprise Cluster (EC) into another EC cluster is not supported.

Upon import, SecureTransport copies shared server configuration parameters to every node in the cluster. Therefore, you only import the shared server configuration once for the whole cluster. Database settings like port, password, and so forth are not intended to be exported because database settings are node specific.

The imported files overwrite the existing files.

the database is updated with the parameter values from the imported files, the imported files are modified to support changes made to SecureTransport, and

the importer adds any new properties needed for the features introduced in the current version of SecureTransport. At the same time, the importer preserves any custom changes you have made to the imported files, applying them to the current version of SecureTransport.

Export methods

There are two methods to export the server configuration files:

using the *Import or Export Server Configuration* page

command line utility. Use the command line utility to customize which folders and files will be exported. For more information, see *Export server configuration files from the command line* on page 341 and *Import server configuration files from the command line* on page 342.

Required permissions

Any administrator with import and export configuration privileges can access the *Import or Export Server Configuration* page to import or export the server configuration information. Any administrator who can access the server can use the command line to import or export server information.

Procedure

You must supply a password that SecureTransport uses to encrypt sensitive information such as private keys and custom attributes during the export process. When you import the server configuration information, you must type the password to import the server configuration files and decrypt the sensitive information.

The imported files overwrite the existing files, the database is updated with the parameter values from the imported files, the imported files are modified to support changes made to SecureTransport, and the importer adds any new properties needed for the features introduced in the current version of SecureTransport. At the same time, the importer preserves any custom changes you have made to the imported files, applying them to the current version of SecureTransport.

The following topics provide how-to instructions for importing and exporting the server configuration, certificates, and messages:

- *Export user limit messages* on page 339
- *Export and import Internal CA files* on page 339

- *Export server configuration using the Administration Tool* on page 340
- *Export server configuration files from the command line* on page 341
- *Import server configuration files using the Administration Tool* on page 342
- *Import server configuration files from the command line* on page 342

Export user limit messages

To export messages defined on the *Limit User Access* page for successful import, add lines for all files that match `lib/msgs/msg.*Class*.*` pattern to `<FILEDRIVEHOME>/conf/export.conf` before exporting.

Export and import Internal CA files

For SecureTransport 5.0 and later, the Internal CA certificate is exported with system export and with account export. In both cases, the private key for the CA is not exported. You cannot use an imported Internal CA to sign additional certificates without the correct private key. To preserve the Internal CA private key, configure server export and import to include the private key. Perform the following procedures before you export the system configuration files. For more information on exporting and importing accounts, see *Export and import accounts* on page 686. For more information about exporting and importing the Internal CA, see *Manage the internal CA* on page 57.

Export the Internal CA with the private key

1. Add the following lines to the `<FILEDRIVEHOME>/conf/export.conf` file:

   ```
   lib/certs/db/ca-crt.pem
   lib/certs/db/ca-key.pem
   lib/certs/db/index
   lib/certs/db/serial
   ```

2. Export the system configuration.

   It contains the Internal CA with its private key.

Import the Internal CA with the private key

1. Delete the temporary Internal CA generated during installation, so that the Internal CA is not incorrectly imported as CA-old.
2. Import the system configuration.

Export server configuration using the Administration Tool

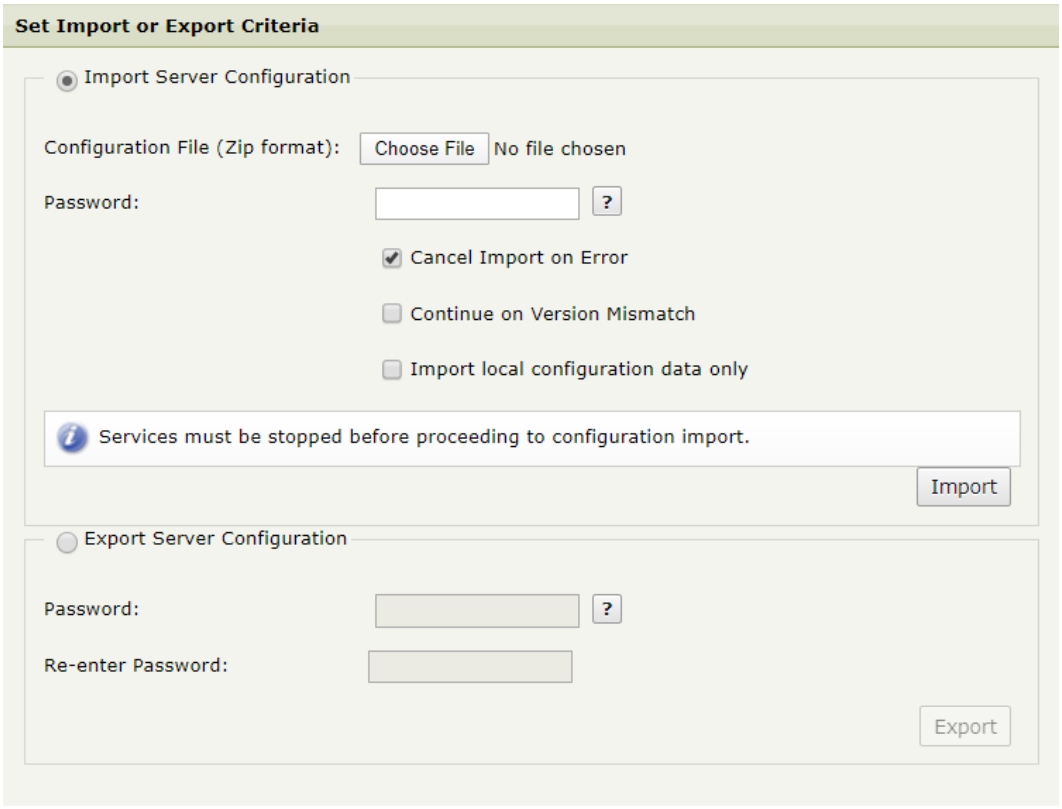
You can export and download server configuration using the SecureTransport Administration Tool. The ZIP file is also automatically backed up on the server as `<FILEDRIVEHOME>/var/tmp/export_configuration.zip`. You cannot specify the file name and location on the server, and the back up file overwrites any existing back up file.

When you export the server configuration from the Administration Tool, SecureTransport uses the file `<FILEDRIVEHOME>/conf/export.conf` to read the list of configuration files to be exported. In addition, the files in the `<FILEDRIVEHOME>/brules/local/wptdocuments` directory are always included.

You can control the file name and location, and the list of files to be exported by using the command line tool to export your server configuration files. For more information, see *Export server configuration files from the command line* on page 341.

1. On the *Server Configuration* page, click **Import/Export Server Configuration**.

   The *Import or Export Server Configuration* page is displayed.



2. Select **Export Server Configuration**.
3. Type the file password in the **Password** and **Re-enter Password** fields.
4. Click **Export**. The *Export Complete* prompt is displayed. The ZIP file is save as `<FILEDRIVEHOME>/var/tmp/export_configuration.zip`.

5. To download the ZIP file to your local computer, click **Download Exported Configuration**. The *File Download* dialog box is displayed.
6. Click **Save** to save the file to a new location or click **Open** to view the contents of the ZIP file.

   To save the file, select the location for the exported server configuration data and click **Save**. You are returned to the Import or Export Server Configuration window.

   If you clicked **Open**, the ZIP file attempts to open and display the contents of the file in a new window.

   If you do not want to download the ZIP file, click **Cancel** to return to the *Import or Export Server Configuration* page.

Export server configuration files from the command line

You can export sever configuration information using a command line tool. When you are using the tool to export a server configuration, you must specify the file name and location that contains the exported configuration.

You can also specify which files you want to export by creating a list file with a `.conf` extension. This file contains the list of configuration files you want to export. This is useful when you have customized SecureTransport and need to export additional files to those listed in the default export list. The default export list is located in `<FILEDRIVEHOME>/conf/export.conf`. Do not modify this file, but create a file with a new name if you need to make a new export list.

SecureTransport provides a script called `system_export` that you can run from the command line to export the server configuration information to a ZIP file. The script has the following options:

- `-exf=<export_file>` where `<export_file>` is the file name and location of the ZIP file. You must specify the file name.
- `-ex1=<export_list>` where `<export_list>` is the file containing a list of all files to be exported. The `<export_list>` file name is relative to `<FILEDRIVEHOME>`. The default is `conf/export.conf`.
- `-help` displays the command format and options.

Note If you run `system_export` without specifying any options, the help message is displayed.

During the export process, you are prompted for an export password. Later, when you import the exported configuration from the command line, you must use the same password for the import process. The following steps illustrate an example sever configuration export:

1. Change to the `<FILEDRIVEHOME>/bin` directory.

   If you installed SecureTransport on Windows, you can run the command without changing to the `/bin` directory.
2. Type one of the following commands:
   - `./system_export -exf=<export_file>` for UNIX-based systems
   - `system_export -exf=<export_file>` for Windows

where <export_file> is the name and location of the ZIP file you are creating.

3. When prompted, type a password for the exported information.
4. Confirm the password by typing it again when prompted.

   The exported file is created in the specified location.

Import server configuration files using the Administration Tool

Using the Administration Tool, you can import either server configuration information for a cluster or only the local server configuration information for a single server. You must know the password entered during the server configuration export.

1. On the *Server Configuration* page, click **Import/Export Server Configuration**.
2. On the *Import or Export Server Configuration* page, select **Import Server Configuration**.
3. Select the **Configuration File** by clicking **Choose File**. The file must be in the zip format.
4. Type the **Password** that was used for the export.
5. Select the options:
   a. Select **Cancel Import on Error** to stop the import process if any error is encountered. This option is selected by default. If the import process is stopped, no changes are made to the server. If you clear this option and the password does not match, the import completes with a warning that information from the zip archive could not be decrypted.
   b. Select **Continue on Version Mismatch** to import server configuration from a different version of SecureTransport.
   c. Select **Import local configuration data only** to exclude cluster configuration data, for example, when you are importing configuration data into a SecureTransport Server that is in an existing cluster. This option is not available if **Continue on Version Mismatch** is selected.
6. Click **Import**.

   The *Import Complete* message is displayed and the server configuration import is successful. If you did not select **Import local configuration data only**, the imported cluster configuration data is propagated to all servers in the cluster.

Note When you import a server configuration, the process overwrites the current configuration. If an improper configuration file is imported (for example, an empty file), no error message is displayed and the configuration files are overwritten.

Import server configuration files from the command line

Note Before you import the server configuration from the command line, stop all services except the Administration Tool service.

SecureTransport provides a command named `system_import` that can be run from the command line to import information from the ZIP file. The command requires that the Administration Tool service is running on the SecureTransport server where you run the command.

In a Standard Cluster (SC), run the `system_import` command on the primary server. When the import completes, the updates are automatically synchronized to the other servers in the cluster.
The script comes with the following options:

- `-exf=<export_file>` where `<export_file>` is the file name and location of the ZIP file. You must specify the file name.
- `-coe=<true | false>` where when set to `true`, the import stops if an error occurs and no changes are made to the server ("cancel on error"). If set to `false` the import continues if an error occurs. The default setting is `true`. If set this option to false and the password does not match, the import completes with a warning that information from the zip archive could not be decrypted.
- `-ivm=<true | false>` where when set to `true`, the import continues even if there is a version mismatch. Setting this option to `false` stops the import if there is a version mismatch. The default setting is `false`.
- `-ilo` means import only local configuration parameters. This options requires `-ivm=false`.
- `-help` displays the command format and options.

Note If you run `system_import` without specifying any options, the help message is displayed.

1. Change to the `<FILEDRIVEHOME>/bin` directory.

   If you installed SecureTransport on Windows, you can run the command without changing to the `/bin` directory.

2. Type one of the following commands:

   `./system_import -exf=<export_file>` for UNIX-based systems

   `system_import -exf=<export_file>` for Windows

   where `<export_file>` is the file name and location of the ZIP file. You must specify the file name.

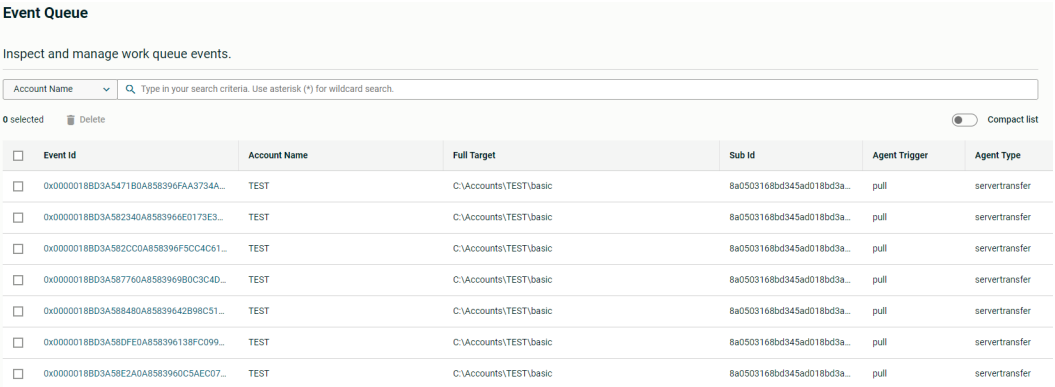
3. When prompted, type the password for the ZIP file. This is password created when the file was exported.

   The server configuration information is imported into SecureTransport.

When you import a server configuration, the process overwrites the current configuration. If the configuration file is corrupted or blank, no error message is displayed and the configuration files are overwritten.
If you import a wrong configuration, and then immediately try to import the correct one, the command displays an error message regarding the database password. You must restart SecureTransport after each system import.

Event Queue

The *Event Queue* page provides operational flexibility for emergency maintenance. You can inspect the internal details of events in the current SecureTransport TM work queue, and delete unwanted pull or stuck events. This page is accessible only to master administrators who have access to the *Usage Monitor* and *File Tracking* pages.



Search for and inspect events

Use the following procedure to search for events and inspect their properties for unusual patterns.

1. Choose your search category:
    - **Account Name**: displays only events associated with accounts matching the search criteria.
    - **Full Target**: displays only events with an absolute *fullTarget* path matching the search criteria.
2. Enter your search criteria (case-sensitive).
3. Click on an Event ID to display the event properties.

Note that events are processed very quickly and some may not exist by the time you have opened them for inspection.

Delete events

Use the following procedure to delete events.

Caution Deleting events may result in unfinished transfer and routing processing.

1. Select one or more events by clicking the checkbox on the left.
2. Click **Delete**.

Set queue size limit and warnings

Use the following server configuration options to set the maximum size for the Event Queue and control the logging of warning messages for changes in the queue size:

- `TransactionManager.ThreadPools.ThreadPool.EventMonitor.maxQueueSize` - Set a limit on the number of events in the queue. The default value is `1024`.
- `TransactionManager.ThreadPools.ThreadPool.EventMonitor.maxQueueSize.usageAlertsLogging` - When enabled, warnings are logged each time the queue size increases or decreases by 10%, given that the queue has reached 50% of the capacity configured in the option above. The default value is `disabled`.

Support tool

The Axway SecureTransport support tool collects information about SecureTransport and its host operating system and saves it in a support information file that you can send to Axway Global Support to help them diagnose an issue.

You use the *Support Tool Configuration* page to specify the information that the support tool saves and where it saves the file. You run a command line utility to create the support information file. If needed, you can add other information to the file by editing a custom script.

The following topics describe how to configure, customize, and run the support tool:

- *Configure the support tool* on page 345 - Provides the how-to instructions for configuring the support tool.
- *Add custom information to the support information file* on page 347 - Describes how to add custom information to the support information file.
- *Run the support tool* on page 347 - Provides the how-to instructions for running the support tool.

Configure the support tool

Use the *Support Tool Configuration* page to specify the information that the support tool saves in the support information file and where it saves the file.

1. Select **Operations > Support Tool** to display the *Support Tool Configuration* page.
2. Specify the fields listed below.
3. Click **Save**.

   SecureTransport saves the configuration and replicates it to all nodes in a Server cluster or to all synchronized Edge servers.
4. Under *Files and Folders to Save*, add, remove, or edit the path names of files and folders to include in the support tool information file. Relative path names are relative to `<FILEDRIVEHOME>`. You can also add absolute path names.

SecureTransport saves each the change and replicates it to all nodes in a SecureTransport Server cluster or to all synchronized SecureTransport Edge servers.

The following topics describe the support tool fields and how-to instructions for excluding files:

- *Fields* on page 346
- *Exclude files* on page 346

Fields

This topic includes information about the fields you must complete.

- **Support Access Code** - The support tool includes the value of this field in the name of the support information file. Enter the Support Access Code for the support contact that covers this system. Axway Global Support uses the code in the file name to make sure the file is handled correctly.
- **Output Directory** - Enter the path to the directory where the support tool writes the support information file. The default is `${FILEDRIVEHOME}/support` where `${FILEDRIVEHOME}` represents the SecureTransport installation directory.
- **Collect ...** - Select the checkbox for each category of information that the support tool saves. For the log information, specify the period by selecting a predefined period for the list or specifying the start and end dates and times. The information available for collection is different on SecureTransport Server and SecureTransport Edge.
- **Collect files and folders** - Select this to specify that the support tool save the files listed in the **Files and Folder to Save** table.
- **Files and Folders to Save** - If **Collect files and folders** is selected, the support tool saves the contents of the files listed in this table in the support information file.

  To add a file, click **Add File or Folder**, type the file or folder name in the File/Folder Name column, and click the Save icon () in the right column. To remove files from the list, select the files in left column, and click **Remove**. To change a file or folder name in the list, click the Edit icon () in the right column, type the file or folder name in the File/Folder Name column, and click the Save icon ().

Exclude files

The support tool does not save the contents of files or folders whose names are listed in the `SupportTool.ExcludeFilesFoldersList` server configuration parameter. By default, the excluded names are `certs` and `passwd`.

To add or remove names from the excluded files list, see *Change a parameter value* on page 335.

Add custom information to the support information file

The support tool is implemented as an executable script, `<FILEDRIVEHOME>/bin/collect_support_information`, that collects the information that is specified on the *Support Tool Configuration* page and saves it in the support information file.

To add custom information to the support information file, edit the `<FILEDRIVEHOME>/bin/custom_collect_support_information` script. The main script calls this script before it creates the support information file from the collected information.

By default, the custom script does nothing. Your custom script must write the information that you need to include in the support information files to the support directory that the main script writes to. The script can create files and directories in the support directory. The last action that the main script takes is to combine all the files in the support directory into the support information file.

When you write a custom script, you can use the following defined environment variables:

- **`${HOST}`–The host name of the system**
- **`${SUPPORT_DIR}`–The full path name of the directory where the custom script must write its information**

The following example `<FILEDRIVEHOME>/bin/custom_collect_support_information` script illustrates writing a file to the support directory.

```
# Collect user files
#
USER_HOME="/home/users"
USERS="partner2 partner7"

echo "=== Saving user files ==="
tar -cf "${SUPPORT_DIR}/users_${HOST}.tar" -C "${USER_HOME}" "${USERS}"
```

Run the support tool

The support tool is implemented as an executable script. To create the support information file, run `<FILEDRIVEHOME>/bin/collect_support_information`. The script saves the information configured on the *Support Tool Configuration* page in the configured output directory. The file name is `support_<supportAccessCode>_<hostName>_<timestamp>.tar.gz where`:

- `<supportAccessCode>` is the value of the **Support Access Code** field on the *Support Tool Configuration* page.
- `<hostName>` is the host name of the system that the support tool is run on.
- `<timestamp>` is the UNIX time (the number of seconds that have elapsed since midnight Coordinated Universal Time on January 1, 1970).

To write the support information file to a different directory from the one specified in the **Output Directory** field on the *Support Tool Configuration* page, use the following command:

```
<FILEDRIVEHOME>/bin/collect_support_information -d
<outputDirectory>
```

If the output directory does not exist, the support tool creates it.

The support tool can run when the SecureTransport database in not available, but it cannot collect the following information that is stored in the database:

- Server log
- File tracking
- Audit log
- Server configuration

During execution, the support tool outputs status messages and a final message that path name of the support information file. You can use SecureTransport to push the file to another location.

Note The TM thread dump can slow the operation of the TM and, more significantly, the TM heap dump can prevent the TM from processing events for 30 minutes or more, depending on the configured heap size. It is best to run the support tool when there is no load on SecureTransport.

Run the support tool automatically when the TM runs out of memory

You can configure SecureTransport to run the support tool automatically when the Transaction Manager (TM) fails with an `OutOfMemoryError`. SecureTransport runs the support tool before it restarts the TM.

1. Edit the `<FILEDRIVEHOME>/bin/crash_tm` file and comment out a line so that it is:

   ```
   # collect_crash_info="false"
   ```

2. If **Collect TM heap dump** is selected on the *Support Tool Configuration* page, edit the `<FILEDRIVEHOME>/bin/start_tm_console` file and comment out a line so that it is:

   ```
   # disableHeapDumpOnOutOfMemoryError="true"
   ```

3. Restart the Transaction Manager on all clustered SecureTransport Servers or all synchronized SecureTransport Edge servers using the `stop_tm` and `start_tm` commands in `<FILEDRIVEHOME>/bin`.

SecureTransport runs the support tool using the configuration from the *Support Tool Configuration* page.

Note Due to a technical limitation, when the `crash_tm` script runs the support tool, the support information file does not include the TM thread dump.

Directory browsing

By setting up a directory structure, you can access system drives and network mounts on Windows.

Set up the structure for directory browsing

Use the following procedure to set up the structure for directory browsing.

1. Enable browsing for system drives.

   For system drives, create a directory under`<FILEDRIVEHOME>\..\cygwin\drives` with the drive letter as a name, and give permission to everyone. For example, to enable browsing of the `C` drive, create the following folder:

   ```
   <FILEDRIVEHOME>\..\cygwin\drives\c
   ```

2. Enable browsing for network shares.

   For network shares `\\<hostname>\<sharename>`, create a directory structure like the following:

   ```
   <FILEDRIVEHOME>\..\cygwin\net\<hostname>\<sharename>
   ```

   and give permission to everyone for this directory. `<hostname>` and `<sharename>` are both required.

   For example, to enable browsing of network share with path:

   ```
   \\myhost\my-shared-folder
   ```

   create the following folder:

   ```
   <FILEDRIVEHOME>\..\cygwin\net\myhost\my-shared-folder
   ```

Note System drives in the user's home folder are enabled for browsing by default.

Server backup

To minimize the risk of data loss, perform regular backups of the SecureTransport Server and SecureTransport Edge data.

When performing a server backup, you must back up the following files and directories in `<FILEDRIVEHOME>`:

- `conf/`
- `lib/certs/`
- `brules/conf/brules.xml`
- `brules/local/`

- `bin/agents/`
- `var/db/mariadb/` (for servers using the embedded database)

Include the following files if changes have been made after the initial installation of SecureTransport:

- `lib/msgs/`
- `share/ftdocs/`

Include the user account home folders.

Add the following files to keep existing log files, statistics, and MDN receipts:

- `var/logs/`
- `var/db/hist/logs/`
- `var/db/stats/`

For SecureTransport Server with an external database, back up the database.

For SecureTransport Server with an embedded database or SecureTransport Edge, use `<FILEDRIVEHOME>/bin/backup_db` to back up the database tables. The `backup_db` *`Backup_File`* command writes the backup data to the file named *Backup_file.sql*. You must enter the database password. You can use the `-skiplog` option to prevent the `TransactionStatus` or `TransactionData` tables from being included. To restore the data, use the following command:

- for servers with MySQL

```
<FILEDRIVEHOME>/mysql/bin/mysql --defaults-
file=<FILEDRIVEHOME>/conf/internaldb.conf -ptumbleweed st <
Backup_file.sql
```

- for servers with MariaDB

```
<FILEDRIVEHOME>/mariadb/bin/mariadb --defaults-
file=<FILEDRIVEHOME>/conf/internaldb.conf -ptumbleweed st <
Backup_file.sql
```

For additional information about export and importing server configuration parameters and files for backup and restore, see *Server configuration* on page 333.

Standard Cluster 4

This topic describes the concepts and procedures for deploying active-active and active-passive configurations of Axway SecureTransport using Standard Clustering. For information about Enterprise Clustering, see *Enterprise Cluster* on page 368.

The following topics describe the Standard Cluster (SC) model, configuring and setting up a cluster, and managing a Standard Cluster:

Note Explore our Modernized Standard Cluster option, currently in Beta and ready for test. Discover its key benefits compared to the legacy solution.
For comprehensive information on Modernized Standard Cluster, please refer to the dedicated guide covering installation, setup, and administration details. The document is available on the Axway Documentation portal only after login.

Standard Cluster model

Note Explore our Modernized Standard Cluster option, currently in Beta and ready for test. Discover its key benefits compared to the legacy solution.
For comprehensive information on Modernized Standard Cluster, please refer to the dedicated guide covering installation, setup, and administration details. The document is available on the Axway Documentation portal only after login.

As described in *Cluster models* on page 32, you can use a Standard Cluster (SC) to provide more capacity than a single SecureTransport Server or to provide a passive standby server to take over processing if an active server fails.

Standard Clustering uses an embedded database in each node. This minimizes external dependencies and overhead and reduces the cost of clustering. SecureTransport synchronizes most configuration changes on all nodes in the cluster.

You can configure a Standard Cluster as an active/active or active/passive (1:1) cluster. You can deploy a maximum of three servers (nodes) in an active/active Standard Cluster. An active/passive Standard Cluster has one active server and one passive standby server.

One node of a Standard Cluster is distinguished as the *primary server*. The passive node of an active/passive cluster and the one to two other nodes of an active/active cluster are *secondary* servers. The Administration Tool login screen and banner indicate **Primary Server** or **Secondary Server**.

The following topics describe the active/active and active/passive clustering and scheduled tasks, the consolidated log data representation, and the services used for cluster management:

- *Active/active and active/passive clustering* on page 352 - Describes active/active and active/passive clustering.
- *Scheduled tasks* on page 356 - Describes active/active and active/passive clustering scheduled tasks.
- *Consolidated log data representation* on page 356 - Describes the consolidated log data representation.
- *Services used for cluster management* on page 356 - Describes the services used for cluster management.

Active/active and active/passive clustering

SecureTransport supports two types of clustering: active/active and active/passive.

The following topics describe the active/active and active/passive clusters and processes:

- *Active/active clusters* on page 352
- *Active/passive clusters* on page 353
- *Active/passive deployment* on page 354
- *Primary server processes* on page 354
- *Failover* on page 355
- *Synchronization* on page 355

Active/active clusters

In an active/active cluster, SecureTransport balances the server-originating load between the primary and the secondary servers. All servers in the cluster are active and provide processing capacity. SecureTransport automatically replicates all event information collected by the primary server to the secondary servers. If the primary server fails, the cluster automatically switches control to the secondary server. An active/active cluster requires a third-party load-balancer. To prevent performance degradation if the primary server fails in an active/active cluster, the servers should have identical hardware.

The main advantages of an active/active cluster are:

- SecureTransport automatically balances the load between the different servers in the cluster.
- The secondary servers are active. This means that they can assume the load from a failed secondary server or take over from a failed primary server almost immediately.

However, to prevent performance problems, you must carefully monitor the load on an active/active cluster. An active/active cluster can suffer performance degradation when one server fails unless the remaining servers can handle the total workload for the cluster.

For more information, see *Set up an active/active cluster* on page 358.

Active/passive clusters

An active/passive cluster consists of one active server and one passive standby server with a third-party load-balancer to determine when failover is required for passive legacy and to send the users to the correct node. For passive, the SecureTransport will failover on its own. To prevent performance degradation if the primary server fails in an active/passive cluster, the servers should have identical hardware.

In an active/passive cluster, the primary server handles the event queue and processes events. While the primary server is active, the standby server remains in a passive state and does not process events. Depending on the server mode (set in the `Cluster.mode` server configuration parameter), an active/passive cluster has different features and advantages.

When the cluster mode is `passive`, the Transaction Manager runs on the secondary standby server and the event data is synchronized from the primary server. The primary server does not dispatch events to the Transaction Manager on the standby server and the standby server does not process events. The Sentinel link data is also synchronized.

The advantages of an active/passive cluster with `passive` cluster mode are:

- The cluster includes a fully redundant secondary standby server to handle the cluster workload if the primary server fails.
- Failover from the primary server to the secondary standby server is automatic.
- On failover, the cluster reports the states of file transfers to Sentinel consistently so that Sentinel can link them, so this mode is required for an active/passive cluster that works with Sentinel.

With `passive_legacy` cluster mode, the Transaction Manager listeners start on the standby server. The event data is not synchronized from the primary server, but the Transaction Manager on the standby server might accept connections and process events. You must stop the Transaction Manager on the standby server or configure your load balancer so that it does not direct traffic to it when the primary server is up and running. The Sentinel link data is not synchronized. You must disable the Folder Monitor and scheduler on the standby server. If the primary server fails, you must perform a manual failover to make the standby server active.

The advantages an active/passive cluster with `passive_legacy` cluster mode are:

- The cluster includes a fully redundant secondary standby server to handle the cluster workload if the primary server fails.
- Because events and Sentinel link data are not sent to a standby server with `passive_legacy` cluster mode, it can be in a different location for a disaster recovery (DR) if the network between the sites has required bandwidth and latency. For information on a more general DR solution using Enterprise Clustering, see *Passive disaster recovery* on page 371.

For more information, see *Set up an active/passive cluster* on page 360.

Active/passive deployment

The following diagram shows a simple active/passive deployment with `passive` cluster mode and one SecureTransport Edge server. On failover, the load balancer in the secure network redirects the connections from the internal clients and servers to SecureTransport Server 2 and the SecureTransport Edge directs the connection from the partner clients and servers to SecureTransport Server 2. For a description of the connections, see *Streaming deployment* on page 229.

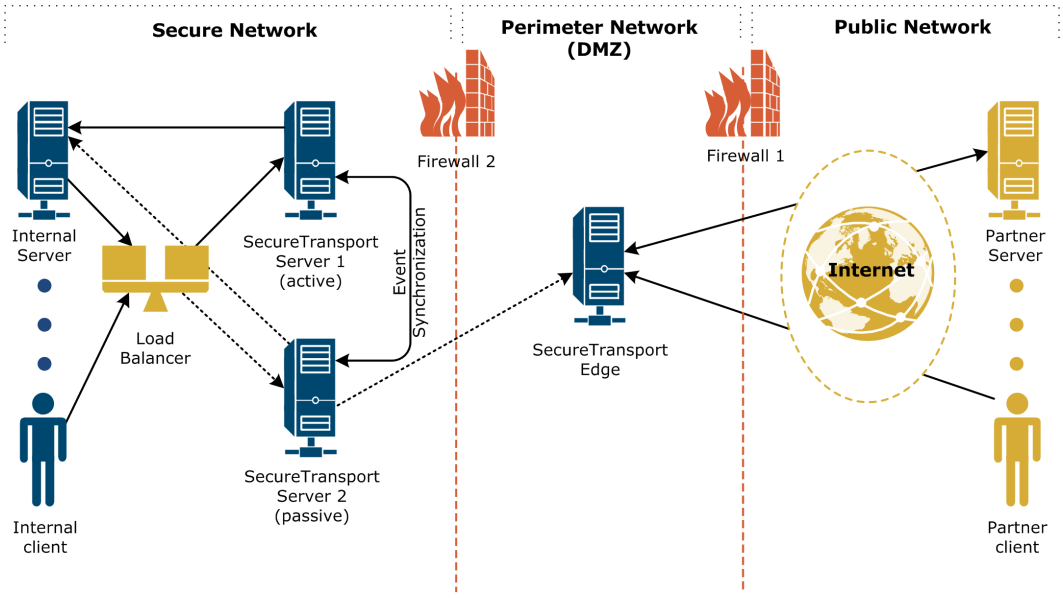


Figure 2. Active/passive deployment

The diagram does not show the shared file system where the user home directories are located.

The diagram does not show any unproxied connections or connections through HTTP proxy servers from the primary active SecureTransport Server or the secondary passive SecureTransport Server to the partner servers. For an illustrations of such a connection and more detail about connections in a streaming deployment, see *Streaming deployment* on page 229.

To handle your file transfer load or to provide redundancy in case of failure, you can deploy more than one SecureTransport Edge servers in the DMZ. Because SecureTransport Edge servers do not create and handle events, they are not clustered, but they can be configured to synchronize configuration changes. For more information, see *SecureTransport Edge synchronization* on page 407. For a deployment diagram showing multiple SecureTransport Edge server, see *Enterprise Cluster deployment* on page 369.

Primary server processes

The primary server hosts the following:

- The Folder Monitor service
- A consolidated transfer log for the cluster, displayed on the Administration Tool *File Tracking* page
- The scheduler that initiates scheduled transfers and maintenance applications

  The scheduler submits items to the internal event queue. The event queue distributes them among servers in the cluster.

Failover

When a server in an active/active cluster fails, two events occur. First, any internal load balancer for the cluster detects the server outage and fails over incoming requests to another server and all protocols servers on SecureTransport Edge servers detect the server outage and begin sending incoming requests to the TM Servers on the other SecureTransport Servers. In this case, the original client sessions are terminated and the clients must reestablish the sessions. Second, items in the event queue are reassigned from the failed server to other servers in the cluster.

When the primary server fails, one of the secondary servers is promoted to primary and the cluster continues processing messages. The secondary server promoted to primary status is determined by the order in which the server names are listed in the `<FILEDRIVEHOME>/lib/admin/config/servers` configuration file. The server listed just below the primary server in the file is considered the new primary server. When the cluster is synchronized, the state of the servers listed in the file are examined. If the first server in the servers list is online, it is reassigned as the new primary.

You can specify a timeout value that controls how long a secondary server waits before becoming the primary. For details, see *Specify the cluster connection timeout* on page 359.

When a secondary server is promoted to primary, it must run the scheduler and the Folder Monitor service. To enable this, the scheduler configuration on the primary server is replicated on all the other computers in the cluster.

When the active (primary) server fails in an active/passive cluster with cluster mode `passive`, the scheduler and the Folder Monitor service are already running on the passive standby server and failover is automatic. As with an active/active cluster, when the original active node recovers and the cluster is synchronized, the servers return to their original roles.

When the active (primary) server fails in an active/passive cluster with cluster mode `passive_legacy`, you must fail over the cluster to the standby server manually. All queued events and Sentinel link data are lost. For details, see *Manual failover* on page 364.

Synchronization

You can configure the cluster on any server. Most configuration changes are dynamically synchronized with the other servers immediately.

You must use the Administration Tool on the primary server to perform manual synchronization after you perform certain actions. Manual synchronization replaces the cluster configuration on the secondary servers with the configuration from the primary server. Data replicated during

synchronization includes database tables, cluster configuration data, and cluster management data. The Administration Tool server also performs manual synchronization each time it starts on the primary server. For more details, see *Standard Cluster synchronization* on page 364.

Scheduled tasks

In an active/active cluster, the SecureTransport scheduler manages all scheduled tasks centrally from the primary server in the cluster. When schedules trigger events for scheduled tasks, one consolidated queue for all events is maintained across the cluster. This queue is shared and replicated across all the servers in the cluster so that they share the load by taking events one item at a time from the queue and performing the actual transfers or other tasks. If the primary server fails, the scheduler start on the server that becomes the new primary server.

In an active/passive cluster, the secondary standby server does not process events. If the server mode is `passive`, the standby server starts the scheduler and starts processing events when the primary server fails.

Consolidated log data representation

The transfer log allows file tracking information to be monitored across the entire cluster. To this end, the Administration Tool on the primary server in the cluster provides a consolidated view of the transfer data stored in the server logs.

If the primary server goes down, a secondary server is promoted to primary and starts maintaining the transfer log information. If the former primary server comes back and resumes its primary role, the transfer log information from the temporary primary server is not migrated.

Services used for cluster management

SecureTransport implements the following services to manage and synchronize the cluster and the computers deployed in it.

The following topics describe the services used for cluster management:

- *Persistent Event Queue service* on page 356
- *Account Manager service* on page 357
- *Transfer Status Manager service* on page 357

Persistent Event Queue service

The Persistent Event Queue service is used to:

- Store certain events and replicates them to all cluster servers
- Synchronize the persistent events state to all cluster servers

- Perform recovery operations when a computer from the cluster fails during the processing of an event
- Spread the execution of following operations to all cluster servers:
    - Add a new event to the persistent queue
    - Delete an event from the persistent queue
    - Mark an event as active
    - Repair an event of a failed computer

Note The Event Queue service dispatches all events related to one remote PeSIT server to the same SecureTransport server in the cluster.

Account Manager service

The Account Manager distributes to all cluster servers dynamic information for the following events:

- Change of a user password.
- User login.

Transfer Status Manager service

The Transfer Status Manager service consolidates the transfer log on the primary server. All transfer log entries are stored in the database on the primary server.

Cluster configuration and setup

Much of SecureTransport configuration is stored in the database and synchronized dynamically to all servers in a cluster. Some configuration is stored in files that require manual synchronization. For more information, see the following topics:

- *Requirements for synchronization* on page 366
- *Server configuration* on page 333
- *Synchronize the cluster from the primary server* on page 366

The following topics provide additional configuration and setup information:

- *Set up an active/active cluster* on page 358 - Provides the how-to instructions for setting up an active/active cluster.
- *Specify the cluster connection timeout* on page 359 - Provides the how-to instructions for specifying the cluster connection timeout parameter.
- *Configure servers in a cluster to trust a certificate* on page 360 - Provides the how-to instructions for configuring the servers in a cluster to trust a certificate.
- *Set up an active/passive cluster* on page 360 - Provides the how-to instructions for setting up an active/passive cluster.

Set up an active/active cluster

You can set up a cluster of SecureTransport Server computers. When you set up a cluster, a shared secret file, (called `taeh` file) is used by each of the servers in the cluster for authentication purposes across servers and for encryption. The `taeh` file contains randomly-generated data that secures the SecureTransport cookies exchanged during server administration. SecureTransport generates the shared `taeh` file as part of the installation process. Refer to the *SecureTransport Installation Guide* for more information.

Note When you install SecureTransport on your secondary computers, you have an opportunity to import the `taeh` file from the primary server. You can only import the `taeh` file on a secondary computer during the installation process.

To set up an active/passive cluster, see *Manage an active/passive cluster* on page 363.

1. Make sure that the hosts file on each SecureTransport Server or SecureTransport Edge host operating system contains the hostnames and IP addresses of all servers with which it communicates: SecureTransport Server, SecureTransport Edge, and internal servers integrated with SecureTransport like LDAP, external database, Sentinel server, SSO server, ICAP server, etc.
2. Select the computer that is to serve as the primary server.
3. Make sure that the `taeh` file of the primary server is installed on all secondary computers.
4. Set up the secondary servers as independent installations. Add licenses for all servers. For instructions, refer to the *SecureTransport Installation Guide* or see *Server licenses* on page 178.
5. Generate an internal CA on each server. For instructions, refer to the *SecureTransport Getting Started Guide* or see *Manage the internal CA* on page 57.
6. Exchange CA certificates between all servers in the cluster. For details, refer to the procedures for exporting and importing SecureTransport Server CA certificates in the *SecureTransport Getting Started Guide* or in *Manage local certificates and certificate signing requests* on page 48.
7. Make sure that Transaction Manager (TM) servers are stopped on all computers.
8. On the primary and all secondary servers, list the servers in the `<FILEDRIVEHOME>/lib/admin/config/servers` configuration file. List the primary server first and continue with the secondary servers in the order you want them promoted to primary server in the event of failover.

   Edit the file and add a line of following form for each server in the cluster:

   ```
   host.example.com https://host.example.com:444
   ```

   where:

   - `host.example.com` is the FQDN or IP address of the computer
   - `https://host.example.com:444` is the URL of the Administration Tool on that computer

   and the two fields are separated by a tab character.

Note The `<FILEDRIVEHOME>/lib/admin/config/servers` file must be the same on all computer in your cluster. You can create it on one server and copy it to the others.

9. On the primary server, generate a local certificate for encrypting cluster communications. For instructions, see *Generate a self-issued server certificate* on page 49.
10. On the primary server, export the certificate in `.p12` format protected with a password. For instructions, see *Export a local certificate* on page 52.
11. Import the certificate on each secondary server. For instructions, see *Import a local certificate* on page 53.
12. On the primary and all secondary servers, configure encryption for cluster communications. On the *Server Configuration* page, change the value of the `Cluster.Crypto.Alias` server configuration parameter to the alias of the certificate.
13. On the primary server and all secondary servers, activate the cluster. On the *Server Configuration* page, change the value of the `Cluster.mode` parameter to `active`.
14. Start the TM server on the primary server and wait until it promotes itself as a primary server. Start the TM server on all other servers in the cluster. For instructions, see *Manage server operations* on page 1.
15. Synchronize the secondary servers manually from the Administration Tool of the primary server. For instructions, see *Requirements for synchronization* on page 366.

During cluster operation, most configuration changes are synchronized dynamically. For more information, see *Synchronization* on page 355.

Configuration optimizations in case of increased transfers load

In case of increased transfer payload, you can configure the maximum number of threads for `GeneralMessageProcessing` in the SecureTransport Server Configuration by increasing the value of the following parameter:

`Cluster.ThreadPools.ThreadPool.GeneralMessageProcessing.maxThreads`

By default this value is set to `4`. For optimal performance, you can increase this value up to `100` or even more.

You have to apply this configuration across all nodes.

Specify the cluster connection timeout

You can specify a timeout value that determines how long a secondary server waits before declaring itself the primary server. This is important in cases where the primary server is heavily loaded and takes an extended period of time to respond to a secondary computer.

- On the *Server Configuration* page of the primary server, change the value of the `Cluster.connectionTimeout` parameter to the value required in milliseconds. The default value is 60000.

Configure servers in a cluster to trust a certificate

In a cluster environment, the CA that issued the certificate must be trusted by all servers in the cluster. Refer to the procedure below. The self-signed certificates used to sign the server certificates during the installation of the servers also need to be configured across the servers. If the signing certificates for the SecureTransport servers are issued by different CAs, configure each server with the root CA certificate of that CA.

1. On primary server, export the CA certificate to a local file using the *Trusted CAs* page.
2. Copy the certificate file with the CA certificate from the primary server to the secondary server and import the certificate using the *Trusted CAs* page.
3. On secondary server, export the CA certificate to a local file using the *Trusted CAs* page.
4. Copy the certificate file with the CA certificate from the secondary server to the primary server and import the certificate using the *Trusted CAs* page.
5. Bounce both servers. For instructions, see *Limit FTP login failures* on page 202.

Set up an active/passive cluster

When you change the cluster settings to be active/passive, the secondary standby server stops processing events. For details, see *Active/active clusters* on page 352.

1. Set up a two-server cluster using the instructions in *Set up an active/active cluster* on page 358.
2. Stop the TM server on both servers in the cluster.
3. On the *Server Configuration* page of each server, change the value of the `Cluster.mode` parameter to `passive` or `passive_legacy`.
4. On the primary server, create a file named `<FILEDRIVEHOME>/var/tmp/sentinel_primary`. This file is not used for integration with Axway Sentinel. It is required whether or not Sentinel is used.

   To create the file, you can use the `touch` command in UNIX or create an empty file with no file extension in Windows. The file must have 0 bytes.
5. If you set `Cluster.mode` to `passive_legacy`, on the standby server, set `FolderMonitor.enable` and `Scheduler.enable` to `false`.
6. Start the TM server on both servers in the cluster.

Manage a Standard Cluster

The dynamic functioning of the cluster framework is part of the TM server and therefore any change of the TM server state affects the dynamic state of the respective computer in the cluster.

The following topics describe how to manage active/active and active/passive clusters and describe how to perform Standard Cluster (SC) synchronization.

- *Manage an active/active cluster* on page 361 - Describes how to perform management tasks for an active/active cluster setup.
- *Manage an active/passive cluster* on page 363 - Describes how to perform management tasks for an active/passive cluster setup.
- *Standard Cluster synchronization* on page 364 - Describes how to perform Standard Cluster synchronization.

Manage an active/active cluster

The following procedures describe how to perform management tasks for an active/active cluster setup.

The following topic provide the how-to instructions for managing an active/active cluster:

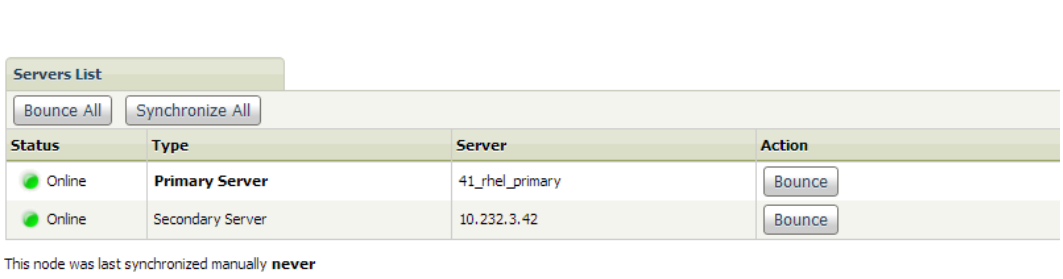
- *Monitor an active/active cluster* on page 361
- *Add a server to an active/active cluster* on page 362
- *Restore a server to an active/active cluster* on page 363
- *Remove a server from an active/active cluster* on page 363

Monitor an active/active cluster

Cluster status is displayed in the Administration Tool.

- Select **Operations > Cluster Management**.

  The *Cluster Management* page is displayed.



For each Server in the cluster, the page lists its status (online or offline), its type (primary or secondary), and its host name or IP address. Online status means the server is running and communicating with the cluster. Offline status means the server has been stopped, has failed, or cannot communicate with the cluster.

The **Bounce**, **Bounce All**, and **Synchronize All** buttons do not appear on secondary servers. To bounce a secondary server locally, see *Reload server configuration* on page 1.

On a secondary server, the time of the last manual synchronization is reported. The timestamp is also reported in the `cluster_last_sync_timestamp` file located in the `<FILEDRIVEHOME>/var/tmp/cluster_last_sync_timestamp` directory.

Cluster status is also stored in the `cluster_state` file. It contains information about which cluster node is the primary server, which are secondary servers, which servers are online, and which are offline. For example:

```
<ClusterGroup name="STCluster">
   <Member hostname="test01.your.cluster" state="online"
      status="primary"/>
   <Member hostname="test02.your.cluster" state="online"
      status="secondary"/>
   <Member hostname="test03.your.cluster" state="offline"
      status="secondary"/>
</ClusterGroup>
```

By default, the `cluster_state` file is located in `<FILEDRIVEHOME>/var/tmp/cluster_state`. You can change the location and file name by editing the following server configuration parameters:

- `Cluster.File.clusterStateFile.relative` – the base location (`fdhome` for `<FILEDRIVEHOME>`)
- `Cluster.File.clusterStateFile.path` – the path and file name relative to `Cluster.File.clusterStateFile.relative`

Add a server to an active/active cluster

1. Stop the TM server on all servers in the cluster.
2. Add the information about the new server into the `<FILEDRIVEHOME>/lib/admin/config/servers` file on the primary server.
3. Copy the `servers` file to the new computer.
4. Follow steps 11 through 15 from *Set up an active/active cluster* on page 358.

Note The new server must use the same secret file as the rest of the nodes in the cluster.

Restore a server to an active/active cluster

1. Start the Administration Tool and TM server on the restored server.
2. Perform a manual synchronization from the Administration Tool of the primary server. For instructions, see *Synchronize the cluster from the primary server* on page 366.

Note Synchronize the restored server as soon as possible. If the restored server becomes primary before it is synchronized, it might process messages based on outdated data it received before it failed.

Remove a server from an active/active cluster

When you remove a server from an active/active cluster, you can configure it as a stand-alone server.

1. On each of the nodes in t he cluster, stop the TM server.
2. On the node you are removing from the cluster:
   a. Open the `<FILEDRIVEHOME>/lib/admin/config/servers` file and remove the lines of information about the servers in the cluster.
   b. On the *Server Configuration* page, set the `Cluster.mode` parameter to `disabled`.
3. On each of the remaining nodes in the cluster, open the `<FILEDRIVEHOME>/lib/admin/config/servers` file and remove the line of information about the server you are removing from the cluster.
4. On each of the remaining nodes in the cluster, start the TM server.
5. On the primary server, perform a manual synchronization from the Administration Tool. For instructions, see *Synchronize the cluster from the primary server* on page 366.

Manage an active/passive cluster

The following procedures describe how to perform management tasks for an active/passive cluster setup.

Note A cluster with cluster mode `passive_legacy` does not monitor the state of each server in the cluster. However, if you attempt to synchronize all the servers in the cluster, and no errors occur, it indicates that both servers in the cluster are up.

Caution In an active/passive cluster, the secondary server cannot be bounced remotely.

The following topics provide how-to instructions for managing an active/passive cluster:

Restore a server to an active/passive cluster

1. Start the Administration Tool server and the TM server on the restored computer.
2. On the primary server, perform a manual synchronization from the Administration Tool. For instructions, see *Synchronize the cluster from the primary server* on page 366.

Note It is recommended that you synchronize the restored computer as soon as possible. If the restored computer becomes primary before it is synchronized, it might process messages based on outdated data it received before it failed.

Remove a server from an active/passive cluster

When you remove a server from an active/passive cluster, the result is two stand-alone servers.

1. Stop the TM server on both servers in the cluster.
2. On each servers, open the `<FILEDRIVEHOME>/lib/admin/config/servers` file and remove the lines of information about the servers in the cluster.
3. On each server, set `Cluster.mode` to `disabled`.
4. If the cluster mode was `passive_legacy`, on the previous standby server, set `FolderMonitor.enable` and `Scheduler.enable` to `true`.
5. Restart the TM server on both servers.

Manual failover

When the primary server fails in an active/passive cluster with cluster mode set to `passive_legacy`, you must perform a manual failover on the passive standby server:

1. Create a file named `<FILEDRIVEHOME>/var/tmp/sentinel_primary`.

   To create the file, you can use the `touch` command in UNIX or create an empty file with no file extension in Windows. The file must have 0 bytes.
2. On the *Server Configuration* page, set both `FolderMonitor.enable` and `Scheduler.enable` to `true`.
3. On the *Server Control* page, start the TM Server.

Standard Cluster synchronization

You can configure the cluster on any server. Most configuration changes are synchronized with the other servers immediately.

When needed, you must propagate configuration data from the primary server to secondary servers across a cluster through manual synchronization. Use the Administration Tool on the primary server to perform synchronization.

You must perform manual synchronization after you:

- Upgrade SecureTransport
- Restart the whole cluster
- Restore a failed primary server
- Restart the Administration Tool server on a secondary server, if you made changes using the Administration Tool on the primary server while it was down

Manual synchronization replaces the information on the secondary servers with the configuration from the primary server. The Administration Tool server also performs dynamic synchronization each time it starts on the primary server.

You cannot use manual synchronization to data move information from the primary server to selected secondary servers.

A full synchronization may take a long time to complete and during the synchronization process, all dynamic updates are stopped, so make sure that you perform the synchronization when your system is not under heavy load.

When you create an application or account, or change the keystore password on the primary server, like most configuration information, the new application, account settings, or keystore password is copied to the secondary servers using dynamic synchronization when you save the settings. However, the audit log entry for the application, keystore password, or account creation is not copied from the primary to the secondary servers.

The directories for accounts that are synchronized manually use the UID and GID specified within the account settings on the secondary servers when the parent directory of the account exists only on the primary server in the cluster.

Note When synchronizing an application, an error message is returned indicating that the application was not copied if the secondary servers do not already have the application type the application is based on. Clicking the **Synchronize** button copies the application types from the primary server to the secondary servers, along with the application instances created.

Servers that are online, but not fully synchronized, are updated with new account and application information without having to perform a manual synchronization. You must still perform a full synchronization to start the TM server on a server that was previously down, even though the account and application information might be current.

The following topics describe the synchronization requirements and what information is synchronized and provide how-to instructions for synchronization:

- *Requirements for synchronization* on page 366
- *What information is synchronized* on page 366
- *Synchronize the cluster from the primary server* on page 366

Requirements for synchronization

Dynamic and manual cluster synchronization requires the following:

- The SecureTransport administrator account names must be the same on all servers.
- When certificate authentication is enabled for the administrator accounts, Cluster.DynamicSync.adminName and Cluster.DynamicSync.keyAlias configuration options must be set on all nodes manually.
- The SecureTransport installation path must be the same on all servers.
- The `taeh` file must be the same on all servers.
- The certificate for encrypting cluster communications specified in the `Cluster.Crypto.Alias` server configuration parameter must be the same on all servers.
- The internal CAs on all nodes must be trusted by all other nodes. Optionally, you can import the same internal CA on all nodes.
- All the SecureTransport server certificates must be issued by a common CA.
- Each server must be hosted on a different computer or virtual machine.
- All the servers in an active/active cluster must be in the same LAN.
- The two servers in an active/passive cluster with cluster mode `passive_legacy` can be in different locations, but they must be in the same low latency network.
- All servers in a cluster must have their clocks set to the same time.
- The TM Server must be running on all servers.
- All database settings must be *identical* on all the servers.

What information is synchronized

The following types of information are synchronized:

- All configuration files listed in the `<FILEDRIVEHOME>/conf/sync.conf` file
- All database tables listed in the `<FILEDRIVEHOME>/conf/sync_tables.conf` file
- All server configuration parameters that are not local to the server

The files listed in `sync.excl` are not copied from the primary to the secondary server.

Synchronize the cluster from the primary server

Use the Administration Tool on the primary server to synchronize the SecureTransport servers in a Standard Cluster.

Note The upper right corner of the Administration Tool shows whether the server on which it is running is a primary or secondary server.

1. Select **Operations > Cluster Management**.

   The *Cluster Management* page is displayed.

2. Click **Synchronize All**.

   When synchronization completes, the page displays the status of the operation and links to the secondary server Administration Tools.

Enterprise Cluster 5

This topic describes the concepts and procedures for deploying active-active clusters and clusters with passive disaster recover (DR) of Axway SecureTransport using Enterprise Clustering. For information about Standard Clustering, see *Standard Cluster* on page 351.

The SecureTransport Enterprise Pack bundles a set of advanced features under a separate license. These features are SecureTransport Cloud Edition (login required), Zero Downtime Update (ZDU), and Enterprise Cluster. The enablement, access and/or use of the Zero Downtime Update functionality, is only authorized under an active subscription license to SecureTransport specifically licensing Customer to Enterprise Pack. *

Related topics

The following topics describe and provide the how-to instructions for managing an Enterprise Cluster (EC):

- *Enterprise Cluster model* on page 368 - Describes the Enterprise Cluster model.
- *Manage an Enterprise Cluster* on page 374 - Provides the how-to instructions for managing an Enterprise Cluster.

Enterprise Cluster model

As described in *Cluster models* on page 32, the managed file transfer workload of an enterprise can exceed the capacity of Standard Clustering. Also, an enterprise might prefer to use an external database to allow their DBAs additional tuning of the underlying database. In either of these cases, you can use Axway SecureTransport 5.5 with the Enterprise Cluster (EC) option to implement at a single site a cluster that provides the capacity to handle the file transfer workload required by your organization. With an Enterprise Cluster, an external database and a high-performance cache-management layer significantly improve efficiency and increase potential scale. This allows up to 20 nodes in an active/active cluster. The Enterprise Cluster option requires your organization to provide and maintain an external database.

An Enterprise Cluster is efficient and flexible. All tasks, including scheduled work, are distributed across the cluster. You can add and remove servers as needed up to the maximum allowed by your SecureTransport features license. You can also control how the workload is directed to the servers in the cluster, for example, based on task type.

* For clarification and the avoidance of any doubt, Customer is not entitled to enable, access, or use ZDU or any of the features in Enterprise Pack without such specific contractual entitlement and license to SecureTransport under a subscription contract, and any use of ZDU or Enterprise Pack features without such specific entitlements is beyond the scope of the license to Customer. Any enablement, access, and/or use of Enterprise Pack or ZDU use will require the purchase of subscription license by Customer at additional cost.

In an Enterprise Cluster, all servers are active, so there are no standby servers to replace an active server that fails. However, because the components that direct and schedule tasks are distributed across the clustered servers, the cluster implements failover by continuing to process its workload with reduced capacity when a server fails. For increased availability, you can remove another potential single point of failure by implementing a database cluster for the shared database.

Also, with the Enterprise Cluster option, you can implement an active/active cluster with a passive Disaster Recovery (DR) site. For this, you must provide a properly distributed and replicated database and file storage. To enable your implementation of passive DR, the primary production active/active cluster can notify an external mechanism to trigger failover automatically when the number of active nodes in the cluster falls below a threshold.

The following topics provide additional Enterprise Cluster information:

- *Enterprise Cluster deployment* on page 369 - Describes the Enterprise Cluster deployment.
- *Workload distribution* on page 371 - Describes the Enterprise Cluster workload distribution.
- *Passive disaster recovery* on page 371 - Describes the Enterprise Cluster passive disaster recovery.

Enterprise Cluster deployment

An Enterprise Cluster (EC) distributes the workload among a collection of networked SecureTransport Servers, all of which are active. All servers in an Enterprise Cluster must be on the same low latency network.

The following diagram illustrates a possible deployment architecture for an Enterprise Cluster that serves partners who access it using the Internet. The arrows show the direction of network connections for all the protocols. Data flows in both directions after the connection made.

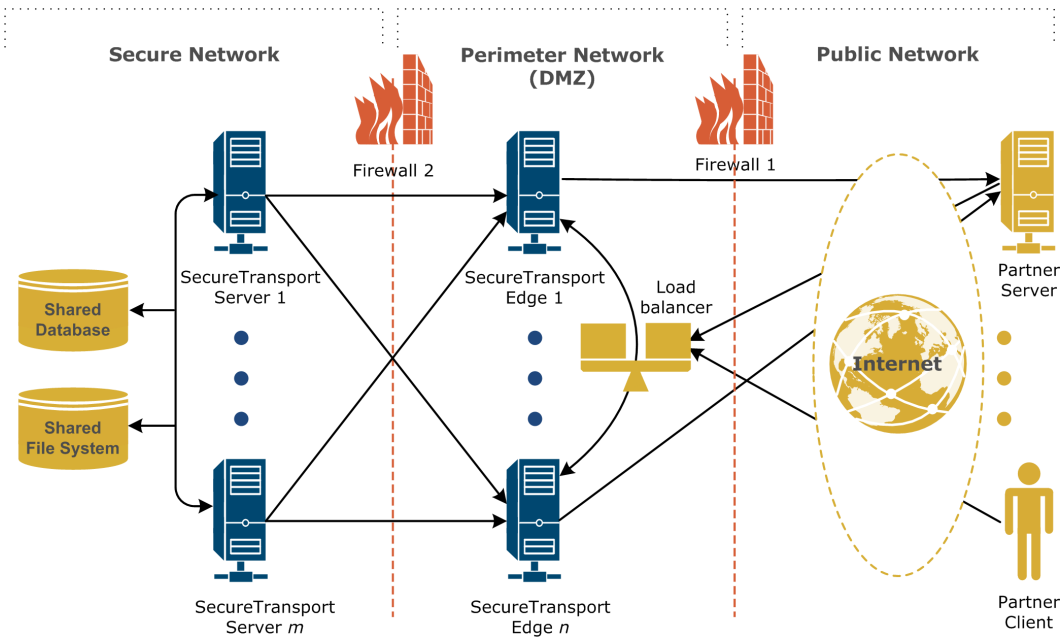


Figure 3. Enterprise Cluster

The diagram does not illustrate the event synchronization among the SecureTransport Servers or any unproxied connections or connections through HTTP proxy servers from the SecureTransport Servers to the partner servers. For an illustration of an unproxied connection and more detail about connections in a streaming deployment and configuration for internal clients and servers, see *Streaming deployment* on page 229.

Deploy the number of SecureTransport Servers and SecureTransport Edge servers required for your file transfer workload. The connections from the TM Servers on the SecureTransport Servers to the protocol and SOCKS5 proxy servers on the SecureTransport Edge servers is many-to-many, so you can design your deployment with consistent or specialized configuration. For more information, see *Communication across Transaction Manager, protocol, and proxy servers* on page 227. Because SecureTransport Edge servers do not create and handle events, they are not clustered, but they can be configured to synchronize configuration changes. For more information, see *SecureTransport Edge synchronization* on page 407.

Components

This example deployment architecture includes the following components:

- **SecureTransport Servers** – An Enterprise Cluster has two or more SecureTransport Servers. Each SecureTransport Server has an installation directory on its local file system. All the installation directories must have the same path. You can also deploy SecureTransport with the EC option as a single Server when you require an external database.
- **Shared external database** – The SecureTransport Servers share an external database that they use to implement the cluster. The shared database stores all the shared (cluster-wide) configuration data and much of the local (individual) configuration data for the servers.
- **Shared file system** – The SecureTransport Servers share a file system on external storage for working directories and files, for example, using a shared disk file system on a storage area network (SAN) or using network-attached storage (NAS). Because all servers in a cluster share many configuration definitions that include references to directories, such as accounts, the shared file system hosts those directories. SecureTransport installation on a private SAN logical unit number (LUN) is also a supported configuration.
- **SecureTransport Edge servers** – If an Enterprise Cluster provides service to systems in a public or external unsecured network, it usually includes SecureTransport Edge servers. A cluster deployment usually includes one SecureTransport Edge for each SecureTransport Server, but this can vary depending on the characteristics of the workload. The SecureTransport Edge servers are not clustered, but their configuration can be synchronized.
- **Firewalls** – The firewalls implement the perimeter network required to serve client systems in a public or external network.
- **Load balancer** – In the Enterprise Cluster deployment diagram, the load balancer implements workload distribution by distributing the incoming requests from the external clients and servers among the SecureTransport Edge gateways. The specific load balancing method depends on the characteristics of your workload and cluster deployment.

Workload distribution

Event distribution in a SecureTransport Enterprise Cluster (EC) is handled by a distributed event processor and a distributed scheduler. As long as one SecureTransport Server is running, incoming and scheduled tasks are assigned.

You can configure the event processor to distribute events based on their attributes, however the event processor assigns all events related to one remote PeSIT server to the same SecureTransport server. Event assignment options are:

- Events can be assigned to a server with required functionality.
- Events associated with an particular account can be assigned to the same server. This can improve the performance of the distributed object cache by caching the object that represents the account and related object where they are used.

Passive disaster recovery

If Axway SecureTransport provides an essential function in your organization, it is likely to require a business continuity plan. To assure business continuity, you can deploy your Axway SecureTransport 5.5 Enterprise Cluster with passive disaster recovery (DR). The following diagram illustrates a recommended deployment architecture for an Enterprise Cluster with passive disaster recovery. The arrows show the direction of network connections for all the protocols. Data flows in both directions after the connection made.

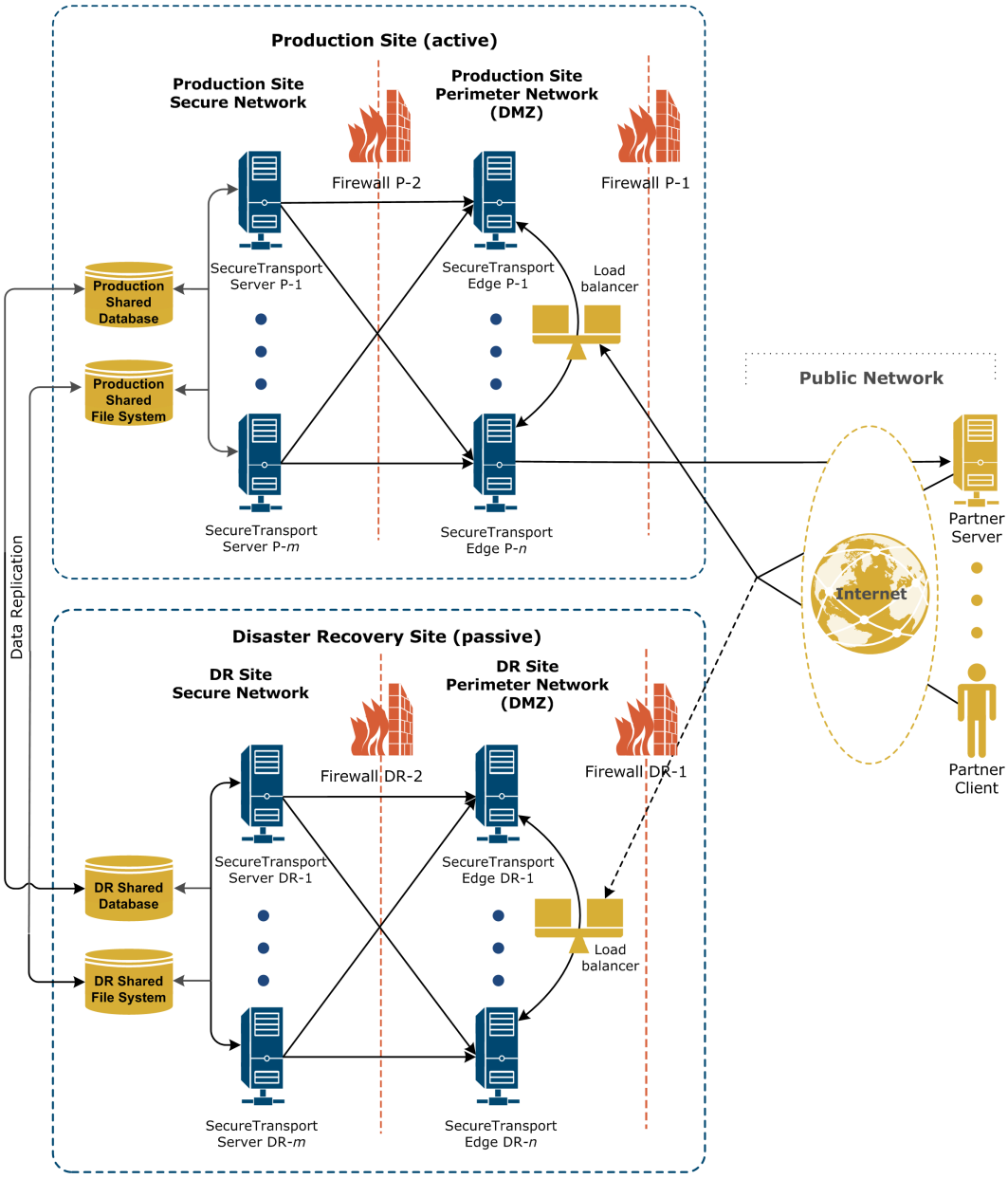


Figure 4. Enterprise Cluster with passive disaster recovery

The diagram does not illustrate the event synchronization among the SecureTransport Servers in one site or any unproxied connections or connections through HTTP proxy servers from the SecureTransport Servers to the partner servers. For simplicity, the diagram shows only one connection from a SOCKS5 proxy on a SecureTransport Edge to a partner server. The SOCKS5 proxies on all the SecureTransport Edge servers can connect to the partner servers through the firewalls as needed. For an illustration of an unproxied connection and more detail about connections in a streaming deployment and configuration for internal clients and servers, see *Streaming deployment* on page 229.

This deployment architecture uses a redundant Enterprise Cluster and SecureTransport Edge servers at a separate site to provide passive DR. The DR cluster must provide the functionality of the production cluster with the same number or fewer servers than the production cluster. The SecureTransport Edge servers must also provide the functionality of the production site with the same number or fewer servers than the production site. All the servers must run on the same type of operating systems with the same installation directory, secret file, and configuration as the production servers. The SecureTransport Servers in the DR cluster must also have the same database type and configuration as the production SecureTransport Servers and the shared file system must be mounted under the same path on all SecureTransport Servers.

Each server in the DR site is associated with a server in the production site. So if there are fewer servers in the DR site, some servers in the production site are not associated with a server in the DR site. If there are fewer servers in the DR cluster, the configuration of the production site servers that are not associated with a server in the DR site must be maintained on the DR site so that the configuration can be replicated to the production site failback. The DR site *Cluster Management* page lists any production site servers that are not associated with a server in the DR site and shows them as offline.

The DR site is a passive standby. The servers in the DR site do not run when the production site is operational, but the configuration of the DR cluster and the DR SecureTransport Edge servers must be consistent with the configuration of the production cluster and SecureTransport Edge servers so that the DR site is ready to replace the production site. However, the IP addresses of the servers in the production site and the servers in the DR site can be different.

Each cluster has its own shared database and shared file system. So that the DR cluster can replace the production cluster when it is needed, you must implement database synchronization between the production cluster database and the DR cluster database so that the configuration and operational data is consistent. You must also implement data synchronization between the production cluster shared file system and the DR cluster shared file system so that the account home folders and other folders are consistent. Do not synchronize the file systems that host the operating systems and the SecureTransport application files. You keep these consistent by applying the same patches, services packs, and configuration file changes to all servers.

The system configuration and account information of the SecureTransport Edge servers in the production site and the SecureTransport Edge servers in the DR site must be consistent. If there is just one SecureTransport Edge server or if the SecureTransport Edge servers in the sites are synchronized, you can export the system configuration and administrator account information from one of the SecureTransport Edge servers in the active site, import it into one of the SecureTransport Edge servers in the inactive site, and synchronize the SecureTransport Edge servers in the inactive site. You need to do this whenever you change the system configuration or accounts on a SecureTransport Edge server in the active site. To import system configuration and account information into a SecureTransport Edge server or to manually synchronize SecureTransport Edge servers, you must start the database and the Administration Tool server. After the SecureTransport Edge servers are updated, you must stop all services on them.

Note If SecureTransport is deployed in a secure perimeter network (DMZ) configuration, the DMZ zones configuration cannot be modified. As a result, streaming is not operational in the DR site (as the nodes have different IP addresses) and all server-initiated transfers set to establish a connection through a DMZ zone will use the IP addresses configured for the production environment.

If you maintain consistency between the production and the DR sites, you can choose to use an external DR solution.

Manage an Enterprise Cluster

As described in *Cluster models* on page 32, an Enterprise Cluster is a SecureTransport deployment that uses a collection of servers at a single site to provide the capacity to handle a very large file transfer workload. All SecureTransport Server systems in the cluster share an external database and a file system. This topic describes how to set up and maintain an Enterprise Cluster (EC). These operations are available only on the SecureTransport Server.

The following topics provide the prerequisites and how-to instructions for managing an Enterprise Cluster:

- *Enterprise Cluster prerequisites* on page 374 - Provides the Enterprise Cluster prerequisites.
- *Set up a cluster* on page 375 - Provides the how-to instructions for setting up an Enterprise Cluster.
- *Add a server to a cluster* on page 380 - Provides the how-to instructions for adding a server to an Enterprise Cluster.
- *Remove a server from a cluster* on page 382 - Provides the how-to instructions for removing a server from an Enterprise Cluster.
- *View cluster status* on page 382 - Provide the how-to instructions for view the status of an Enterprise Cluster.
- *Notification of cluster status* on page 383 - Provides the how-to instructions for setting up email notification of cluster status.
- *Set up a disaster recovery cluster* on page 383 - Provides the how-to instructions for setting up a disaster recovery cluster.
- *Maintain a disaster recovery cluster* on page 385 - Describes maintaining a disaster recovery cluster.
- *Disaster recovery failover and fallback* on page 385 - Describes disaster recovery site failover and fallback.
- *Direct cluster workload* on page 386 - Describes balancing the direct cluster workload and provides how-to instructions for balancing the workload.

Enterprise Cluster prerequisites

Before you deploy an Enterprise Cluster, you must have:

- A features license that permits the number of clustered SecureTransport Servers (nodes) in your cluster.
  The enablement, access, and/or use of Enterprise Pack, including Zero Downtime Update (ZDU), is only authorized under an active subscription license to SecureTransport that specifically

licenses the customer to use the Enterprise Pack.*

- A license for all the SecureTransport Edge servers in your cluster.
- An external Oracle, PostgreSQL, or Microsoft SQL Server database server or cluster that satisfies the requirements described in the *SecureTransport Installation Guide*.
- A file system that all servers in the cluster can access (for example, using network-attached storage).
- The working network, including the firewall and load balancer systems required by your cluster deployment.
- The time settings (clocks) on all computers in the network synchronized.
- If a secure connection to the database is used, the cluster server installer will need to have the database certificate or access to the Java key store containing the database certificate.

Set up a cluster

This is an overview of the steps required to create an Enterprise Cluster (EC). For the procedures to perform the server installation and initial configuration, refer to the *SecureTransport Installation Guide*.

Note If you use the **ClusterAuto-Register IP/FQDN** option during the installation of the SecureTransport Servers, the nodes of the cluster will be registered automatically. In this case, after completing step 4, you can directly proceed with step 9; you do not have to follow steps 5 to 8.

1. Implement and test the network and computers for the cluster, including the shared database and shared file system.
2. Install the first SecureTransport Server.

   When you install the first SecureTransport Server, the installer creates the schema for the cluster in the shared database.
3. Install the SecureTransport licenses and perform the initial configuration for the first SecureTransport Server.

* For clarification and the avoidance of any doubt, Customer is not entitled to enable, access, or use ZDU or any of the features in Enterprise Pack without such specific contractual entitlement and license to SecureTransport under a subscription contract, and any use of ZDU or Enterprise Pack features without such specific entitlements is beyond the scope of the license to Customer. Any enablement, access, and/or use of Enterprise Pack or ZDU use will require the purchase of subscription license by Customer at additional cost.

4. Install the other SecureTransport Servers. Specify the same installation directory, specify usage of the existing database schema, and import the `taeh` file from the first server.

   Note The Administration Tool service on a newly installed server starts after the installation; however, it is not operational.

5. Stop the Administration Tool service on each newly installed SecureTransport Server.
6. Stop all the protocol servers and services on all nodes except on the first SecureTransport Server. Make sure that the Administration Tool service is running only on that SecureTransport Server.
7. Log on to the Administration Tool on the first installed server as the `admin` user, and add to the cluster each of the cluster nodes, including the one you are logged on to. For details, see *Add a server to a cluster on page 380*.

   Note Do not restart any other SecureTransport Server until it is added to the cluster.

   Note Verify that all SecureTransport plug-ins, if using any, are deployed on each SecureTransport Server before adding it to the cluster.

8. Start the Administration Tool service on each newly installed SecureTransport Server.
9. Install the SecureTransport licenses on each newly installed SecureTransport Server.

   Do not perform the other steps of the initial configuration because the configuration is copied to the other servers when they are added to the cluster.

10. Restart all SecureTransport Servers.

The Enterprise Cluster is now operational with its basic initial configuration.

TLS encrypted communication across server nodes

Communication across SecureTransport Server nodes in an Enterprise Cluster can be encrypted using TLS. The certificate with the `admind` alias is used for the encryption. The certificate must have the Client Authentication extended key usage.

- On a fresh installation of the November 2020 SecureTransport 5.5 build or later, TLS encrypted communication across SecureTransport Server nodes is applied by default.
- If SecureTransport is upgraded to the November 2020 SecureTransport 5.5 build or later from any previous version (for example, 5.4 latest patch or 5.5 GA), the server nodes communication is not encrypted by default.

You can modify this behavior using the `Cluster.enable.SSL` server configuration option.

You can then proceed with the rest of the configuration setup, for example, perform the initial configuration for the SecureTransport Edge Servers.

Tip You can perform most configuration tasks once on any SecureTransport Server. The configuration is saved in the database and shared across all other SecureTransport Server nodes in the cluster. For more details on server configuration specifics, see *Server configuration on page 333*.

Note If the certificate used for SSL connection is not valid (has expired or is not chained to a trusted root), the cluster does not form and the Administration Tool service is started only on the node that was started first. After a valid certificate is generated or imported, all services must be manually restarted on all nodes.

Coherence unicast discovery

SecureTransport ships with Coherence cluster implementation, configured for unicast discovery. The well-known addresses list is populated at runtime, based on the environment-specific network configuration. If for some reason the list cannot be populated, for example after a prolonged database connectivity outage or invalid DNS resolution, the Coherence cluster may not form, the Admin service may be unable to start, and the following error may be logged in the *FILEDRIVEHOME/tomcat/admin/log/catalina.out* file:

```
com.tangosol.net.RequestTimeoutException: Timeout during service
start
```

As a preventative measure, it is recommended to reconfigure Coherence to use a hardcoded list of well-known addresses in the *tangosol-coherence-override.xml* file, as described in KB 178019.

Performance tuning for increased transfer load

This topic provides guidance for optimizing SecureTransport performance in cases of increased transfer load.

TM thread pool size

The following Transaction Manager thread pool server configuration options have a default maximum threads value of `768`, which can be adjusted if necessary:

- `TransactionManager.ThreadPools.ThreadPool.EventMonitor.maxThreads`
- `TransactionManager.ThreadPools.ThreadPool.ServerTransfer.maxThreads`
- `EventQueue.ThreadPools.ThreadPool.maxThreads`
- `EventQueue.ThreadPools.AdvancedRouting.maxThreads`

Database connection pool size

To tune the size of database connection pools, see *Connection pools for SecureTransport components* on page 110.

Memory allocation

The protocol daemon services and the TM service run as separate Java processes. The amount of RAM memory allocated to each service is set in the *STStartScriptsConfig* file, see *Advanced service configuration and memory allocation on page 287*.

Advanced cache tuning

When enough RAM is allocated to the services based on the expected load, you can perform advanced cache tuning by adjusting the cache size for the corresponding objects in the *hibernate-cache-config.xml* and *coherence-cache-config-tm.xml* files located in the `FILEDRIVEHOME/conf` directory.

Hibernate-cache-config

Below are the default values of the *hibernate-cache-config.xml* file.

```
local-scheme:
accounts-local-tm: "{size-limit 10000}"
accounts-properties-local: "{size-limit 4234}"

distributed-scheme:
accounts-back-tm: "10000"

cache-mapping:
SiteCache: "1000"
CustomAttributesCache: "100000"
DataTransformationCache: "800"
SubscriptionCache: "10000"
TransferConfigurationCache: "800"
ApplicationCache: "150"
ConfigurationCache: "45000"
ConfigurationProfileCache: "100"
RouteCache: "10000"
AddressBookSourceCache: "5000"
org.hibernate.cache.*: "20000"
*: "1000"
```

The *accounts-properties-overflow* scheme's `expiry-delay` is set to 15 minutes.

The *accounts-properties-external* paged cache is stored in `FILEDRIVEHOME/var/tmp/tm/coherence-offheap-cache-disk` with a `page-duration` of 5 minutes and a `page-limit` of 80.

Coherence-cache-config-tm

Below are the default values of the *coherence-cache-config-tm.xml* file.

```
cache-mapping:
SiteCache: "10000"
IdfCache: "2500"
AccountLocalKeyCache: "5000"
AccountLocalPGPKeyCache: "2000"
CustomAttributesCache: "250000"
DataTransformationCache": "20000"
SubscriptionCache: "20000"
TransferConfigurationCache: "20000"
LocalKeyCache: "1000"
LoginCertificateCache: "5000"
CertificateReferenceCache: "10000"
TrustedCACertificateCache: "2000"
PartnerCertificateCache: "2000"
LocalPGPKeyCache: "4500"
PartnerPGPKeyCache: "4500"
ApplicationCache: "2000"
BusinessUnitCache: "2000"
RouteCache: "20000"
AddressBookSourceCache: "5000"
LoginRestrictionCache: "2000"
```

Invocation-scheme configuration

The following suggested configuration allows the SecureTransport administrator to specify the number of daemon threads used by the invocation service in the invocation-scheme configuration in `conf/hibernate-cache-config.xml`:

```
<!-- The DefaultInvocationService is used by the
com.tumbleweed.st.server.api.cluster.InvocationManager -->
<invocation-scheme>
<scheme-name>invocation-service</scheme-name>
<service-name>DefaultInvocationService</service-name>
<thread-count>0</thread-count>
<autostart>true</autostart>
</invocation-scheme>
```

Note `<thread-count>` is an optional parameter which specifies the number of daemon threads used by the invocation service. When set to zero, all relevant tasks are performed on the service thread. Accepted values include '`0`' and positive integers. The default value is the value specified in the `tangosol-coherence.xml` descriptor.

Add a server to a cluster

If a SecureTransport Server uses the cluster shared database schema and shared file system, you can add it to the cluster using the Administration Tool. To connect a SecureTransport Server to an existing database schema, see *Migrate from embedded database to external Oracle database on page 96* and *Change Oracle database configuration on page 99*.

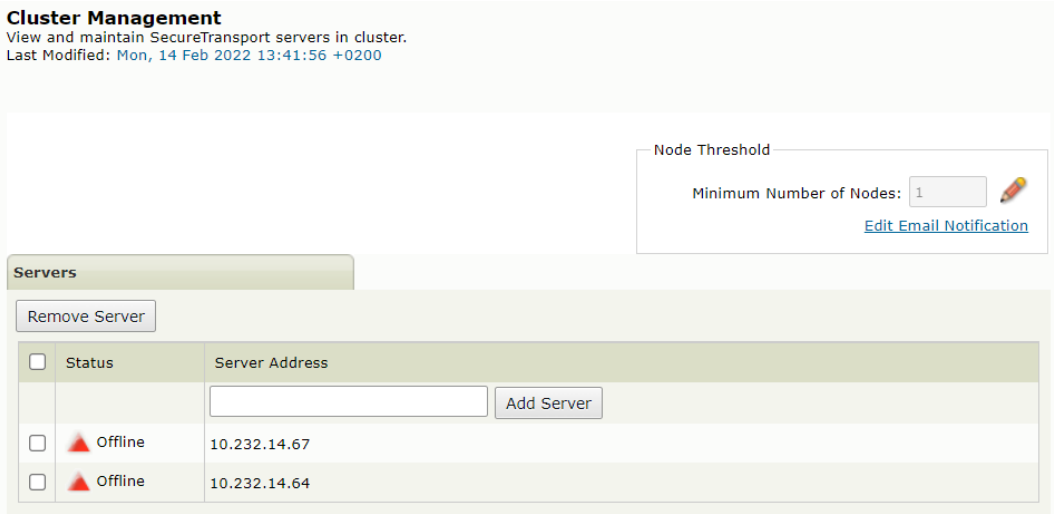
To use IPv6 addresses for communication between servers in an Enterprise Cluster (EC), you must edit the `<FILEDRIVEHOME>/conf/tangosol-coherence-override.xml` file on each server and set the value of the `<address>` element to the IPv6 address of the network interface used for cluster communications. The first lines of the `<cluster-config>` element must be:

```
<cluster-config>
    <unicast-listener>
         <address>IPv6-address</address>
```

Use the following procedure to add a SecureTransport Server to an existing Enterprise Cluster. For instructions on how to set up a cluster, see *Set up a cluster* on page 375.

Note Verify that all necessary SecureTransport plug-ins are deployed on the server prior to adding it to the cluster. If the plug-ins are not present when the server is started, authentication and authorization plug-in configuration will be lost.

1. Stop all services on all cluster nodes by running `<filedrivehome>/bin/stop_all` on each one.
2. Stop all services on the server or servers you want to add to the cluster.
3. Start the Administration Tool on the first installed node by running `<filedrivehome>/bin/start_admin`.
4. Log in to the Administration Tool and select **Operations > Cluster Management**.

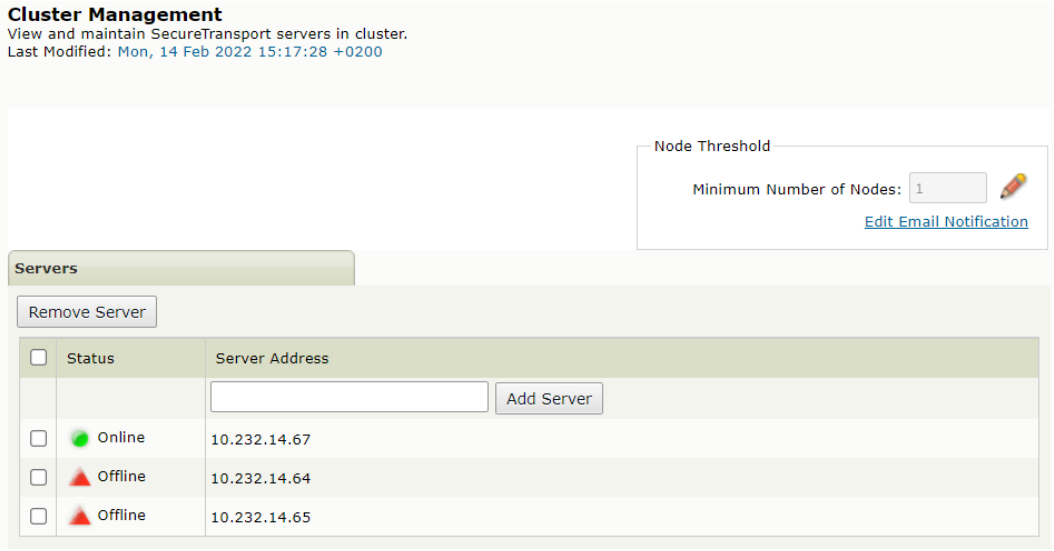


5. In the **Server Address** field in the **Servers** table, enter the IP address or the FQDN of the SecureTransport Server you want to add to the cluster and click **Add Server**. Repeat this step for any other server you want to add.

   Note When adding a server by its FQDN, it must have a successful DNS resolution.

   Note The Administration Tool does not prevent you from adding the same IP address to the cluster more than once, but this is not a valid operation and the cluster may not form if an IP address is duplicated.

6. Start all services on the first installed node by running `<filedrivehome>/bin/start_all`.



   Only the node you are currently logged on to is displayed with an Online status. The newly added server is displayed in the table with an Offline status, indicating it is not yet connected to the cluster.

7. Start all services on the newly added server or servers by running `<filedrivehome>/bin/start_all`.

8. Validate that all servers have been successfully added to the cluster (displayed with an Online status on the *Cluster Management* page).

Note In order for the cluster synchronization to work properly, the Administration Tool service must run on all nodes in the cluster even if you are not accessing them through the Administration Tool. Note that if the `Cluster.enable.SSL` server configuration option is set to `true` and the certificate used for SSL connection has expired or is not chained to a trusted root, the cluster does not form and the Administration Tool service is started only on the node that was started first.

Note The feature license defines the maximum number of nodes in an Enterprise Cluster. If you attempt to start a Transaction Manager that exceeds the maximum number of nodes, the Transaction Manager will not start or process tasks.

Remove a server from a cluster

Use the following procedure to remove a SecureTransport Server from a cluster.

1. Make sure all services are still running on the node or nodes you are about to remove from the cluster.
2. Log in to the Administration Tool (preferably on the first installed node) and select **Operations > Cluster Management**.
3. In the **Servers** table, select the node or nodes to be removed from the cluster.
4. Click **Remove Server**.

   The selected server is removed from the **Servers** table and is no longer a part of the cluster.
5. Run `<filedrivehome>/bin/stop_all` on the removed server.

Note The local server configuration settings for a removed server remain in the shared database. To avoid problems with the shared database, do not start any removed SecureTransport Server until you add it back to the same cluster.

Note If a server is removed from the cluster and added back after a daemon configuration change, the private zone of the secondary server will not be updated to reflect the change. You will need to apply the changes manually by updating the *Server Control* page of the added server and restarting the changed daemons and the Transaction Manager.

View cluster status

Use the following procedure to view the cluster status.

1. Select **Operations > Cluster Management.**

   The *Cluster Management* page is displayed.
2. In the **Servers** table, check the **Status** column.
   - **Online** – The SecureTransport Server is reachable and both the TM and Administration Tool server are running.
   - **Offline** – Either the SecureTransport Server is not reachable, the Transaction Manager is not running, or the Administration Tool server is not running.

SecureTransport makes an entry in the server log when a server in the cluster goes offline.

To refresh the **Status** column, select **Operations > Cluster Management**.

If the status of a SecureTransport Server is Offline, you can check its status in more detail.

- Log in to the server using the Administration Tool, select **Operations > Server Control**, and make sure that the TM Server is running.
- If you cannot connect to the server using the Administration Tool, log in to the computer that hosts the server and make sure that the SecureTransport Administration Tool and TM server are running.

Note If a SecureTransport Server fails, the distributed event manager assigns pending events to other servers, but user sessions connected to the failed server are closed. However, Axway Secure Client automatically reestablishes the closed session and resumes any transfers.

Notification of cluster status

SecureTransport can send an email notification when the number of online servers in the cluster falls below a limit that you set. If enabled, SecureTransport sends an email each time it detects a server failure. You can use this notification for the following purposes, among others:

- Set the limit to one less than the number of nodes in the cluster and use the notification to inform you to restore the node that is offline.
- Set the limit to one less than the number of nodes required for acceptable performance and use the notification to inform you to evaluate whether to fail over to your standby disaster recovery site.

You configure the email notification in the **Node Threshold** topic of the *Cluster Management* page.

1. Select **Operations > Cluster Management**.

   The *Cluster Management* page is displayed.
2. To change the value of the **Minimum Number of Nodes** field:
   a. Click the Edit icon ( ).
   b. Type the new value in the field.
   c. Click the Save icon ( ).
3. To configure the notification email, click **Edit Email Notification**.

   The *Email Notification* page is displayed.
   a. To send the emails, select **Send Notification**.
   b. Type the email subject in the **Subject** field and the email body in the **Notification** field.
   c. Click **Save** to save your changes or **Cancel** to reset the values to the last saved values.
   d. Select **Operations > Cluster Management to** return to the *Cluster Management* page.
4. To set the address the email is sent to, select **Setup > Miscellaneous**.
5. Make sure all the fields in the *FTP/HTTP Startup Password Timeout Configuration* pane have valid values.

Note You must change the default value, `root@localhost`, of the **Notify e-mail** field to a complete and valid email address. This address is used for both the sender address and recipient address.

Set up a disaster recovery cluster

Use the following procedure to set up a disaster recovery cluster.

1. For each SecureTransport Server in your production cluster, edit the `<FILEDRIVEHOME>/conf/options-overwrite.conf` file and replace

   `#Cluster.DeploymentSite=Prod`

   with

   `Cluster.DeploymentSite=Prod`

2. On one SecureTransport Server in the production site:
   a. On the *Server Configuration* page:
      - Set `Cluster.EnableDRConfiguration` to `true`.
      - Make sure that `Cluster.DeploymentSite` is set to `Prod`.
   b. For each network zone node, make sure the **Deployment Site** field is set to `Prod` so that this node will be used when `Cluster.DeploymentSite` is set to `Prod` and define another node for the DR site with the **Deployment Site** field set to `DR`.
3. Deploy a separate cluster that duplicates your production cluster. For information you need when you plan your DR cluster and install the servers, see *Passive disaster recovery* on page 371. To initialize the DR site:
   a. Synchronize the database for the DR cluster with database for the production cluster so that configuration is consistent with the production cluster.
   b. Synchronize the data for the DR cluster with data for the production cluster so that the account home folders and other folders are consistent.
4. For each SecureTransport Server in the DR site:
   a. Copy the `<LocalConfigurationsId>` element in `<FILEDRIVEHOME>/conf/configuration.xml` from the corresponding server in the production site.

      The content of the `<LocalConfigurationsId>` element establishes the correspondence between a production server and its corresponding DR server.
   b. Edit the `<FILEDRIVEHOME>/conf/options-overwrite.conf` file.
      - Replace

        `#Cluster.DeploymentSite=Prod`

        with

        `Cluster.DeploymentSite=DR`
      - Replace

        `#node.ip=`

        with

        `node.ip=`*`IP_address`*

        where *`IP_address`* is the IP address of the system that the SecureTransport Server is running on.

      When SecureTransport Server starts, it updates the database with this IP address.

Note You can use `options-overwrite.conf` to overwrite any local or shared editable server configuration parameter that you can set on the *Server Configuration* page.

5. For each SecureTransport Edge in the DR site:
    a. Update the network zone node of the `Private` network zone so that it references the SecureTransport Servers in the DR cluster.
    b. Export system configuration from the associated production SecureTransport Edge and import it.
6. If you are using a separate shared databases for each site, log in to the Administration Tool on each SecureTransport Server in the DR cluster and change the database to the DR shared database.
7. Set up email notification on the production cluster and define the procedure by which you decide to switch to the DR site and the method you use to switch.
8. Set up and test the data replication from the production cluster to the DR cluster.

Maintain a disaster recovery cluster

Once your DR site is set up, you must maintain consistency with the production site as described in *Passive disaster recovery* on page 371 so that the data in the DR shared database and the DR shared file system are current.

When you edit a configuration file or a script on an active SecureTransport Server or SecureTransport Edge, consider if you need to make the same changes on the corresponding server in the other site.

After you fail over to the DR cluster, you need to restore your production cluster's functionality. Before you switch back to your production cluster and return your DR cluster to standby status, you must replicate the data in the DR shared database and the DR shared file system to the production site.

Disaster recovery failover and fallback

You can use cluster status notification to decide when you must fail over to the disaster recovery site. See *Notification of cluster status* on page 383.

When the production site is unable to process file transfers, failover to the disaster recovery site:

1. If possible, make sure that the disaster recovery site configuration and data is consistent with the production site.
2. On every SecureTransport Server and SecureTransport Edge in the production site, use the `<FILEDRIVEHOME>/bin/stop_all` (or, on Windows, `<FILEDRIVEHOME>\bin\stop_all.com`) command to stop all SecureTransport processes.

3. On every SecureTransport Server and SecureTransport Edge in the DR site, use the `<FILEDRIVEHOME>/bin/start_all` (or, on Windows, `<FILEDRIVEHOME>\bin\start_all.com`) command to start all SecureTransport processes.
4. Make any changes to your load balancers or other network infrastructure to direct your SecureTransport traffic to the disaster recovery site.

When the production site is able to process file transfers, failback:

1. If possible, make sure that any configuration changes made on the DR site are transferred to the production site and make sure that production data is consistent with the disaster recovery site.
2. On every SecureTransport Server and SecureTransport Edge in the production site, use the `<FILEDRIVEHOME>/bin/start_all` (or, on Windows, `<FILEDRIVEHOME>\bin\start_all.com`) command to start all SecureTransport processes.
3. On every SecureTransport Server and SecureTransport Edge in the DR site, use the `<FILEDRIVEHOME>/bin/stop_all` (or, on Windows, `<FILEDRIVEHOME>\bin\stop_all.com`) command to stop all SecureTransport processes.
4. Make any changes to your load balancers or other network infrastructure to direct your SecureTransport traffic to the production site.

Note The time needed to perform a switch to the DR site can be estimated by assessing the time consumed for each of the following stages: stop and start ST services, perform the manual steps and maintain the data integrity. However, the time needed to keep the data integral is dependent on multiple factors, such as the chosen solution for database and filesystem synchronization, deployment specifics, connectivity, amount of data to transfer, distance between remote locations, etc.

Direct cluster workload

As described in *Workload distribution* on page 371, an event processor directs events that represent workload tasks to the SecureTransport Servers in the cluster. Using its default policies, the event processor directs all events associated with an account to the same set of servers. This policy improves the performance of the distributed object cache that SecureTransport uses to avoid database fetches for object references. If all tasks associated with an account are performed by the same set of servers, the object that represents that account and related objects are cached at that server. This improves performance because the cache manager does not need to fetch them from another server.

In addition to the default account-based event management, event processor policies can direct events to particular servers based on attributes of each event. By default, the event processor uses a round-robin policy to direct events to servers in the set that is processing events for the account, directing each event to the next server in sequence.

If a server in your cluster has performance characteristics or other resources required for certain tasks, you can direct events that represent those tasks that to those particular servers. By default the following task-type attributes are defined:

- **FM_TRIGGERED** – Caused by an action of a transfer site that uses the Folder Monitor protocol
- **MAINTENANCE** – Log applications: LogEntry Maintenance, Sentinel Link Data Maintenance, and Transfer Log Maintenance
- **PGP** – PGP encryption or decryption
- **PULL** – Server-initiated incoming transfer
- **PUSH** – Server-initiated outgoing transfer

For example, to improve performance of PGP encryption and decryption, you can include in your cluster a server with better computational performance, and direct all PGP task to that server.

The `EventQueue.DispatchPolicy.name` system configuration parameter controls the behavior of the event queue. Valid values are:

- `cacheBasedPolicy` (default) – Directs all events associated with an account to the same server to improve performance
- `attributeMatchPolicy` – Directs events based on specified attributes
- `roundRobinPolicy` – Causes the event queue to direct each event to the next server in sequence without regard to account or attributes

The `attributeMatchPolicy` is not used when a node in the cluster is overloaded. For more information, refer to the description of the `EventQueue.taskProcessor.threshold` parameter on the *Server Configuration* page.

In each case, either the policy identifies a set of servers or, if no servers match, the criteria, a set that includes the whole cluster. The event manager uses the round-robin method and select the next server from the set in sequence. Of course, round-robin distribution does not occur when only one server is selected by the policy.

You can configure a SecureTransport Server to process tasks of one or more types by editing a local configuration parameter.

1. Log on to the SecureTransport Server.
2. Select **Operations > Server Configuration**.

   The *Server Configuration* page is displayed.
3. Search for the local parameter `EventQueue.taskProcessor.attributes`.

   The value is a comma-separated list of task types to direct to this server. The default value is `PGP=false, PUSH=false, PULL=false, FM_TRIGGERED=false, MAINTENANCE=false`.
4. Edit the value and change `false` to `true` for each event type to direct to this server.
5. Save the new value.
6. Search for the parameter `EventQueue.DispatchPolicy.name`, edit it, and change its

value to `attributeMatchPolicy`.

7. Select **Operations > Server Control** and restart the TM Server.

The distributed Event Manager now prefers the Server for events of the selected types.

If you configure a SecureTransport Server to process PGP tasks, you must make other changes on every node of your cluster.

1. If your deployment uses NFS to provide the shared file system, you must export the file system with the `sync` and `no_wdelay` options.
2. Log on the SecureTransport Server computer.
3. If your deployment uses NFS to provide the shared file system, change the NFS mount to use the `-o noac,sync` options.
4. Make a backup copy and open the `<FILEDRIVEHOME>/brules/local/wptdocuments/Streaming.xml` file in a text editor.
5. Find the following text:

```
<!-- Uncomment the following if you want to enable the
     PGP Event type distribution -->
<!--
<operator name="or" />
   <expression>
      <item>
         <attribute>EventType</attribute>
         <comparator name="equal" />
         <value>Transformation</value>
      </item>
      <operator name="and" />
      <item>
         <attribute>DXAGENT_TRANSFORMATION_TYPE</attribute>
         <comparator name="equal" />
         <value>PGP</value>
      </item>
   </expression>
-->
```

6. Delete the following comment lines from that text:

   `<!--`

   and

   `-->`

7. Save the file.
8. Select **Operations > Server Control** and restart the TM Server.

Note When a SecureTransport Server is configured to process PGP tasks, Sentinel reporting for those transfers is not accurate because Sentinel cannot combine the processes of the transfer.

You can add task types (attributes) to `EventQueue.taskProcessor.attributes` by editing the `EventQueue.submissionConfigurator.additionalAttributes` server configuration parameter on any server in the cluster.

The items in this comma-separated list of attributes are the names of environment variables that you need to use to identify events. For example:

- `DXAGENT_APPLICATION_NAME` is used to identify events for all users who are subscribed to the named application.
- `DXAGENT_SITE_NAME` is used to identify events for all users who are subscribed to a transfer site with the given name.

Once you add task types to `EventQueue.submissionConfigurator.additionalAttributes`, you edit `EventQueue.taskProcessor.attributes` in the same way you do for the standard task types. The items you add have the form:

```
<environment variable>=<value>
```

where the environment variable is listed in `EventQueue.submissionConfigurator.additionalAttributes` and the value selects the events to direct to the server. For example:

- To define a task type for events for all users who are subscribed to an Site Mailbox application named `smbBU1`, the attribute is `DXAGENT_APPLICATION_NAME=smbBU1`.
- To define a task type for events for all users who have a transfer site named `FtpNode3Unit1`, the attribute is `DXAGENT_SITE_NAME=FtpNode3Unit1`.

Zero Downtime Update

Zero Downtime Update (ZDU) is a process to update an active Enterprise Cluster to a later release of SecureTransport with no interruption in file transfers, scheduling and event logging. The solution described here utilizes scripts and services that are made available with Update 5.5-20240627. This means that the first possible option to update SecureTransport with zero downtime will be from 5.5-20240627 to 5.5-20240725. The solution is applicable to all subsequent updates.

The "Zero Downtime Update" chapter is designed to be a comprehensive resource for both experienced administrators who need quick reference steps and those new to the process requiring detailed instructions. It contains the following sections:

- Prerequisites and preparation - lists the requirements that must be fulfilled before initiating the ZDU process
- Detailed process description - offers an in-depth walkthrough, complete with explanations and examples
- Quick steps - provides concise steps to complete a ZDU

- Known limitations - describes known limitations of the current ZDU solution
- Rollback procedure - describes the steps to revert a SecureTransport update with zero downtime

Before you perform a ZDU for the first time, review the whole chapter. Note that there are differences in the procedure depending on whether this is the first or a subsequent zero downtime update.

Important notice

The Zero Downtime Update feature is exclusively available within the Enterprise Pack bundle. The enablement, access and/or use of Enterprise Pack, including Zero Downtime Updates, is only authorized under an active subscription license to SecureTransport specifically licensing Customer to Enterprise Pack. *

ZDU: Prerequisites and preparation

Complete the following steps before you start the ZDU process:

Caution You are strongly advised not to make configuration changes during the entire ZDU process.

Prerequisites

General requirements:

- An entitlement to Enterprise Pack

Environment requirements:

- Enterprise cluster deployment with at least two SecureTransport Servers and SecureTransport Edges that meets the requirements described here.
- If Edge synchronization is enabled in the original environment, it must be deactivated before starting the ZDU.

* For clarification and the avoidance of any doubt, Customer is not entitled to enable, access, or use ZDU or any of the features in Enterprise Pack without such specific contractual entitlement and license to SecureTransport under a subscription contract, and any use of ZDU or Enterprise Pack features without such specific entitlements is beyond the scope of the license to Customer. Any enablement, access, and/or use of Enterprise Pack or ZDU use will require the purchase of subscription license by Customer at additional cost.

- External load balancer:
  - Capable of pointing to all Edge nodes in the Blue and Green clusters
  - Configured to distribute traffic based on the SecureTransport health check service

Database requirements:

- Supported databases: Oracle and PostgreSQL
- ZDU requires three database schemas, two of which must be created empty beforehand. If these schemas are in different databases, the latter must meet the requirements for PostgreSQL or Oracle databases:
  - SharedRuntime - This schema originates from the initial database schema and contains the Audit and Server log records, File tracking, and Scheduler. During the ZDU process, both clusters (Blue and Green) will access this schema. After the ZDU completes, the cluster running the new version (Green) will continue to use it.
  - Blue schema - This schema contains configuration data (such as routes, accounts, etc.) and event data. Before performing ZDU for the first time, you should create this schema, the user and/or tablespaces. It will be used during every subsequent ZDU. Ensure this schema remains open: for example, on Oracle, the schema may lock after some time. If possible, prevent it from locking; if not, unlock the schema before each ZDU.
  - Green schema - This schema will be populated during the ZDU process with configuration data. It will be used by the cluster running the newer version. The replication is performed at the beginning of the ZDU process. Note that the user and/or tablespaces need to be created beforehand. In some cases, for example, if the new schema is on the existing Oracle database used by SecureTransport, the tablespaces do not need to be recreated but the new user should be granted rights to use them instead.
- Sufficient space to accommodate the increased database size resulting from the use of multiple schemas.

Prepare your environment for ZDU

Perform the following steps to prepare your environment for ZDU:

1. Backup the SecureTransport Server and SecureTransport Edge data. See instructions.
2. Prepare the database properties file:
   a. Find the template file *<FILEDRIVEHOME>/conf/zdu/db.properties* and make a copy.
   b. Edit the new *.properties* file to specify the configuration of the Green database that will be used by the updated cluster.
   c. Store the file on shared storage or make a local copy on each node, so that it is accessible to all backend cluster nodes.
3. Configure the SecureTransport health check service. For detailed instructions, see *Health Check Configuration* on page 296
4. Configure your load balancer to use SecureTransport readiness check as shown below:

Protocol: HTTP
Port: as specified in `StatusChecker.port`
Path: `{IP}:{PORT}/readiness`

5. Disable Edge synchronization:
    a. On each Edge server, edit the *<FILEDRIVEHOME>/lib/admin/config/servers* configuration file to remove all the servers.
    b. On the primary server, go to the *<FILEDRIVEHOME>/var/tmp* directory and remove the file named "sentinel_primary".
    c. Log in to the Administration Tool again.

Note After the ZDU process completes, you can enable Edge synchronization back following the procedure described here.

You can now start the ZDU process. Check the following topics for step-by-step update instructions:

- Quick steps - provides concise steps to update SecureTransport with zero downtime.
- Detailed process description - offers an in-depth walkthrough of the ZDU process, complete with explanations and examples.

Configure the database properties file

The database properties file contains connection details (host, port, dbuser, etc.) for two database schemas, Blue and Green, which are switched between during the ZDU process. The ZDU scripts use this file to clone the SecureTransport's configuration data to the new database and to migrate nodes to the new cluster. This file can be reused for subsequent updates and rollbacks.

A template *db.properties* file is located in the *<FILEDRIVEHOME>/conf/zdu* directory. Copy this template to a shared location and then modify the file to suit your migration. Here is what the properties file looks like:

```
#postgresql.1.host=
#postgresql.1.port=
#postgresql.1.username=
#postgresql.1.password=
#postgresql.1.databaseName=
#postgresql.1.useSecureConnection=
#postgresql.1.secure.certificateFile=

#postgresql.2.host=
#postgresql.2.port=
#postgresql.2.username=
#postgresql.2.password=
#postgresql.2.databaseName=
#postgresql.2.useSecureConnection=
#postgresql.2.secure.certificateFile=
```

The file contains configuration properties for two databases of each type - PostgreSQL, Oracle, and MSSQL. The properties are commented out, indicating they are not active. To use them, remove the # at the beginning of each line and specify a value.

PostgreSQL properties

Each set of properties is numbered to distinguish between the two databases.

- *postgresql.1.** - Properties for the first PostgreSQL database.
- *postgresql.2.** - Properties for the second PostgreSQL database.

Property	Description
postgresql.*.host	The FQDN or IP address of the database server.
postgresql.*.port	The port used to access the database server.
postgresql.*.username	The name of the user authorized to create and populate the SecureTransport schema.
postgresql.*.password	The password for the user. It must be encrypted with the SecureTransport Server key.
postgresql.*.databaseName	The database name used to connect to the PostgreSQL Server.
postgresql.*.useSecureConnection	When set to `true`, the database connection will be established using SSL. Default value: `false`.
postgresql.*.secure.certificateFile	The location of the database server certificate file used to establish the secure connection.

Oracle properties

Each set of properties is numbered to distinguish between the two databases.

- *oracle.1.** - Properties for the first Oracle database.
- *oracle.2.** - Properties for the second Oracle database.

Property	Description
oracle.*.host	The host name or IP address of the Oracle server.
oracle.*.port	The port used to access the database server.
oracle.*.username	The name of the user authorized to create and populate the SecureTransport schema.
oracle.*.password	The password for the database user. It must be encrypted with the SecureTransport Server key.

Property	Description
oracle.*.serviceName	The Service name used to connect to the Oracle server.
oracle.*.useSecureConnection	When set to `true`, the database connection will be established using SSL. Default value: `false`.
oracle.*.secure.serverCertificateDN	The Server certificate DN value. If provided, it will be matched against the certificate provided by the database server.
oracle.*.secure.enabledProtocols	A comma-separated list of allowed TLS/SSL protocol versions. Valid values: `SSLv3`, `TLSv1`, `TLSv1.3`, `TLSv1.2`, `TLSv1.1`, `SSLv2Hello`.
oracle.*.secure.enabledCipherSuites	A comma-separated list of allowed cipher suites for secure connection to the database.
oracle.*.secure.certificateFile	The location of the database server certificate file used to establish the secure connection.
oracle.*.useCustomJdbcUrl	If set to `true`, you need to provide a JDBC connection string in the `oracle.*.customJdbcUrl` property. Default value: `false`.
oracle.*.useProxyAuthentication	Set this option to `true` to use the native Oracle proxy authentication feature. Default value: `false`.
oracle.*.proxied.username	The username of the proxied account.

Kerberos-specific properties

Property	Description
oracle.*.useKrbAuthentication	When set to `true`, SecureTransport will connect to the database password-less using Kerberos authentication. Note that the Oracle database server must already be configured for Kerberos. Default value: `false`.
oracle.*.krb.credentialCacheFile	Specify the storage location of the Kerberos credentials cache file.

Property	Description
oracle.*.krb.useSystemConfigurationFile	Configure SecureTransport to refer to the Kerberos configuration file directly. When this option is set to `true`, SecureTransport will not copy the file locally and will not synchronize it between nodes. The specified Configuration File Path will be used for establishing a password-less connection to the database. Default value: `false`.
oracle.*.krb.systemConfigurationFile	Specify the storage location of the Kerberos configuration file (`krb5.conf`).

Microsoft SQL Server properties

Each set of properties is numbered to distinguish between the two databases.

- *mssql.1.** - Properties for the first Microsoft SQL Server database.
- *mssql.2.** - Properties for the second Microsoft SQL Server database.

Property	Description
mssql.*.host	The FQDN or IP address of the database server.
mssql.*.port	The port used to access the database server.
mssql.*.username	The name of the user authorized to create and populate the SecureTransport schema.
mssql.*.password	The password for the user. It must be encrypted with the SecureTransport Server key.
mssql.*.databaseName	The database name used to connect to the Microsoft SQL Server database.
mssql.*.useSecureConnection	When set to `true`, the database connection will be established using SSL. Default value: `false`.
mssql.*.secure.serverCertificateCn	The server certificate CN value. If provided, it will be matched against the certificate provided by the database server.
mssql.*.secure.certificateFile	The location of the database server certificate file used to establish the secure connection.

Property	Description
mssql.*.useCustomJdbcUrl	If set to `true`, you need to provide a JDBC connection string in the `mssql.*.customJdbcUr` property. Default value: `false`.

ZDU: Detailed process description

A zero downtime update (ZDU) can be performed on Enterprise clusters with at least two SecureTransport Servers and two Edges. Our ZDU solution involves two clusters, Blue (the original) and Green (running the new version), that use identical database schemas. The Green cluster is created gradually from the Blue cluster nodes using a rolling update technique: one or a subset of nodes in the Blue cluster is taken offline, updated to a newer version, and brought back online as part of the new cluster. This process is repeated for all Blue nodes until all are updated and joined to the new cluster.

After the first set of updated instances joins the cluster, the administrator must validate the update on the Green environment and activate the new cluster so that it starts receiving traffic. For all subsequent migrated nodes, this process happens automatically as the load balancer starts distributing application traffic to them once they pass its readiness check. Since there is always at least one cluster node running at any given time (current or new), SecureTransport remains operational without experiencing any downtime.

The key to enabling transition and coexistence between the old and new versions of SecureTransport is to ensure they have identical configurations. To achieve this, three database schemas are replacing ST_DATA. We will refer to them as Blue, Green, and SharedRuntime:

- Blue schema - initially empty, this schema will be populated during the first ZDU with configuration (such as routes, accounts, etc.) and event data from the original cluster.
- SharedRuntime schema - the original pre-ZDU database schema that contains ST_DATA objects. It will be accessible and used by both clusters simultaneously. The presence of the Scheduler ensures continuous and uninterrupted execution of server-initiated transfers and scheduled jobs.
- Green schema - a replica of the Blue schema that will be used by the (Green) cluster running the new SecureTransport version. This replication is performed in the beginning of a ZDU process using a script that creates the required tables and populates them with data.

During the ZDU process, when both the old and updated SecureTransport nodes are running and receiving traffic, any transfer request could get routed to either of the two versions. To ensure that there will be no functional differences between the Blue and Green cluster and allow for a rollback, their configurations must remain identical. Therefore, it is extremely important that no configuration changes are made throughout the entire ZDU process. Programmatic access to updating all the configuration settings (including accounts, routes and other transfer-related elements) through the REST API will be disabled. However, in case of emergencies, configuration changes can still be made by administrators using the Administration Tool or accessing the REST API from a web browser. Please note that these will not be replicated and it is the administrator's responsibility to make the changes in both the Blue and the Green clusters.

The following section illustrates the ZDU process:

- The existing cluster running the old SecureTransport version is labeled as "Blue". For example purposes, we use the following Enterprise Cluster setup: three backend servers (BE) and three EDGEs, shared filesystem and shared database that have three schemas respectively for the runtime data, configuration data, and one empty (which to be used later by the updated cluster).
- The cluster formed from the updated Blue instances is referred to as "Green".

Before you start the ZDU, it is recommended to review the prerequisites and preparation steps.

I. Start Maintenance mode

Caution By enabling Maintenance mode, you acknowledge and agree that the use of Enterprise Pack, including Zero Downtime Updates (ZDU), is only authorized under an active subscription license specifically licensing you to Enterprise Pack.

The ZDU process starts with enabling Maintenance mode. You can do that on any backend server. This mode is indicated in the Administration Tool with a red label.

Maintenance mode applies only to SecureTransport Servers, and not to Edge. It restricts automated systems from making configuration changes by blocking the relevant REST API resources. Configuration changes can only be made if absolutely necessary and manually by an administrator using the Administration Tool or accessing the REST API from the web browser. They must be applied uniformly across both clusters to prevent configuration data discrepancies.

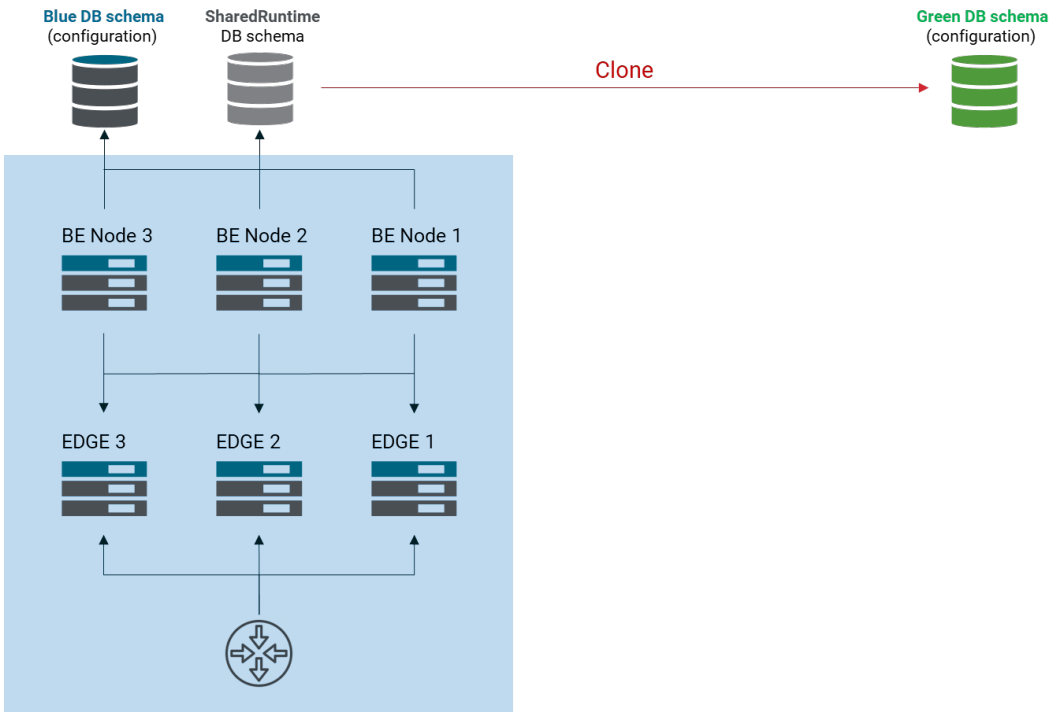
Maintenance mode activation: through the dedicated Rest API endpoint */configurations/maintenance/operations*.

II. Prepare the Green schema

To ensure both cluster environments have identical configurations, you must clone the configuration data from the original schema to the one that will be used by the updated cluster. Use the *clone_blue* script located in the *<FILEDRIVEHOME>/bin/zdu* directory. It requires that the target (Green) database schema be empty and that you provide a path to the properties file with the connection information for the target schema. If you have already performed a ZDU once, for further updates you may enter information on both the Blue and the Green schemas in the file. The script identifies the schema currently in use, creates the required tables, and clones the configuration data to the empty target database schema.

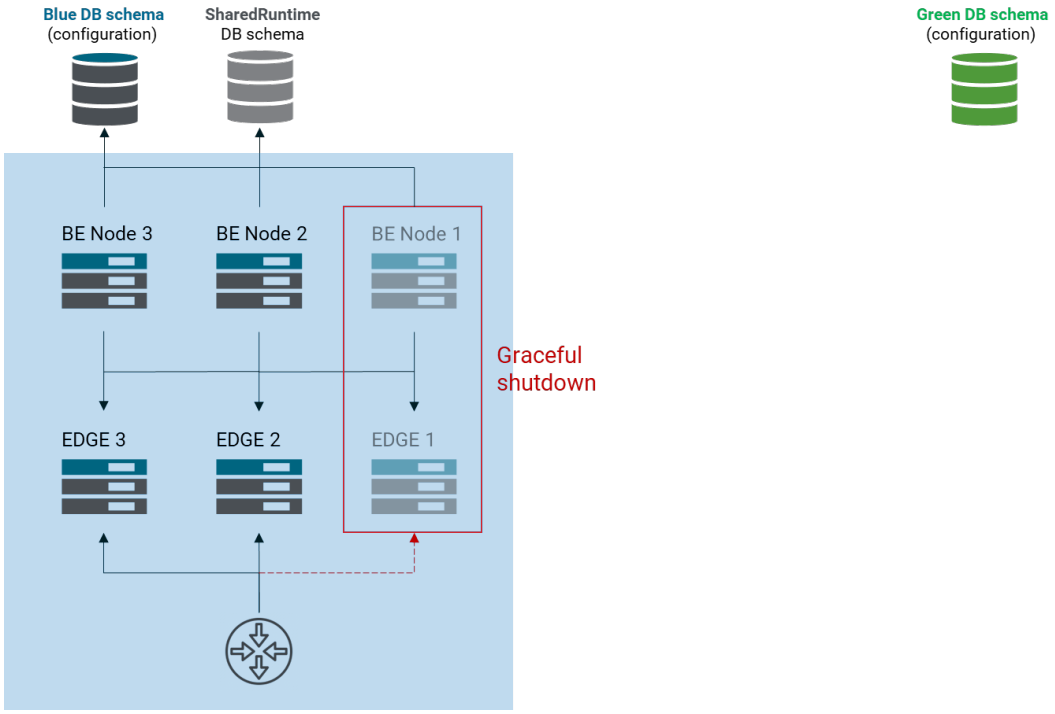
Script usage:

```
./clone_blue -f <full path to the database properties file>
```



III. Remove nodes from the Blue cluster

Choose a subset of the Blue cluster to update. This subset could be either a backend, a backend and an Edge, or a backend with several Edges. There is no restriction on the number of Edge nodes that can be migrated at once. However, they must be migrated together with a SecureTransport Server. When deciding how many instances to take offline, consider that the Blue cluster will operate below its capacity during the update process. To prevent any interruptions in service availability, ensure that there is at least one operational node running the required protocol services.



To gracefully shut down one Server and one Edge, follow the procedure below. Stop the Edge servers first, and then the backend.

1. Navigate to the *<FILEDRIVEHOME>/bin* directory.
2. Stop the Monitor server:

   ```
   ./stop_monitor
   ```

3. Initiate the graceful shutdown:

   ```
   ./stop_all -g
   ```

   By default, the grace period for the instance to complete ongoing requests is set to 86400 seconds /24 hours/. If a timeout is not set explicitly, the default is used. You can set a new timeout in seconds as shown below:

   ```
   ./stop_all -g -timeout <interval_in_seconds>
   ```

4. Wait until it's completely stopped.

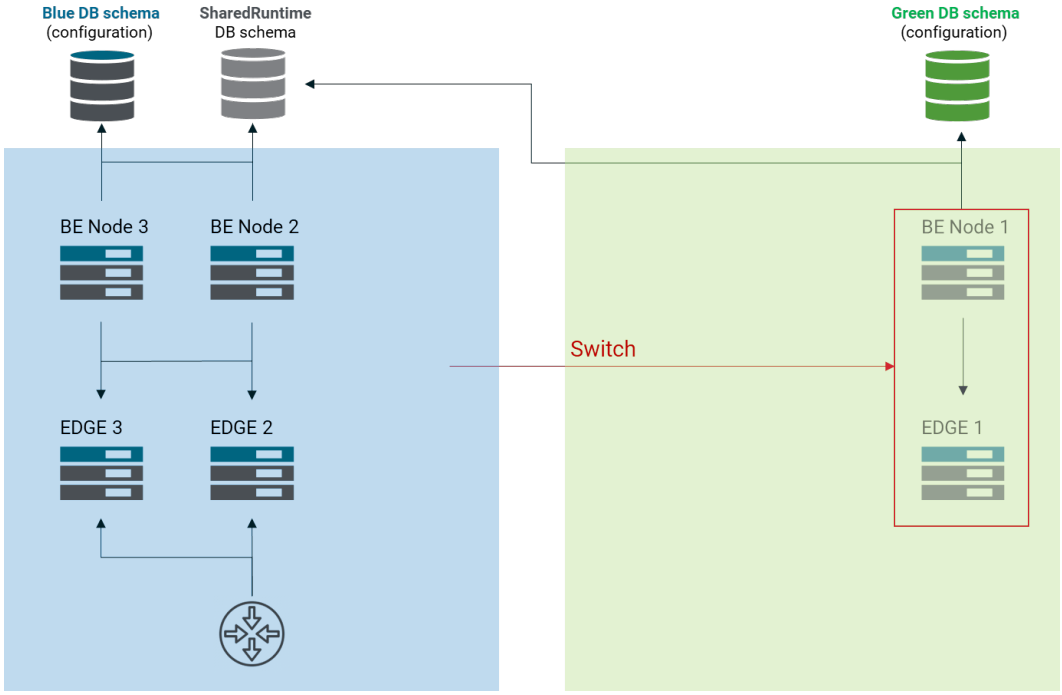
For additional information, refer to Graceful shutdown.

IV. Add the first set of nodes to the Green cluster

At this step, we are going to move the instances that have been gracefully shutdown out of the Blue cluster to the new Green cluster. The script that automates the process, *switch_blue_green*, is located in the *<FILEDRIVEHOME>/bin/zdu* directory.

Script usage:

```
./switch_blue_green -f <full path to the database properties file> -e <edgeIP>
```

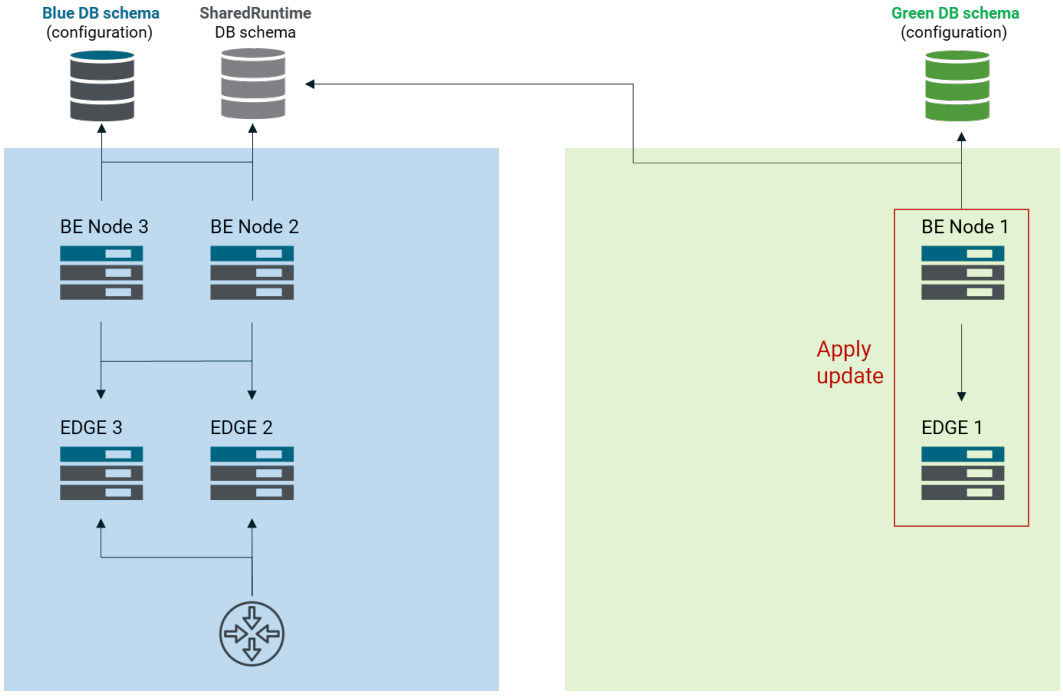


What does the script do? It identifies the instance by its IP address, and

- Checks the database properties file, identifies the source and the target database schema, and tests the connection to the target database schema.
- For a SecureTransport Server, removes it from the Blue cluster and adds it to the cluster node table of the Green cluster, makes the necessary changes in the nodes's *configuration.xml* file, so that it uses both the Green and the SharedRuntime schemas.
- For an Edge, removes the entry from the Blue network zone table and adds it to the Green one.
- Only when adding the first node to the Green cluster: the script sets the `Zdu.Validate.Update` configuration option to `false`, so that the cluster will not process traffic until the update is validated.

V. Update Green instances to a newer version

Update the Green instances to a newer SecureTransport version. Step-by-step instructions are provided in each update's README file.

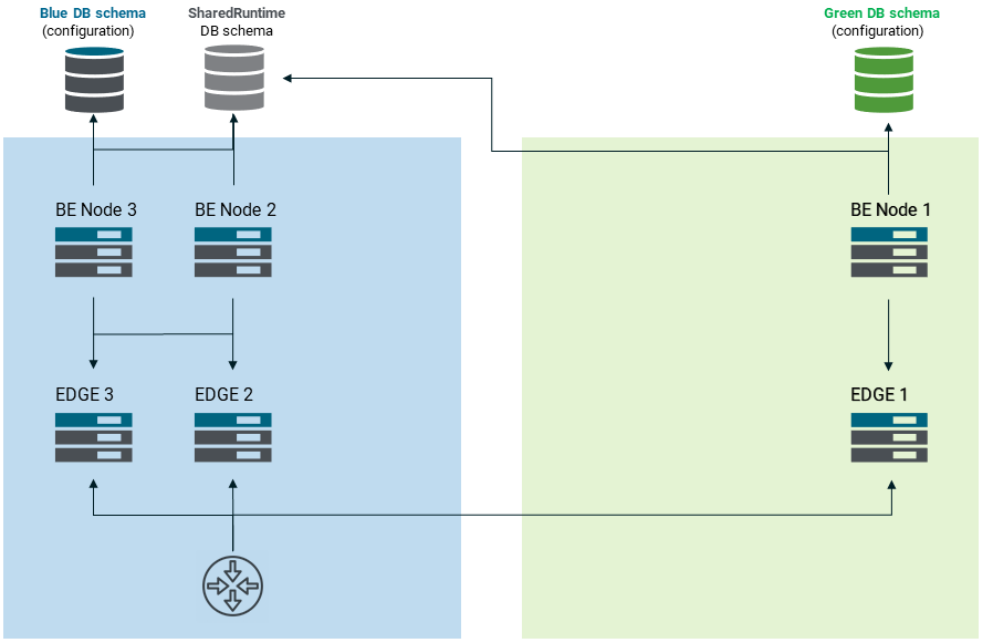


VI. Validate the update

When the update completes, perform tests to validate the updated instance functionality. If the Green cluster works as expected, you can bring it online:

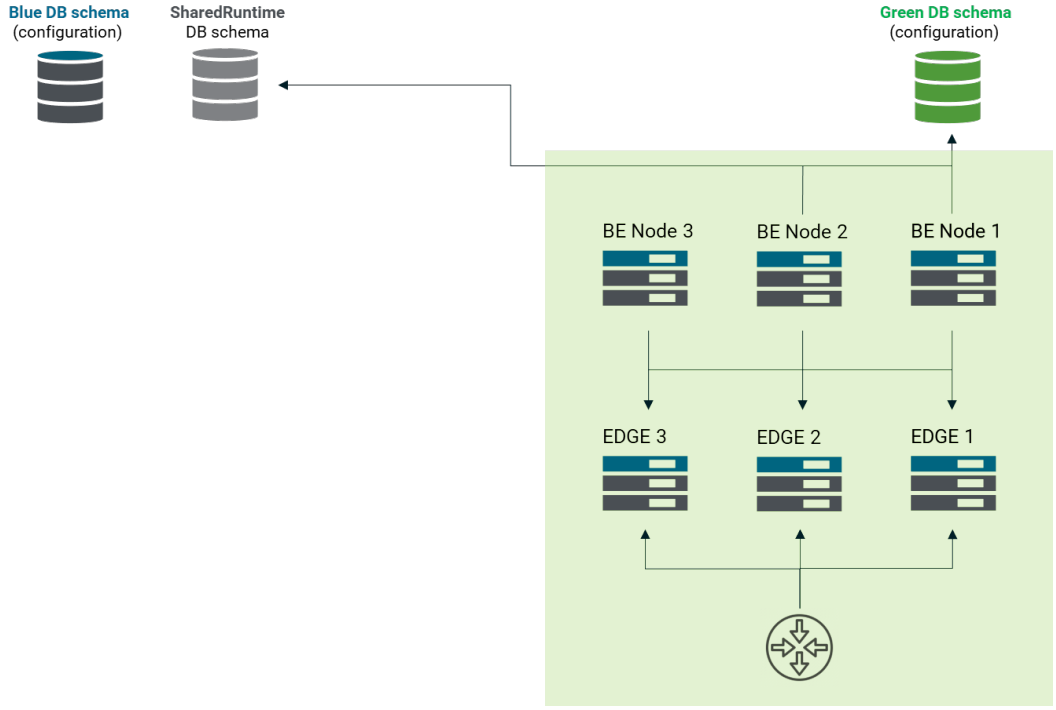
1. On each node of the Blue cluster, stop the Folder Monitor.
2. On the newly updated backend server (Green), set the `Zdu.Validate.Update` configuration option to `true`. Enabling this option starts the Folder Monitor and Scheduler on the Green cluster. It is one of the requirements for the readiness check.

From this moment on, old and updated SecureTransport nodes are running and receiving traffic.



VII. Gradually migrate Blue nodes to the Green cluster until all of them have been updated

To migrate a subsequent set of nodes from the Blue to the Green cluster, repeat *Steps III* to *V*. Once the update completes, the nodes will immediately start receiving traffic, as the `Zdu.Validate.Update` configuration option has already been set to `true`.



This process is repeated for each node until all nodes from the original Blue cluster have been migrated or updated to the new Green cluster.

VIII. Stop Maintenance mode

Turn off Maintenance mode using the REST API */configurations/maintenance/operations* endpoint.

Known limitations that will be targeted in future releases

The current ZDU solution does not support the following:

- SecureTransport with Oracle on more than one database schema
- Flat file for Server Log
- Changing database configuration through the Administration Tool after ZDU
- Disaster recovery
- User password change/self-lock
- AdHoc auto-enrollment
- SecureTransport Edge servers synchronization
- Blue schema clean-up after the ZDU process completes
- Message in the Administration Tool indicating the start of the ZDU
- Automatic Folder Monitor switch
- Health check via TLS
- Graceful shutdown: With HTTP, when you initiate a graceful shutdown during active file uploads with ST Web Client, these uploads will be processed until the current chunk upload is completed. This might stop the upload of files larger than the configured chunk size.

Quick steps

This section provides concise steps to update SecureTransport without any downtime. If you perform a ZDU for the first time, check *ZDU: Detailed process description* on page 396 for in-depth explanation of each step.

Caution ZDU can be applied on environment with at least two SecureTransport Servers (backends). There is no restriction on the number of Edge servers that can be migrated at once. However, they must be migrated together with a backend.

Before you start the update procedure, ensure all prerequisites are met and the environment is ready for ZDU.

1. Enable Maintenance mode through the REST API */configurations/maintenance/operations* endpoint.

2. Clone SecureTransport configuration data from the current schema to the new one that will be used by the updated cluster.
   On a SecureTransport Server, run the *clone_blue* script located in the *<FILEDRIVEHOME>bin/zdu* directory and pass the location to the database properties file:

   ```
   ./clone_blue -f <full path to the database properties file>
   ```

   Example: `./clone_blue -f /FDH/SecureTransport/conf/zdu/db.properties`

3. On each SecureTransport instance you want to update, navigate to *<FILEDRIVEHOME>/bin/* and stop the Monitor service first and then initiate graceful shutdown. Stop the Edge servers first, and then the backend server.

   ```
   ./stop_monitor
   ./stop_all -g
   ```

4. On the stopped backend server, run the *switch_blue_green* script that migrates Blue nodes to the Green cluster. The script is located in the *<FILEDRIVEHOME>bin/zdu* directory.

   ```
   ./switch_blue_green -f <full path to the database properties file> -e <edgeIP>
   ```

   where `-e <edgeIp>` is the IP or FQDN address of the Edge that you want to migrate.

   Caution The `<edgeIp>` value must match exactly how the Edge was configured in the original network zone. You can check the configuration in the Administration Tool **Setup > Network Zones**.

   The Edge parameter is optional and may not be provided at all. Also, it's possible to specify multiple Edge IP addresses in the following manner:
   `-e edge1.host.name -e edge2.host.name`

5. Apply the SecureTransport update (5.5-20240725 or later) following the instructions in the update's README file.
6. Validate SecureTransport functionality after update. If the updated cluster works as expected, bring it online by setting the *Zdu.Validate.Update* configuration option to *true*.
7. Repeat *Steps 3 to 5* to migrate subsequent sets of nodes until all are running the new version.

   The newly added nodes start automatically once the update completes successfully.
8. Turn off Maintenance mode through the REST API */configurations/maintenance/operations* endpoint.

Rollback of SecureTransport updated with zero downtime

This section describes the steps to roll back a SecureTransport update with zero downtime. The ZDU process is reversible at any time, with different procedures depending on whether you roll back configuration changes have been made in the cluster running the newer version.

Caution If this is the first time you perform a zero downtime update, use the procedure for rollback described in Scenario 2.

Scenario 1: No configuration changes in the updated cluster

If the update does not work as expected, you can revert the updated nodes to the original Blue environment:

1. Check the database properties file to make sure that it contains the properties for the target schema.
2. Check if the Green cluster is in Maintenance mode. If not, enable it through the REST API */configurations/maintanance/operations* endpoint.
3. On each SecureTransport instance you want to roll back, navigate to *<FILEDRIVEHOME>/bin/*, stop the Monitor service first and then initiate graceful shutdown. Stop the Edge servers first, and then the backend server.

   ```
   ./stop_monitor
   ./stop_all -g
   ```

4. On the stopped backend server, run the *switch_blue_green* script that migrates Green nodes to the Blue cluster. The script is located in the *<FILEDRIVEHOME>bin/zdu* directory.

   ```
   ./switch_blue_green -f <full path to the database properties file> -e <edgeIP>
   ```

   where `-e <edgeIp>` is the IP or FQDN address of the Edge that you want to migrate.

   Caution The `<edgeIp>` value must match exactly how the Edge was configured in the original network zone. You can check the configuration in the Administration Tool **Setup > Network Zones**.

   The Edge parameter is optional and may not be provided at all. Also, it's possible to specify multiple Edge IP addresses in the following manner:
   `-e edge1.host.name -e edge2.host.name`

5. Revert the SecureTransport update following the instructions in the update's README file.
6. Repeat *Steps 2 to 4* to migrate subsequent sets of nodes until all are running the initial version.
7. Turn off Maintenance mode through the REST API */configurations/maintanance/operations* endpoint.

Scenario 2: Configuration changes made in the updated cluster

If configuration changes have been made in the updated environment, follow these steps:

1. Check the database properties file to make sure that it contains the properties for the target schema (Blue).
2. Put the Green cluster into Maintenance mode through the REST API */configurations/maintanance/operations* endpoint.
3. Clone SecureTransport configuration data from the Green to the new (Blue) database that will be used by the reverted cluster nodes.
   On a SecureTransport Server, run the *clone_blue* script located in the *<FILEDRIVEHOME>bin/zdu* directory and pass the location to the database properties file:

   ```
   ./clone_blue -f <full path to the database properties file>
   ```

4. On each SecureTransport instance you want to roll back, navigate to *<FILEDRIVEHOME>/bin/* and stop the Monitor service first and then initiate graceful shutdown. Stop the Edge servers first, and then the backend server.

   ```
   ./stop_monitor
   ./stop_all -g
   ```

5. On the stopped backend server, run the *switch_blue_green* script that migrates Green nodes to the Blue cluster. The script is located in the *<FILEDRIVEHOME>bin/zdu* directory.

   ```
   ./switch_blue_green -f <full path to the database properties file> -e <edgeIP>
   ```

   where `-e <edgeIp>` is the IP or FQDN address of the Edge that you want to migrate.

   Caution The `<edgeIp>` value must match exactly how the Edge was configured in the original network zone. You can check the configuration in the Administration Tool **Setup > Network Zones**.

   The Edge parameter is optional and may not be provided at all. Also, it's possible to specify multiple Edge IP addresses in the following manner:
   `-e edge1.host.name -e edge2.host.name`

6. Revert the SecureTransport update following the instructions in the update's README file.
7. Validate SecureTransport functionality after the revert. If the updated cluster works as expected, bring it online by setting the *Zdu.Validate.Update* configuration option to *true*.
8. Repeat *Steps 4 to 6* to migrate subsequent sets of nodes until all are running the older version.

   The nodes will automatically start once the revert completes successfully.
9. Turn off Maintenance mode through the REST API */configurations/maintanance/operations* endpoint.

SecureTransport Edge synchronization 6

You can deploy SecureTransport Edge servers so that configuration changes are dynamically synchronized from a primary SecureTransport Edge to the other (secondary) SecureTransport Edge servers. This synchronization works like configuration synchronization in a Standard Cluster (SC), but because SecureTransport Edge servers do not process events, they are not in a cluster.

You configure the secondary SecureTransport Edge servers on the primary SecureTransport Edge server. Most configuration changes are dynamically synchronized across the SecureTransport Edge servers immediately.

The following topics describe and provide how-to instructions to synchronize SecureTransport Edge servers:

- *Manual synchronization* on page 407 - Describes manual synchronization of SecureTransport Edge servers.
- *Requirements for synchronization* on page 408 - Lists the requirements for synchronizing SecureTransport Edge servers.
- *What information is synchronized* on page 408 - Lists the information that is synchronized.
- *Set up SecureTransport Edge servers for synchronization* on page 409 - Provides the how-to instructions for setting up SecureTransport Edge servers for synchronization.
- *Manually synchronize SecureTransport Edge servers* on page 410 - Provides the how-to instructions for manually synchronizing SecureTransport Edge servers.
- *Maintain synchronized SecureTransport Edge servers* on page 410 - Describes maintaining synchronized SecureTransport Edge servers and provides how-to instructions for maintaining synchronized SecureTransport Edge servers.

Manual synchronization

You can also propagate configuration information across a Standard Cluster (SC) by synchronizing the secondary servers from the primary server manually.

A full synchronization can take some time to complete, so perform it when your SecureTransport system is not under heavy load.

Perform manual synchronization after you:

- Upgrade SecureTransport Edge
- Restart all the SecureTransport Edge servers
- Restore a failed primary SecureTransport Edge server

- Add or update administrators or administrative roles, either using the Administration Tool or by importing account configuration
- Add or remove SecureTransport Edge servers
- Change the primary server
- Restart the Administration Tool server on a secondary SecureTransport Edge server, if you made changes using the Administration Tool on the primary server while it was down

Requirements for synchronization

Cluster synchronization requires the following:

- The `<FILEDRIVEHOME>/lib/admin/config/servers` configuration file is correct and identical on all SecureTransport Edge servers.
- The `<FILEDRIVEHOME>/var/tmp/sentinel_primary` file exists on the primary SecureTransport Edge only.
- A shared common secret file (named `taeh` file) is used on all servers.
- Each server is hosted on a different computer or VM.
- All servers use the same installation path.
- All the servers are on the same LAN.
    - The primary administrator user ID is the same on all servers and all have the same password.
- The clocks are set to the same time on all servers.
- The internal CAs on all servers is trusted by all other servers. Optionally, you can import the same internal CA on all servers.
- All database settings are identical on all the computers that will be synchronized.
- Only files used for server configuration are configured for synchronization.
- All the server certificates are issued by a common CA.

What information is synchronized

SecureTransport Edge synchronization moves configuration data from the primary SecureTransport Edge server to all the SecureTransport Edge secondary servers only. SecureTransport does not support moving data from secondary servers to the primary server or from the primary server to selected secondary servers.

The following information is synchronized dynamically from the primary SecureTransport Edge server to the secondary SecureTransport Edge servers when you change it on the primary server:

- All server configuration parameters that are not local to the server

The following information is also synchronized during manual synchronization:

- All configuration files listed in the `<FILEDRIVEHOME>/conf/sync.conf` file
- All database tables listed in the `<FILEDRIVEHOME>/conf/sync_tables.conf` file

The files listed in `<FILEDRIVEHOME>/conf/sync.excl` are not copied from the primary to the secondary server.

Set up SecureTransport Edge servers for synchronization

Use the following procedure to set up SecureTransport Edge servers for synchronization.

1. Install SecureTransport Edge on the system that will be the primary server.
2. Copy the `taeh` file from the `<FILEDRIVEHOME>/bin/` directory to all the other systems.
3. Using the `taeh` file, install SecureTransport Edge on the other systems.
4. Add licenses for all servers. For instructions, refer to the *SecureTransport Getting Started Guide*.
5. Generate an internal CA on each server. For instructions, refer to the *SecureTransport Getting Started Guide*.
6. Exchange CA certificates between all servers in a Standard Cluster (SC) or Enterprise Cluster (EC). For details, refer to the procedures for exporting and importing SecureTransport Server CA certificates in the *SecureTransport Getting Started Guide*.
7. On the primary and all secondary servers, list all the servers in the `<FILEDRIVEHOME>/lib/admin/config/servers` configuration file. List the primary server first and continue with the secondary servers.

   Edit the file and add a line of following form for each server in a cluster:

   `<host> https://<host>:<port>`

   where:

   - `<host>` is the FQDN or IP address of the system
   - `https://<host>:<port>` is the URL of the Administration Tool on that system
   - `<port>` is usually 444

   The two fields are separated by a tab character.

   The `<FILEDRIVEHOME>/lib/admin/config/servers` file must be the same on all computer in your cluster. You can create it on the primary server and copy it to the others.
8. On the primary server, create a file named `<FILEDRIVEHOME>/var/tmp/sentinel_primary`. This file is not used for integration with Axway Sentinel. It is required whether or not Sentinel is used.

   To create the file, you can use the `touch` command in UNIX or create an empty file with no file extension in Windows. The file must have 0 bytes.

9. Log out of the primary SecureTransport Edge server and log in again. Make sure that the server is identified as the primary server and that the **Synchronize All** and **Bounce All** buttons are displayed.
10. Synchronize the secondary servers manually from the Administration Tool of the primary server.
11. On the primary server, either import an external CA or generate a local CA. For instructions, refer to the *SecureTransport Getting Started Guide*. Dynamic synchronization copies the new CA to all servers in a cluster.
12. On the primary server, generate the server certificate required by your configuration and complete other configuration tasks.

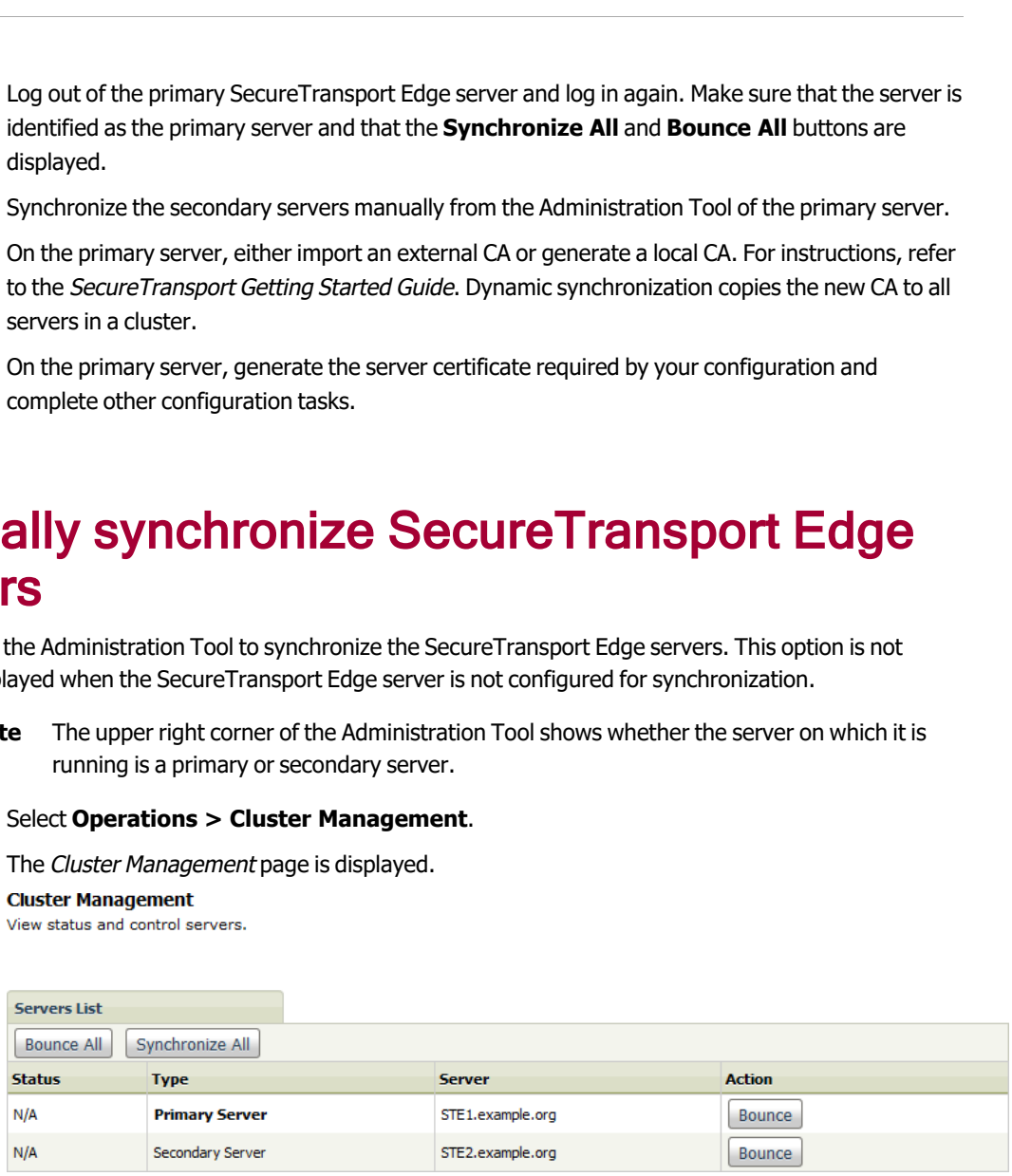
Manually synchronize SecureTransport Edge servers

Use the Administration Tool to synchronize the SecureTransport Edge servers. This option is not displayed when the SecureTransport Edge server is not configured for synchronization.

Note The upper right corner of the Administration Tool shows whether the server on which it is running is a primary or secondary server.

1. Select **Operations > Cluster Management**.

   The *Cluster Management* page is displayed.



2. Click **Synchronize All**.

Note The Status column *Cluster Management* page contains N/A entries because the Transaction Manager does not run on the SecureTransport Edge .

Maintain synchronized SecureTransport Edge servers

As long as your synchronized SecureTransport Edge servers meet the requirements for synchronization, you can make changes to the group, such as:

- Change the primary server (manually failing over the primary server to another server or restoring a server to its role as primary server)

- Add a server
- Remove a server

The key requirement is that the `<FILEDRIVEHOME>/lib/admin/config/servers` configuration file is correct and identical on all servers and that only the primary server has a `<FILEDRIVEHOME>/var/tmp/sentinel_primary` file.

1. Stop all the SecureTransport Edge servers.
2. If you are adding or removing a secondary server, update the `<FILEDRIVEHOME>/lib/admin/config/servers` file on the primary server and copy the servers file to all the SecureTransport Edge server systems.
3. If you are changing the primary server, list it as the first line in the `<FILEDRIVEHOME>/lib/admin/config/servers` file on the primary server, copy the `servers` file to all the SecureTransport Edge server systems, delete the `<FILEDRIVEHOME>/var/tmp/sentinel_primary` file on the previous primary server, and create the `<FILEDRIVEHOME>/var/tmp/sentinel_primary` file on the new primary server.
4. Restart all the SecureTransport Edge servers.
5. Log in to the Administration Tool and manually synchronize the servers.

SiteMinder integration 7

SecureTransport can be integrated into a SiteMinder SSO environment and use SiteMinder to SSO authenticate and authorize resource access using only HTTP or HTTPS.

Note You cannot configure both SiteMinder integration and LDAP integration.

The following topics provide a SiteMinder integration overview and how-to instructions for managing the SiteMinder integration:

SiteMinder overview

SiteMinder is a third-party application that controls user access to secured applications and provides a Single Sign-On (SSO) portal. A Single Sign-On portal is a Web gateway or proxy that enables users to access multiple secured Web applications using a single user name and password that are provided once only at the start of the user session.

SecureTransport can be integrated into a SiteMinder SSO environment using Agent API. When integrated, users can authenticate using SiteMinder SSO. In addition, users can authenticate via FTP and SFTP though SiteMinder using user name and password.

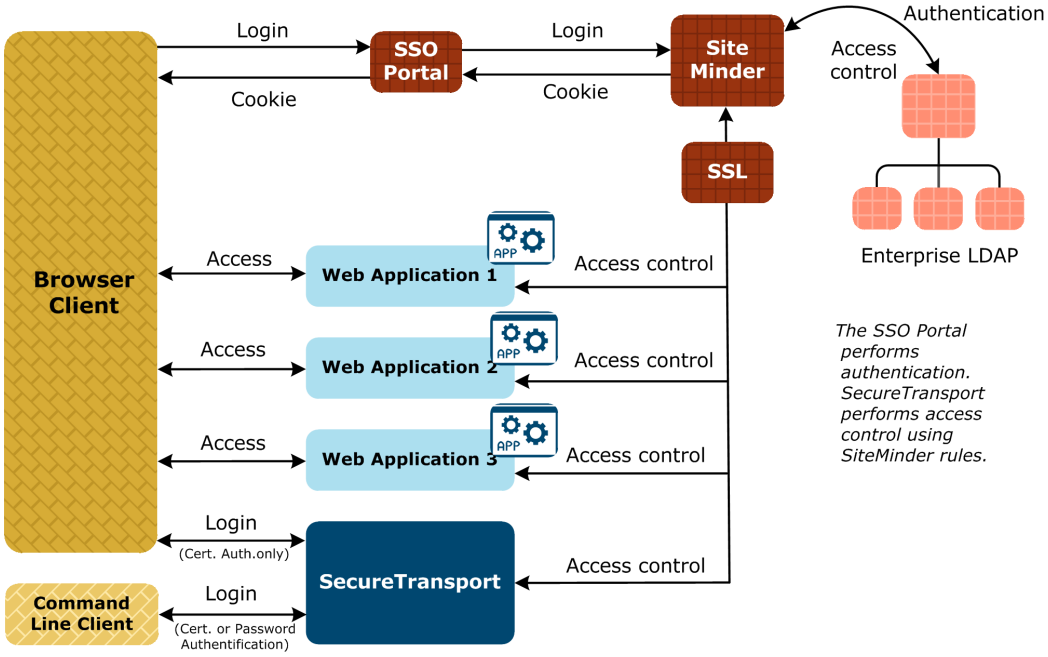


Figure 5. Integration of SecureTransport into a SiteMinder deployment

Typically, in a SiteMinder SSO environment, users log onto the SSO portal using a browser client. The SSO portal directly handles all user authentication and access control for the secured Web applications in the SiteMinder network. However, because SecureTransport supports command line clients, it requires more control over authentication and authorization than a typical Web application. The following topic describes how SecureTransport user authentication and authorization is handled in a SiteMinder environment.

User authentication

In a SiteMinder environment, a user can access SecureTransport resources using a web client or a command line client. If the SecureTransport SiteMinder module is enabled, SecureTransport uses different authentication methods depending on the type of client.

Note All SiteMinder users are represented in SecureTransport by default as virtual users, unless no other specific user class is created for them from **Access > User Classes**.

The following topics describe SiteMinder authentication:

- *Web client (HTTP and HTTPS) user authentication* on page 414 - Describes SiteMinder web client user authentication.
- *Command line client (FTP, FTPS, HTTPS, and SSH) user authentication* on page 414 - Describes SiteMinder command line client user authentication.
- *User access control (authorization)* on page 414 - Describes user access control authorization and authentication.

Web client (HTTP and HTTPS) user authentication

Web clients can log into SecureTransport directly or through the SiteMinder Single Sign-On (SSO) portal. Depending on the SiteMinder configuration, when a web client logs in, SecureTransport can require a client certificate, which it presents to SiteMinder for authentication.

When a web client logs on through the SiteMinder SSO portal, the portal authenticates the user and provides the client with a SiteMinder session cookie. When the client tries to access a SecureTransport resource, SecureTransport presents the user's session cookie to SiteMinder for authentication.

Requirements for the SiteMinder SSO portal:

- It must be in the same domain as the SecureTransport Server because session cookies are by domain.
- It must be accessed using a fully-qualified domain name (FQDN) because SiteMinder uses domain cookies, and it is possible that different browsers can handle the conversion from partial name to FQDN incorrectly. In the latter case, access to the SSO portal can be denied.

Command line client (FTP, FTPS, HTTPS, and SSH) user authentication

Command line clients log on to SecureTransport directly, not through the SSO portal. If the client presents a user name and password or a certificate, SecureTransport first tries to authenticate the user by comparing it to the existing user profiles. In case SecureTransport cannot authenticate the user, it passes the user name and password to SiteMinder for authentication. If all authentication attempts fail, the user receives an error message and is disconnected.

Note The SecureTransport command line client does not work through a Web gateway or proxy (even when using HTTPS). So, SecureTransport Server must be made directly accessible to command line clients.

User access control (authorization)

Regardless of the type of client, access control is always performed by SecureTransport, which uses SiteMinder rules to obtain the user's read/write authority to a particular resource. The SSO portal cannot perform access control for SecureTransport resources.

After a user is authenticated, SecureTransport sends an authorization request to SiteMinder to determine whether the user has read/write access to the requested resource. If SiteMinder grants the appropriate access level, SecureTransport proceeds with the file operation.

Note The SiteMinder SSO portal must be accessed using FQDN because SiteMinder uses domain cookies and it is possible that different browsers can handle the conversion from partial name to FQDN in an incorrect manner. In the latter case, access to the SSO portal can be denied.

The following topics describe the SiteMinder authorization rules and provide an authorization request example:

SiteMinder authorization rules

Configure the SiteMinder authorization rules to accommodate SecureTransport authorization requests. A SecureTransport authorization request contains the following elements:

- **Resource Path** – The absolute URI of the resource being accessed by the user. SecureTransport determines the actual file path and uses that file path as the URI. This URI can be modified (section removed or prefix added) based on the **File Storage Root Path** and **SiteMinder Path Prefix** in the SecureTransport SiteMinder settings. (For details, see *Configure SiteMinder settings in SecureTransport* on page 416.) For Windows systems, backslashes (\) in the file path are replaced with forward slashes (/) before the file path is modified as described above.
- **Command** – Either `GET` or `PUT` depending on whether the user is requesting a read or write operation.

Example authorization request

The following example shows how SecureTransport would construct an authorization request to SiteMinder.

Assume a user requests a listing for the following resource:

```
/mnt/data/SecureTransport/MyDirectory
```

If the **File Storage Root Path** (in the SecureTransport SiteMinder settings) is configured as:

```
/mnt/data/SecureTransport
```

and the **SiteMinder Path Prefix** (in the SecureTransport SiteMinder settings) is configured as:

```
/ST
```

the authorization request for this file operation would be:

```
GET /ST/MyDirectory/
```

In this example, SiteMinder authorization rule would be configured to allow or deny the `GET` command to access the `/ST/MyDirectory/` resource. When SiteMinder is enabled, all SecureTransport users must have `GET` access to the path specified in the **SiteMinder Path Prefix** to successfully log in. If the **SiteMinder Path Prefix** setting is left blank, then users must have `GET` access to the / directory. The SiteMinder Policy Server must be set accordingly.

Configure SiteMinder for SecureTransport integration

To successfully integrate SecureTransport with SiteMinder, SiteMinder must be configured appropriately using the SiteMinder administration system. This topic provides general guidelines for configuring SiteMinder 4.X and 5.X. For more information, refer to the SiteMinder documentation.

1. Create a SiteMinder agent that SecureTransport is to use to connect to the SiteMinder Policy Server.

   When creating the agent, either select the **4.X** compatibility option and fill in the **IP address** of the SecureTransport Server (not the SiteMinder Policy server) and **Shared secret** or select the **5.X** compatibility option and fill in the name of the agent only.
2. Create an authentication scheme by selecting one of the following types:
   - **Basic Template** – password authentication
   - **X509 Client Cert Template** – certificate authentication
   - **X509 Client Cert or Basic Template** – certificate *or* password authentication
   - **X509 Client Cert and Basic Template** – certificate *and* password authentication
3. Create a Realm, selecting the agent and authentication scheme that have been created for SecureTransport.
4. Create new rules under the Realm.
5. Create a Response that returns the attributes required by the SecureTransport SiteMinder settings. For details, see *SiteMinder integration configuration* on page 470.
6. Apply the new rules to the necessary SiteMinder Policy.

Configure SiteMinder settings in SecureTransport

Before you start with SecureTransport configuration, you must have the following:

- SiteMinder Policy Server installed and configured. See *Configure SiteMinder for SecureTransport integration* on page 416.
- SecureTransport installed and operational
- SiteMinder SDK (e.g., siteminder.ca-sdk-12.8-win64.exe) downloaded

JAR files and dependencies

The integration mechanism requires four SiteMinder binary files to be added to SecureTransport:

1. Go to `<FILEDRIVEHOME>/lib/jars/external`.
2. Ensure that there are no SiteMinder binaries.

   On SecureTransport instances running a version older than 5.5-20230727, delete the following SiteMinder files:

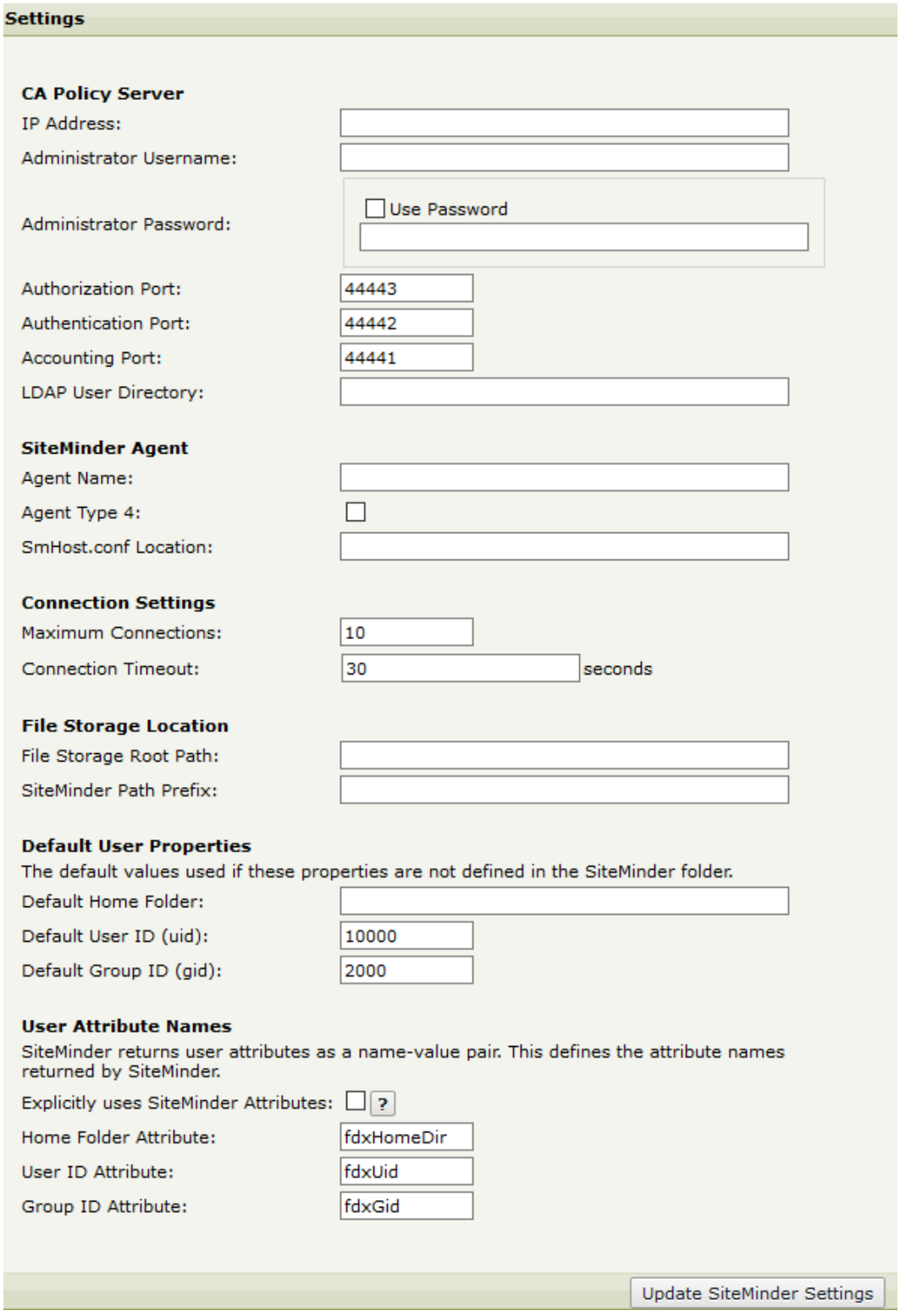
   - siteminder-cryptoj.jar
   - siteminder-smagentapi.jar
   - siteminder-smjavaapi.jar
   - siteminder-smjavasdk2.jar
3. Install SiteMinder SDK on the machine running SecureTransport.
4. Copy the following files from the `<SDK Install Directory>/java` folder to the `<FILEDRIVEHOME>/lib/jars/external` folder:

   - smagentapi.jar
   - smcrypto.jar
   - smjavaagentapi.jar
   - SmJavaApi.jar
   - smjavasdk2.jar
5. Restart all services.

Configure authentication with SiteMinder server

The authentication with SiteMinder server is configured through the SecureTransport Administration Tool, *SiteMinder Settings* page. This page is not available on SecureTransport Edge. If SecureTransport is deployed in a secure perimeter network (DMZ) configuration, configure the SiteMinder settings on SecureTransport Server as described below.

1. Select **Authentication > SiteMinder Settings**.



2. Complete the configuration as described in the following table:

Name	Description	Required/ optional
IP Address	The network address of the SiteMinder Policy Server.	Required
Administrator Username	The user name used to connect to the SiteMinder database.	Optional
Administrator Password	If a password is required, select Use Password and enter it in the field provided. **Note** Exported configuration from SecureTransport 4.x.y systems does not include the SiteMinder administrator password.	Optional
Authorization Port	The authorization port for the SiteMinder Policy Server.	Required
Authentication Port	The authentication port for the SiteMinder Policy Server.	Required
Accounting Port	The accounting port for the SiteMinder Policy Server.	Required
LDAP User Directory	Name of the SiteMinder user directory used to retrieve the home folder, user ID, and group ID.	Optional
Agent Name	The name for the SiteMinder agent that SecureTransport should use when connecting to the SiteMinder Policy Server.	Required
Agent Type	For SiteMinder protocol version 4 the shared secret used to communicate with the SiteMinder Policy Server. For version 5, the path to `SmHost.conf`.	Required
Shared Secret	The password for the SiteMinder agent that SecureTransport uses to connect to the Policy Server.	Required
Maximum Connections	The maximum number of SiteMinder connections that SecureTransport can have open simultaneously. This does not limit the number of users who can log in to the SecureTransport Server using the Site Minder SSO portal.	Required

Name	Description	Required/ optional
Connection Timeout	The amount of time (in seconds) that a SiteMinder connection can be idle before it is closed. The default is 30 seconds. This is independent of user session timeout.	Required
File Storage Root Path	The segment of the absolute URI that is removed before it is submitted to the SiteMinder Policy Server for authorization. If the entire absolute URI is submitted for authorization, type `/` in this field.	Required
SiteMinder Path Prefix	After the File Storage Root Path is removed, but prior to SiteMinder authorization, this entry is prefixed to the absolute URI. For example, if the absolute URI is `/mnt/ab/user1`, the File Storage Root Path is `/mnt/ab`, and the SiteMinder Path Prefix is `/root`; then `/root/user1` is sent to SiteMinder for authorization. If this box is left blank, no prefix is applied to the URI prior to authorization. **Note** When SiteMinder is enabled, all SecureTransport users must have GET access to the path specified in the SiteMinder Path Prefix to successfully log in. If this setting is left blank, then users must have GET access to /. The SiteMinder administrator must set up the SiteMinder Policy Server accordingly.	Optional
Default Home Folder	The absolute URI of the default home folder of the local user. The default home folder is used when a home folder is not supplied by the SiteMinder Policy Server. **Requirements:** • The folder must be created manually on the machine. • On Windows, the folder should not use shared storage. • On Linux, the permissions of the folder must be set to "777" on all nodes in the cluster.	Required

Name	Description	Required/ optional
Default Local User ID	The numeric user ID (UID) of a user that has full read/write access to the directory specified as the File Storage Root Path and its subdirectories. This default is used only if a UID is not supplied by SiteMinder. **Note** On Windows, type the name of the respective virtual user. Windows does not support UIDs.	Required
Default Local Group ID	The numeric group ID (GID) of a user that has full read/write access to the directory specified in the File Storage Root Path and its subdirectories. If no UID or GID are supplied by SiteMinder, these defaults are used for all file operations (including ownership of new files) performed by SecureTransport for users authenticated by SiteMinder.	Required
Explicitly uses SiteMinder Attributes	When selected, SecureTransport uses the values specified in the *User Attribute Names* section. If not selected, and • if the user is assigned to an account template, the User ID, Group ID, and Home Folder are determined from the template. • if the user is not assigned to an account template, the default home folder, local user ID, and local group ID are used. The state of the checkbox has no effect on SiteMinder users mapped as virtual users.	Optional

Name	Description	Required/ optional
Home Folder Attribute	SiteMinder returns information about the user as name=value pairs. The attributes define the name part of the pair which must be added manually. In `Home Folder Attribute`, you should provide the name of the SiteMinder attribute which value is the absolute URI of the user's home folder. **Requirements:** • The folder must be created manually on the machine. • On Windows, the folder should not use shared storage. • On Linux, the permissions of the folder must be set to "777" on all nodes in the cluster. **Note** When changing `Home Folder Attribute`, the Transaction Manager should be restarted in order for the changes to be applied. **Note** If `Home Folder Attribute`, `Group ID Attribute` and `User ID Attribute` are left blank, and the attributes therefore not defined, the default Home Folder, Local User ID, and Local Group ID are used.	Optional
User ID Attribute		Optional
Group ID Attribute		Optional

3. Click **Update SiteMinder Settings**.

   To use one or two more SiteMinder servers for failover, specify server configuration parameters that correspond to the fields described above.

   The names of the parameters for the second server start with `Siteminder.PolicyServers.Second.PolicyServer`.
   The names for the third server start with `Siteminder.PolicyServers.Third.PolicyServer`.
   The final parts of the names are given in the following table:

Field	Sever configuration parameter
Enable SiteMinder Module	`enable`
IP Address	`host`

Field	Sever configuration parameter
Administrator Username	`adminUsername`
Administrator Password	`adminPassword`
Authorization Port	`authorizationPort`
Authentication Port	`authenticationPort`
LDAP User Directory	`ldapUserDirectory`
Maximum Connections	`maxConnections`
Connection Timeout	`timeout`

There are also parameters for `minConnections` and `connectionsStep` which are not set in the *CA SiteMinder Setting* page.

Disable the SecureTransport login

Web clients that have logged on through the SiteMinder SSO portal should not log on again to the SecureTransport login page. Therefore, it is necessary to disable the SecureTransport login page for these clients.

1. Select **Operations > Server Configuration**.

   The *Server Configuration* page is displayed.
2. Search for the `Http.FdxAuthReply` parameter.
3. Change the value to `PREAUTH`.
4. Restart the HTTP server and the TM Server.

Configure client certificate authentication settings

Instead of logging on through the SSO portal, web clients and command line clients can log on to SecureTransport directly and request authentication using a client certificate. SecureTransport then presents the client certificate to SiteMinder for authentication. To configure SecureTransport to perform client certificate authentication using SiteMinder, complete these steps:

1. Import into SecureTransport the trusted Certificate Authority (CA) certificates for the client certificates to be authenticated. Client certificates are authenticated using indirect trust. For instructions on importing trusted CA certificates, refer to *Import a trusted CA certificate* on page

56.

2. Restart the SSH Server.
3. Bounce the SecureTransport Servers.

Integration troubleshooting

Use the recommendations in this topic to troubleshoot problems that you might encountered when integrating SecureTransport with SiteMinder.

SiteMinder troubleshooting tools

One of the first things to determine is if a problem is in the SiteMinder configuration settings. The following SiteMinder tools are helpful in troubleshooting the SiteMinder configuration:

- **SiteMinder Test Tool** – helpful in testing agent, rule, and policy configurations in SiteMinder.
- **SiteMinder Debug Log files** – helpful in troubleshooting SiteMinder Policy Server configuration issues. These debug log files are configured using the Netegrity Policy Server Management Console.

Refer to the SiteMinder documentation for information about these tools.

SecureTransport troubleshooting tools

If problems still exist after troubleshooting with the SiteMinder tools, use the SecureTransport log files to try to ascertain the problem.

To use the SecureTransport log files effectively to troubleshoot SiteMinder integration issues, consider the following recommendations:

- Increase the log level of `com.tumbleweed.st.server.siteminder` log generator in `<FILEDRIVEHOME>/conf/tm-log4j.xml` to `debug`. Increase the log levels of other log generators in this file as well.
- Increase all the log levels in `<FILEDRIVEHOME>/conf/tm-log4j.xml` from `warn` to `debug`.
- Check the `<FILEDRIVEHOME>/var/logs/tm.log` file and the *Server Log* page for any messages when a SiteMinder login fails.

After increasing the logging level settings, restart Transaction Manager to apply the changes.

Authentication 8

The following set of topics provides detailed SecureTransport login setting configuration and authentication information for end-users and administrators and LDAP and SiteMinder configuration information:

- *Single Sign-On (SSO) and Single Logout (SLO)* on page 426 - Describes the SecureTransport Single Sign-On and Single Logout functionality.
- *Single Sign-On (SSO) configuration* on page 427 - Provides configuration prerequisites, an overview of the main configuration files, and describes configuring SSO.
- *Enable Single Sign-On (SSO) for administrators* on page 432 - Describes how to enable SSO for administrators.
- *Enable Single Sign-On (SSO) for end-users* on page 440 - Describes how to enable SSO for end-users.
- *Multiple Identity Provider configuration* on page 451 - Describes configuring multiple Identity Providers.
- *Configure Single Sign-On (SSO) for streaming* on page 454 - Describes configuring SSO for streaming.
- *Configure Single Sign-On (SSO) for clusters* on page 455 - Describes configuring SSO for clusters.
- *SecureTransport as an Identity Provider* on page 455 - Describes using SecureTransport as an Identity Provider.
- *Single Sign-On SSO authentication flows* on page 456 - Describes the SSO and SLO authentication flows.
- *Configure Kerberos as an Identity Provider in SecureTransport* on page 457 - Describes Kerberos SSO authentication with Active Directory.
- *Login settings* on page 465 - Provides configuration information and instructions for enabling or disabling SSO authentication of end users and administrators, enabling or disabling certificate authentication and client certificate authentication for end users and administrators, enable or disable dual authentication, and set LDAP and SiteMinder authentication levels.
- *SiteMinder integration configuration* on page 470 - Provides how-to instructions for configuring the SiteMinder integration.
- *LDAP integration* on page 475 - Describes the SecureTransport LDAP integration.
- *LDAP connections, binds, and searches* on page 476 - Describes the configuration of LDAP connections, binds, and searches.
- *LDAP logins* on page 476 - Describes how LDAP logins are used and searched when LDAP is enabled.
- *LDAP domains* on page 478 - Describes the management and configuration of LDAP domains.

- *LDAP home folders* on page 496 - Describes LDAP homes folders and provides LDAP home folder configuration instructions.
- *LDAP user type ranges* on page 498 - Describes the configuration of LDAP user type ranges on UNIX systems.

Single Sign-On (SSO) and Single Logout (SLO)

Single Sign-On (SSO) is a user authentication process that authenticates the user for all the applications they have been given rights to and eliminates further prompts when they switch applications during a particular session. Essentially, it removes the need for your users to log into multiple applications in a particular browser session. Once they log into one system, it exchanges authentication data with another service you have SSO set up with and automatically logs the user in.

SecureTransport supports signed authentication response assertions and this is transparent to SecureTransport as long as correct configuration is supplied. A signed response proves it is sent from an intended Identity Provider. Adding an extra security layer by disabling plain SAML connections is highly recommended, and it ensures confidentiality. SAML profiles are configured by means of SAML bindings.

The Single Logout (SLO) profile enables a user to log out of all participating sites in a created session nearly simultaneously. The user may log out globally from any site, whether Service Provider (SP) or Identity Provider (IdP), as determined by respective Web applications. The associated IdP deployment handles all logout requests and responses for participating sites.

SecureTransport implements:

- Single Sign-On (SSO)
- Single Logout (SLO)

Supported protocols:

- SAML 2.0 for Administrators and End-users
- Kerberos 5 for End-users

Note SSO authentication using SAML 2.0 and Kerberos is only applicable for the HTTP protocol.

Note When SecureTransport is behind a reverse proxy/load balancer, the HOST header needs to be modified to match the hostname of the proxy, not the host where the request is proxied to. For Apache this is achieved by setting the `ProxyPreserveHost` property to **On**. For additional information, refer to https://httpd.apache.org/docs/2.4/mod/mod_proxy.html#proxypreservehost.

It also supports user attribute mapping and authentication decisions on attribute maps.

The SSO authentication process is owned and managed by an Identity Provider (IdP). The IdP provides a token authentication object consisting of a user name and a map of attributes. The SecureTransport filters and re-maps the attributes and prepares them to be consumed by SecureTransport.

Single Sign-On (SSO) configuration

Before configuring SecureTransport to use SSO, install and configure your selected *Single Sign-On (SSO)* on page 39.

Note In a SecureTransport deployment, consisting of both backend and edge nodes, SSO must be configured and enabled on all nodes even if users are only logging in through the Edge. On the Edge, you must enable SSO for end-users.

SecureTransport Single Sign-On (SSO) configuration prerequisites

1. Navigate to **Operations > Server Configuration**.
2. Make sure the SameSite policy is set to `Lax` for both the Administration Tool server (`Admin.Security.SameSite`) and the HTTP server (in the server settings, under HTTP Security).
3. On the *Server Configuration* page, click on **Configuration Files**.
4. Click on **SSO Configuration Files** to download them as a ZIP file (`SSO-configuration-export.zip`).
5. Unzip the `SSO-configuration-export.zip` file.

The default SSO configuration files are:

- `sso-admin.xml` – Configure the SSO for administrator component.
- `sso-enduser.xml` – Configure the SSO for end users.
- `krb5-login.conf` – A Kerberos configuration file.
- `sample-kerberos.keytab` – A Kerberos `.keytab` file.

Single Sign-On (SSO) configuration files overview

The main SSO configuration files (`sso-admin.xml` for administrators and `sso-enduser.xml` for end-users) are considered mandatory for configuring the SSO functionality in SecureTransport for the respective components.

For the sample SSO configuration files, refer to *SSO configuration file for administrators* on page 435 and *SSO configuration file for end-users* on page 445.

The main SSO configuration file contains the following elements:

- `<SSOConfiguration>` element – This is the root element of the configuration descriptor. This section contains one `<CertificateValidation>` element (optional), one `<ServiceProvider>` element (required) and one `<IdentityProviders>` element (required).

  Note You can specify only one `<ServiceProvider>` element.

- `<CertificateValidation>` element – Describes the certificate validation. Configures certificate validation. Validates the Service Provider and Identity Providers certificates specified in its configuration. Validation happens at start-up and at regular intervals. This element is optional. Possible attributes for this element:
  - `trustStoreInitializer` – Set `com.axway.st.server.sso.impl.TrustStoreInitializer` value for `trustStoreInitializer` in order to use SecureTransport trust store.
  - `delayBetweenValidations` – Defines at which interval certificates validation occurs, in hours. Default value is 3 hours.
- `<ServiceProvider>` element – Configures the Service Provider. This element is required. Main attributes for this element:
  - `entityId` – Sets the unique identifier of the Service Provider. This identifier is sent to the Identity Provider so it can know who is requesting an authentication or a logout. This identifier is used by the Identity Provider to differentiate what Service Provider is requesting an authentication or a logout.
  - `filteredUri` – Specifies the URI of the authentication process entry point. The value must be `/*` for both administrators and end-users.
  - `logoutUri` – Specifies the URI which triggers logout process. The value must be `/logout` for both administrators and end-users.
  - `keystoreInitializer` – Configures key store to use. That key store keeps key-pairs taking part in authentication process. Set `com.axway.st.server.sso.impl.KeyStoreInitializer` as value in order to use the SecureTransport local key store.
  - `keyAlias` – Specifies key alias of the private key used to decrypt SAML messages and assertions and to sign SAML messages and assertions.
  - `sessionIdCookieName` – Sets the name of the cookie to store the SSO session identifier if sessions are managed by the SSO module.
- `<AssertionConsumerService>` element – Specifies an entry point for receiving SAML Assertions from the Identity Provider.
- `<SingleLogoutService>` element - Specifies the Identity Provider URL where the logout responses are sent.

  Note The recommended values for both `<AssertionConsumerService>` and `<SingleLogoutService>` are listed in *SSO configuration file for administrators* on page 435 and *SSO configuration file for end-users* on page

445.

- `<Features>` element - The SSO agent can be fine-tuned by using an extra configuration features. In most cases, the values of these features don't have to be modified. The recommended features are:
  - `<Feature key="secure-cookie" value="true" />` - Configures the session cookie whether to be set with Secure flag. Recommended value is true.
  - `<Feature key="uid-generator" value="com.axway.st.server.sso.impl.UIDGenerator" />` - Type of unique identifier generator to use to assign ids to SAML messages. The value must be `com.axway.st.server.sso.impl.UIDGenerator`.
    - `<IdentityProviderResolution>` - For more information, refer to *Multiple Identity Provider configuration* on page 451.
    - `<TenantResolution>` - For more information, refer to *Multiple Identity Provider configuration* on page 451.
- `<IdentityProviders>` element - Identity provider definitions. Configures various aspects of interaction with Identity Providers. This element is required. The main attributes for this element are:
  - `entityId` – Sets the unique identifier of the Service Provider. This identifier is sent to the Identity Provider so it can know who is requesting an authentication or a logout. For SAML-based Identity provider add here `<EntityDescriptor>` element `entityID` attribute value, from the Identity provider metadata file.
  - `metadataUrl` – Specify the relative location of the Identity provider metadata file. Use only for SAML-based Identity provider.
  - `configurationUrl` – Specify the configuration file absolute path. Use only for Kerberos-based Identity provider.
  - `verifyAssertionExpiration` - Turn on/off verification of the validity period of assertions. Consider to set to false if Service Provider and Identity Provider times are not synchronized. Default value is true.
  - `sign` - If set to true, all SAML messages and their assertions sent by the Service Provider will be signed. There are a couple of features (see below) for fine-grained control of signing. Optional - if not present, default value is false.
  - `userNameAttribute` - Sets the name of the Identity Provider attribute that provides the user name.
- `<Mappings>` element:
  - `<FilterMapping>` - This mapping creates output attributes when a filter matches the input attributes from the Identity Provider. For more information about the Filter Mapping syntax, refer to *SSO filter mapping* on page 443.
  - `<RenameMapping>` - With this mapping, you can rename an attribute from the Identity Provider, keeping its value. For more information, refer to *Accessing Single Sign-On (SSO) attributes* on page 430.

- `<Features>` element - Features control specific behavior of SAML message processing. The recommended features are:
  - `<Feature key="saml-allow-http-connection" value="false"/>` - Allows interaction with the IdP by plain HTTP. Default value is false.
  - `<Feature key="saml-allow-unsigned-assertion" value="false"/>` - Allows unsigned assertions in messages received from the Identity Provider. Default value is false.
  - `<Feature key="saml-verify-metadata-signature" value="false"/>` - Enable or disable the signature verification of the metadata file and the certification path of the certificate used to sign. Set to false if metadata file is not signed. Default value is true.
  - `<Feature key="saml-sign-authnrequest" value="true"/>` - Enable or disable signing of Authentication Request messages. Presence of this feature and its value overrides the meaning of the sign attribute of `IdentityProvider` element.
  - `<Feature key="saml-sign-logoutrequest" value="true"/>` - Enable or disable signing of Logout Request messages. Presence of this feature and its value overrides the meaning of the sign attribute of `IdentityProvider` element.
  - `<Feature key="saml-sign-logoutresponse" value="true"/>` - Enable or disable signing of Logout Response messages. Presence of this feature and its value overrides the meaning of sign attribute of `IdentityProvider` element above.
  - `<Feature key="saml-allow-unsigned-assertion" value="false"/>` - Allows unsigned assertions in messages received from the Identity Provider. Default value is false.
  - `<Feature key="saml-response-binding" value="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect"/>` - Sets the default response binding value to be `urn:oasis:names:tc:SAML:2.0:bindings:HTTP-Redirect`.

Note SecureTransport supports signature and decryption of SAML requests. Adding an extra security layer is highly recommended. Enable SSL by setting `<Feature key="saml-allow-http-connection" value="false"/>` in the `sso-admin.xml` and `sso-enduser.xml` SSO configuration files.

Accessing Single Sign-On (SSO) attributes

When your SSO login is successful, your Identity Provider will forward a set of user attributes to the requested Service Provider (SecureTransport). The SSO attributes are used by SecureTransport to configure user account instances, agent sessions, and advanced routing attributes.

Note Accessing Single Sign-On (SSO) attributes is not possible when using SSO with Kerberos authentication protocol. It is possible only with SAML.

There are several system attributes that are considered important for SecureTransport and there are custom mappings for the attributes:

Attribute name (Identity Provider)	Attribute name in SecureTransport*
email	fdxEmail
uid	fdxUid
gid	fdxGid
homeDir	fdxHomeDir

*The expected SecureTransport attribute name in order for the attribute to be mapped correctly.

For more information about the SSO attributes, refer to *SSO filter mapping* on page 443.

Access the system attributes

1. Access the SSO system attributes in an Advanced Routing environment:

Attribute name	Expression syntax
email	${sso.email}
uid	${sso.uid}
gid	${sso.gid}
homeDir	${sso.homeDir}
tenant	${sso.tenant}
idpId	${sso.idpId}
userName	${sso.userName}

2. Access the SSO system attributes from a Session:

Attribute name	Expression syntax
email	${sess.STSESSION_SSO.email}
uid	${sess.STSESSION_SSO.uid}

Attribute name	Expression syntax
gid	${sess.STSESSION_SSO.gid}
homeDir	${sess.STSESSION_SSO.homeDir}
tenant	${sess.STSESSION_SSO.tenant}
idpId	${sess.STSESSION_SSO.idpId}
userName	${sess.STSESSION_SSO.userName}

3. Access the custom SSO attributes. If we have a custom attribute with name `customAttribute`, you can access the first value in the following manner:
   a. From a Session:

      ```
      ${sess.STSESSION_SSO.customAttribute[0]}
      ```

   b. From an Advanced Routing environment:

      ```
      ${sso.attributes['customAttribute'][0]}
      ```

Enable Single Sign-On (SSO) for administrators

The following steps are the general configuration steps to enable SSO functionality for administrators in SecureTransport.

1. Navigate to **Authentication > Login Settings**.
2. In *Administrator login options* pane, select **Required** for *SSO*.
3. Click **Save**.

Configure Single Sign-On (SSO) for administrators

Before configuring SSO for administrators, refer to *SecureTransport Single Sign-On (SSO) configuration prerequisites* on page 427.

In order to configure SSO functionality for administrators, you need to update the `sso-admin.xml` file and remember that SSO authenticated administrators are only mapped to existing SecureTransport administrator accounts. For additional information, refer to *Add an administrator account* on page 702.

Note Do not rename the `sso-admin.xml` configuration file.

Note SecureTransport only supports SAML-based Identity providers for SSO for administrators.

Note The following configuration steps describe the setup of a single Identity provider. For multiple Identity Provider configuration, refer to *Multiple Identity Provider configuration* on page 451.

Note Before configuring SSO for Administrators using a SAML-based Identity Provider, make sure that you configured it properly.

Configure SSO for administrators using SAML-based Identity Providers:

1. Download the SAML-based Identity Provider metadata file from your Identity Provider instance.

   Note Do not modify the SAML-based Identity Provider metadata file.

2. Open the `sso-admin.xml` file. The following changes are required:
   - In the `SamlIdentityProvider` element, change the following attribute values:
     - `metadataUrl` to be `./(name of the SAML-based Identity provider metadata file)`
     - `entityId` - add the `<EntityDescriptor>` element `entityID` attribute value, from the SAML-based Identity Provider metadata file.
   - `Mappings` element: Both `<FilterMapping>` and `<RenameMapping>` elements are not applicable for administrators.
   - `Features` element: The recommended features are listed in *SSO configuration file for administrators* on page 435.
3. Save the `sso-admin.xml` file.
4. Zip the `sso-admin.xml` file and the SAML-based Identity Provider metadata file from Step 1.

   Note Do not put the configuration files in a sub-directory inside the ZIP file.

5. Navigate to **Operations > Server Configuration**. Click on **Configuration Files**.
6. Select the **Browse** button for SSO Configuration Files. Choose the ZIP file containing the `sso-admin.xml` file and the SAML-based Identity Provider metadata file.
7. Click on the checkbox for **SSO Configuration Files**.
8. Click **Upload**.
9. Restart the admin service.

Note If, for some reasons after importing SSO configuration files and enabling SSO for administrators, you are still redirected to the default Administrator login page, perhaps there is some misconfiguration. To resolve this situation you can use SecureTransport as an Identity provider to login with the local stored credentials and troubleshoot. For more information, refer to *SecureTransport as an Identity Provider* on page 455.

Note In both Standard Clusters and Enterprise Clusters, after successfully importing the SSO configuration files, they will be automatically redistributed across all nodes in the cluster. Restart of admin service is required on all nodes.

Note In cluster environment SecureTransport will always redirect to the node that is configured in the Service Provider, even if the request came from a different node.

Note Due to the limitation of having only one Service Provider entity ID for the `sso-admin.xml` configuration file and the fact that configuration files are synced between the cluster nodes, all administrators will have the same service provider configuration. Since the IdP cannot differentiate which request is coming from which node, it will always return the user to the assertion consumer service configured on the IdP. This could be worked around by having a separate IdP for each cluster node and the user could select the node they want to login to by choosing the dedicated IdP. For more information about how to configure multiple Identity Provider in SecureTransport, refer to *Multiple Identity Provider configuration* on page 451.

Single Sign-On (SSO) administrators configuration

For more information about how to setup the SecureTransport administrators to use SSO, refer to *Manage administrator accounts* on page 700.

SSO configuration file for administrators

This topic contains a code snippet of the default *sso-admin.xml* file. To access and download the file from the Administration Tool, follow these steps:

1. Go to **Operations > Server Configuration**.
2. Click the **Configuration files** button.
3. Click the plus sign **(+)** to next to **SSO Configuration Files** to see the available SSO configuration files.
4. Click on **sso-admin.xml** to download the file.

sso-admin.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- This is a sample file for SSO configuration for Admin component. -->
<SSOConfiguration>

    <!--
         Configures certificate validation. Validates the Service Provider and Identity
Providers certificates specified
         in its configuration. Validation happens at start-up and at regular intervals.
Optional.
    -->
    <!--
         Attributes:
         1) trustStoreInitializer - Set com.axway.st.server.sso.impl.TrustStoreInitializer
value for trustStoreInitializer
                                    in order to use SecureTransport trust store. Recommended
value: com.axway.st.server.sso.impl.TrustStoreInitializer
         2) delayBetweenValidations - Defines at which interval certificates validation
occurs, in hours. Default value is 3 hours.
    -->
    <CertificateValidation
            trustStoreInitializer="com.axway.st.server.sso.impl.TrustStoreInitializer"
            delayBetweenValidations="3">
    </CertificateValidation>

    <!-- Configures the service provider. -->
    <!--
    Main attributes:
        entityId - Sets the unique identifier of the service provider. This identifier is
sent to the Identity Provider so it can know who is
                   requesting an authentication or a logout. This identifier is used by the
Identity Provider to differentiate what Service
                   Provider is requesting an authentication or a logout.
        filteredUri - Specifies the URI of the authentication process entry point. The value
must be /*
        logoutUri - Specifies the URI which triggers logout process. The value must be
/logout.
        keystoreInitializer - Configures key store to use. That key store keeps key-pairs
taking part in authentication process.
                              Set com.axway.st.server.sso.impl.KeyStoreInitializer value in
order to use SecureTransport local key store.
        keyAlias - Specifies key alias of the private key used to decrypt SAML messages and
assertions and to sign SAML messages and assertions.
```

```
        sessionIdCookieName - Sets the name of the cookie to store the SSO session identifier
if sessions are managed by the SSO module.
     -->
    <ServiceProvider
            entityId="st.sso.admin"
            filteredUri="/*"
            logoutUri="/logout"
            keystoreInitializer="com.axway.st.server.sso.impl.KeyStoreInitializer"
            keyAlias="ssokey"
            sessionIdCookieName="STAdminSsoCookie"
            useAppSessions="false"
            >

        <!-- Specifies an entry points for receiving SAML Assertions from the Identity
Provider. The below tags are recommended. -->
        <AssertionConsumerService binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
POST" location="/saml2/sso/post/j_security_check"/>
        <AssertionConsumerService binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" location="/saml2/sso/redirect/j_security_check"/>
        <AssertionConsumerService binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
PAOS" location="/saml2/sso/paos/j_security_check"/>
        <SingleLogoutService binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
POST" location="/saml2/slo/post/j_security_check"/>
        <SingleLogoutService binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" location="/saml2/slo/redirect/j_security_check"/>

        <!-- Features tag is optional - Here are the default values -->
        <Features>

            <!-- Configures the session cookie whether to be set with Secure flag.
Recommended value: true. -->
            <Feature key="secure-cookie" value="true" />

            <!--
                 Type of unique identifier generator to use to assign ids to SAML messages.
                 The value must be com.axway.st.server.sso.impl.UIDGenerator
            -->
            <Feature key="uid-generator" value="com.axway.st.server.sso.impl.UIDGenerator" />
        </Features>

        <!--
            Identity Provider resolution provides support for choosing the right Identity
Provider
            based on configuration and run-time metadata.
            If such resolution is not present, the first Identity Provider is selected among
ones specified
            under IdentityProviders element below.
            The supported ways to do that are by:
            1) Query parameter provided by a user request (see the example below).
            2) Header value provided by a user request. An example follow:
                <Header name="idp_id">
                    <Mapping value="keycloackIdp" entityId="https://st.keycloack.axway.int/"
/>
                    <Mapping value="shibbolethIdp"
entityId="https://st.shibboleth.axway.int/" />
                </Header>
                In the example above if a user authentication request has header with name
'idp_id' and
                corresponding value equals to 'keycloackIdp', then Identity Provider with
entityId equals to
                https://st.keycloack.axway.int/ will be chosen to authenticate the user
agent.
          Note: Only one of these way can be done.
```

```
        -->
        <IdentityProviderResolution>
            <!--
                 Identity provider mapping using a query parameter. The name of query
parameter resolution will be
                 searched for in request parameters during runtime and its value should match
to the value attribute of a
                 Mapping element. If both query parameter name and value match, then
corresponding entityId is used to
                 select Identity Provider.
                 Examples:
                         1) https://localhost/?idp_id=keycloackIdp in the below case will
match the
                            Identity provider with entityId=https://st.keycloack.axway.int/
                         2) https://localhost/?idp_id=shibbolethIdp in the below case will
match the
                            Identity provider with entityId=https://st.shibboleth.axway.int/
             -->
            <QueryParameter name="idp_id">
                <!-- Note: The name of the query parameter/header should match the value of
the ST configuration option
                          LoginSettings.Admin.SSO.idpResolverKey. -->
                <!-- Note: Ensure the name of the query parameter/header to be different than
ST configuration option
                          LoginSettings.Admin.SSO.localIdpId in order to be able to
configure selection of ST as
                          local authentication provider and SSO Identity Providers. -->
                <Mapping value="keycloackIdp" entityId="https://st.keycloack.axway.int/" />
                <Mapping value="shibbolethIdp" entityId="https://st.shibboleth.axway.int/" />
            </QueryParameter>
        </IdentityProviderResolution>

        <!--
            This element is optional.
            Tenant resolution provides support for choosing the right Identity Provider based
on
            configuration and run-time metadata.
            If both tenant resolution and identity provider resolutions are present, then
tenant resolution takes precedence.
            Tenants are defined inside Identity Providers so resolving the IdP in turn will
resolve a tenant.
            The supported ways to do that are by:
            1) QueryParameter
                <QueryParameter name="idp_id">
                    <Mapping tenant="Axway" entityId="https://st.keycloack.axway.int/" />
                    <Mapping tenant="Sopra" entityId="https://st.shibboleth.axway.int/" />
                </QueryParameter>
            2) Header (example below)
            Notes:
                  1) Only one of these way can be done.
                  2) Header evaluation takes precedence on query parameter one.
                  3) If mapping is not present, IdentityProviderResolution is used.
                     If IdentityProviderResolution is not present first listed IdP is used.
        -->
        <TenantResolution>
            <!-- Note: The name of the query parameter/header should match the value of the
ST configuration option
                          LoginSettings.Admin.SSO.idpResolverKey. -->
            <!-- Note: Ensure the name of the query parameter/header to be different than ST
configuration option
                          LoginSettings.Admin.SSO.localIdpId in order to be able to
configure selection of ST as
                          local authentication provider and SSO Identity Providers. -->
```

```
            <Header name="idp_id">
                <Mapping tenant="Axway" entityId="https://st.keycloack.axway.int/" />
                <Mapping tenant="Sopra" entityId="https://st.shibboleth.axway.int/" />
            </Header>
        </TenantResolution>
    </ServiceProvider>

    <!-- Identity provider definitions. Configures various aspects of interaction with
identity providers. -->
    <IdentityProviders>
        <!-- A SAML sample IdP definition. -->
        <!--
        Main attributes:
            entityId - Sets the unique identifier of the service provider. This identifier is
sent to the Identity Provider
                       so it can know who is requesting an authentication or a logout.
                       Add here EntityDescriptor entityID value, from the idpMetadata.xml
            metadataUrl - Specify the relative location of the metadata file.
                       Specifies a relative location of the metadata file to sso-admin.xml
file.
                       NOTE: ST does not support the metadata URL to be a HTTP site.
            verifyAssertionExpiration - Turn on/off verification of the validity period of
assertions. Consider to set to false if
                                        service provider and identity provider times are not
synchronized. Default: true.
            sign - If set to true, all SAML messages and their assertions sent by the service
provider will be signed.
                    There are a couple of features (see below) for fine-grained control of
signing. Optional - if not present, default value is false.
            userNameAttribute - Sets the name of the identity provider attribute that
provides the user name.
         -->

         <!-- Sample Keycloak Identity provider definition. -->
        <SamlIdentityProvider
                entityId="https://st.keycloack.axway.int/"
                metadataUrl="./keycloack-idp-metadata.xml"
                verifyAssertionExpiration="false"
                sign="true">

            <!-- Mappings tag is optional -->
            <Mappings>
                <!-- NOTE: SecureTransport does not support the attribute mapping for Admin
component. -->
            </Mappings>

            <!-- Features control specific behavior of SAML message processing. -->
            <Features>
                <!-- Allows interaction with the IdP by plain HTTP. Default: false. -->
                <Feature key="saml-allow-http-connection" value="false"/>

                <!-- Allows unsigned assertions in messages received from the Identity
Provider. Default: false. -->
                <Feature key="saml-allow-unsigned-assertion" value="false"/>

                <!--
                    Enable or disable the signature verification of the metadata file and the
certification
                    path of the certificate used to sign. Set to false if metadata file is
not signed.
                    Default: true.
                -->
                <Feature key="saml-verify-metadata-signature" value="false"/>
```

```
                <!--
                    Enable or disable signing of Authentication Request messages. Presence of
this feature and its
                    value overrides the meaning of sign attribute of IdentityProvider element
above.
                -->
                <Feature key="saml-sign-authnrequest" value="true"/>

                <!--
                    Enable or disable signing of Logout Request messages. Presence of this
feature and its value
                    overrides the meaning of sign attribute of IdentityProvider element
above.
                 -->
                <Feature key="saml-sign-logoutrequest" value="true"/>

                <!--
                    Enable or disable signing of Logout Response messages. Presence of this
feature and its value
                    overrides the meaning of sign attribute of IdentityProvider element
above.
                -->
                <Feature key="saml-sign-logoutresponse" value="true"/>

            </Features>
        </SamlIdentityProvider>

        <!-- Sample Shibboleth Identity provider definition. -->
        <SamlIdentityProvider
                entityId="https://st.shibboleth.axway.int/"
                metadataUrl="./shibboleth-idp-metadata.xml"
                verifyAssertionExpiration="false"
                userNameAttribute="urn:oid:0.9.2342.19200300.100.1.1"
                sign="true" >

            <!-- Mappings tag is optional -->
            <Mappings>
                <!-- NOTE: SecureTransport does not support the attribute mapping for Admin
component. -->
            </Mappings>

            <!-- Features control specific behaviour of SAML message processing. -->
            <Features>
                <!-- Allows interaction with the IdP by plain HTTP. Default: false. -->
                <Feature key="saml-allow-http-connection" value="false"/>

                <!-- Allows unsigned assertions in messages received from the Identity
Provider. Default: false. -->
                <Feature key="saml-allow-unsigned-assertion" value="false"/>

                <!--
                    Enable or disable the signature verification of the metadata file and the
certification
                    path of the certificate used to sign. Set to false if metadata file is
not signed.
                    Default: true.
                -->
                <Feature key="saml-verify-metadata-signature" value="false"/>

                <!--
                    Enable or disable signing of Authentication Request messages. Presence of
this feature and its
```

```
                    value overrides the meaning of sign attribute of IdentityProvider element
above.
                -->
                <Feature key="saml-sign-authnrequest" value="true"/>

                <!--
                    Enable or disable signing of Logout Request messages. Presence of this
feature and its value
                    overrides the meaning of sign attribute of IdentityProvider element
above.
                 -->
                <Feature key="saml-sign-logoutrequest" value="true"/>

                <!--
                    Enable or disable signing of Logout Response messages. Presence of this
feature and its value
                    overrides the meaning of sign attribute of IdentityProvider element
above.
                -->
                <Feature key="saml-sign-logoutresponse" value="true"/>

            </Features>
        </SamlIdentityProvider>

    </IdentityProviders>
</SSOConfiguration>
```

Enable Single Sign-On (SSO) for end-users

The following steps are the general configuration steps to enable SSO functionality for end-users in SecureTransport.

1. Navigate to **Authentication > Login Settings**.
2. In *End-users login options* pane, select **Required** for *SSO*.
3. Click **Save**.

Note When SSO for end-users is enabled, the following configuration options will be updated automatically:

1. `Http.FdxAuthReply` with value **PREAUTH**.
2. `Http.AllowedAuthenticationParameters` with value **SAMLResponse;RelayState**.
3. `AllowedAuthenticationParametersMaxSize` with value **32768**.

Configure Single Sign-On (SSO) for end-users

The following configuration steps describe the setup of the single Identity provider. For multiple Identity Provider configuration, refer to *Multiple Identity Provider configuration* on page 451.

Before configuring SSO for end-users, ensure that all *SecureTransport Single Sign-On (SSO) configuration prerequisites* on page 427 are met and the Identity provider is configured properly. For a full list of supported Identity providers, refer to *Single Sign-On (SSO)* on page 39.

In order to configure SSO functionality for end-users, you need to update the `sso-enduser.xml` file and remember that SSO authenticated users are only mapped to existing SecureTransport user accounts or account templates with user classes.

Note Do not rename the `sso-enduser.xml` configuration file.

To configure SSO for End-users using SAML-based Identity provider:

1. Download the SAML-based Identity provider metadata file from your Identity Provider instance.

   Note Do not modify the SAML-based Identity Provider metadata file.

2. Open the `sso-enduser.xml` file. The following changes are required:
   - In the `<SamlIdentityProvider>` element, change the following attribute values:
     - `metadataUrl` to be `./(name of the SAML-based Identity provider metadata file)`
     - `entityId` - add the `<EntityDescriptor>` element `entityID` attribute value, from the SAML-based Identity Provider metadata file.
   - `<Mappings>` element:
     - `FilterMapping` - For information on filter mapping, refer to the *SSO filter mapping* on page 443.
     - `RenameMapping` – For mapping custom attributes from an Identity Provider, refer to *Accessing Single Sign-On (SSO) attributes* on page 430.
   - `<Features>` element: The recommended features are listed in *SSO configuration file for end-users* on page 445.
3. Save the `sso-enduser.xml` file.
4. Zip the `sso-enduser.xml` and the SAML-based Identity Provider metadata file from Step 1.

   Note Do not put the configuration files in a sub-directory inside the ZIP file.

5. Navigate to **Operations > Server Configuration**. Click on **Configuration Files**.
6. Select the **Browse** button for SSO Configuration Files. Choose the ZIP file containing the `sso-enduser.xml` file and the SAML-based Identity Provider metadata file.
7. Click on the checkbox for **SSO Configuration Files**.
8. Click **Upload**.
9. Restart the Transaction Manager service and the HTTP service.

Configure SSO for end-users using Kerberos:

1. Configure Kerberos as an Identity provider. For the configuration with Kerberos as an Identity Provider, refer to *Configure Kerberos as an Identity Provider in SecureTransport* on page 457.
2. Open the `sso-enduser.xml` file. The following changes are required:

3. In `KerberosdentityProvider` element change the following attribute values:
   - `configurationUrl` to be the absolute path to the Kerberos `.conf` file.

     Note The Kerberos `.conf` file and `.keytab` file should be added to the SSO configuration ZIP file.

   - `entityId` - add the `entityID` attribute value
   - `<Mappings>` element:
     - `FilterMapping` – For information on filter mapping, refer to the *SSO filter mapping* on page 443.
     - `RenameMapping` – For mapping custom attributes from an Identity Provider, refer to *Accessing Single Sign-On (SSO) attributes* on page 430.
   - `<Features>` element: The recommended features are listed in *SSO configuration file for end-users* on page 445.
4. Save the `sso-enduser.xml` file.
5. Zip the `sso-enduser.xml` and all additional files (the Kerberos configuration file and the `.keytab` file).

   Note Do not put the configuration files in a sub-directory inside the ZIP file.
6. Navigate to **Operations > Server Configuration**. Click on **Configuration Files**.
7. Select the **Browse** button for SSO Configuration Files. Choose the ZIP file containing the `sso-enduser.xml` and the Identity provider metadata file.
8. Click on the checkbox for **SSO Configuration Files**.
9. Click **Upload**.
10. Restart the Transaction Manager service and the HTTP service.

Note If, for some reasons after importing SSO configuration files and enable SSO for end-users you still redirect to default SecureTransport end-users login page, perhaps there is some misconfiguration. To resolve this situation you can use SecureTransport as an Identity provider to login with the local stored credentials and troubleshoot. For more information, refer to *SecureTransport as an Identity Provider* on page 455.

Note In both Standard Cluster and Enterprise Cluster, after successfully importing the SSO Configuration files, they will be automatically redistributed across all nodes in the cluster. Restart operation is required of Transaction Manager service on all nodes.

Note SSO for end-users can be configured using only `sso-enduser.xml` file only for backend instance.

Note Due to the limitation of having only one Service Provider entity ID for the `sso-enduser.xml` configuration file and the fact that configuration files are synced between the cluster nodes, all end-users will have the same service provider configuration. Since the IdP cannot differentiate which request is coming from which node, it will always return the user to the assertion consumer service configured on the IdP. This could be worked around by having a separate IdP for each cluster node and the user could select the node they want to login to by choosing the dedicated IdP. For more information about how to configure multiple Identity Provider in SecureTransport, refer to *Multiple Identity Provider configuration* on page 451.

Single Sign-On (SSO) account configuration

For more information about how to configure SecureTransport accounts with SSO, refer to *Accounts* on page 500 and *Advanced account administration* on page 686.

SSO filter mapping

An Identity Provider can return attributes attached to the authenticated user. A mapping using this attributes may be executed by the SSO agent before they are transmitted to the application.

Two kinds of mapping are supported: *Rename* mapping and *Filter* mapping.

Rename mapping

With this mapping, you can rename an attribute from the Identity Provider, keeping its value.

Filter mapping

This mapping creates output attributes when a filter matches the input attributes from the Identity Provider.

Note Currently, the output attribute value is fixed. It cannot take the value from an input attribute.

Filter syntax

Only a subset of the full syntax is supported as described below. A filter consists of one or more criteria. If more than one criterion exist in one filter definition, they can be concatenated by logical operators.

Criteria

The criteria have to be put in parentheses. A criteria can only be an equality.

Example:

```
(givenName=Sandra)
```

Operators

The logical operators are always placed in front of the criteria. The whole term have to be put in parentheses.

AND Operator

`(&(criteria1)(criteria2))` means: criteria1 AND criteria2

With more than two criteria: `(&(criteria1)(criteria2)(criteria3)(criteria n)`

OR Operator

`(|(criteria1)(criteria2))` means: criteria1 OR criteria2

With more than two criteria: `(|(criteria1)(criteria2)(criteria3)(criteria n))`

NOT Operator

`(!(criteria1)` means NOT criteria1

Nested Operators

`(&(|(criteria1) (criteria2))(|(criteria3) (criteria4)))` means : (criteria1 OR criteria2) AND (criteria3 OR criteria4)

Examples

Rename the *user* attribute to *username*:

```
<Mappings>
    <RenameMapping source="user" target="username"/>
</Mappings>
```

Add two attributes when the name attribute from the Identity Provider is set to Bob:

```
<Mappings>
    <FilterMapping>
        <Filter>(name=Bob)</Filter>
        <OutputAttribute name="role">SPRole</OutputAttribute>
        <OutputAttribute name="user">Bob</OutputAttribute>
    </FilterMapping>
</Mappings>
```

SSO configuration file for end-users

This topic contains a code snippet of the default *sso-enduser.xml* file. To access and download the file from the Administration Tool, follow these steps:

1. In the Administration Tool, go to **Operations > Server Configuration**.
2. Click the **Configuration files** button.
3. Click the plus sign **(+)** next to **SSO Configuration Files** to see the available SSO configuration files.
4. Click on **sso-enduser.xm** to download the file.

sso-enduser.xml

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<!-- This is a sample file for SSO configuration for End-user component. -->
<SSOConfiguration>

    <!--
         Configures certificate validation. Validates the Service Provider and Identity
Providers certificates specified
         in its configuration. Validation happens at start-up and at regular intervals.
Optional.
    -->
    <!--
         Attributes:
         1) trustStoreInitializer - Set com.axway.st.server.sso.impl.TrustStoreInitializer
value for trustStoreInitializer
                                    in order to use SecureTransport trust store. Recommended
value: com.axway.st.server.sso.impl.TrustStoreInitializer
         2) delayBetweenValidations - Defines at which interval certificates validation
occurs, in hours. Default value is 3 hours.
    -->
    <CertificateValidation
            trustStoreInitializer="com.axway.st.server.sso.impl.TrustStoreInitializer"
            delayBetweenValidations="3">
    </CertificateValidation>

    <!-- Configures the service provider. -->
    <!--
    Main attributes:
        entityId - Sets the unique identifier of the service provider. This identifier is
sent to the Identity Provider so it can know who is
                   requesting an authentication or a logout. This identifier is used by the
Identity Provider to differentiate what Service
                   Provider is requesting an authentication or a logout.
        filteredUri - Specifies the URI of the authentication process entry point. The value
must be /*
        logoutUri - Specifies the URI which triggers logout process. The value must be
/logout.
        keystoreInitializer - Configures key store to use. That key store keeps key-pairs
taking part in authentication process.
                              Set com.axway.st.server.sso.impl.KeyStoreInitializer value in
order to use SecureTransport local key store.
        keyAlias - Specifies key alias of the private key used to decrypt SAML messages and
assertions and to sign SAML messages and assertions.
```

```
        sessionIdCookieName - Sets the name of the cookie to store the SSO session identifier
if sessions are managed by the SSO module.
     -->
    <ServiceProvider
            entityId="st.sso.enduser"
            filteredUri="/*"
            logoutUri="/logout"
            keystoreInitializer="com.axway.st.server.sso.impl.KeyStoreInitializer"
            keyAlias="ssokey"
            sessionIdCookieName="STEndUserSsoCookie"
            useAppSessions="false"
            >

        <!-- Specifies an entry points for receiving SAML Assertions from the Identity
Provider. The below tags are recommended. -->
        <AssertionConsumerService binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
POST" location="/saml2/sso/post"/>
        <AssertionConsumerService binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" location="/saml2/sso/redirect"/>
        <AssertionConsumerService
binding="urn:oasis:names:tc:SAML:2.0:bindings:PAOS" location="/saml2/sso/paos"/>
        <SingleLogoutService binding="urn:oasis:names:tc:SAML:2.0:bindings:HTTP-
Redirect" location="/saml2/slo/redirect"/>

        <!-- Features tag is optional - Here are the default values -->
        <Features>

            <!-- Configures the session cookie whether to be set with Secure flag.
Recommended value: true. -->
            <Feature key="secure-cookie" value="true" />

            <!--
                 Type of unique identifier generator to use to assign ids to SAML messages.
                 The value must be com.axway.st.server.sso.impl.UIDGenerator
            -->
            <Feature key="uid-generator" value="com.axway.st.server.sso.impl.UIDGenerator" />
        </Features>

        <!--
            Identity Provider resolution provides support for choosing the right Identity
Provider
            based on configuration and run-time metadata.
            If such resolution is not present, the first Identity Provider is selected among
ones
            specified under IdentityProviders element below.
            The supported ways to do that are by:
            1) Query parameter provided by a user request (see the example below).
            2) Header value provided by a user request. An example follow:
                <Header name="idp_id">
                    <Mapping value="keycloackIdp" entityId="https://st.keycloack.axway.int/"
/>
                    <Mapping value="shibbolethIdp"
entityId="https://st.shibboleth.axway.int/" />
                    <Mapping value="kerbIdP" entityId="kerberos" />
                </Header>
                In the example above if a user authentication request has header with name
'idp_id' and
                corresponding value equals to 'keycloackIdp', then Identity Provider with
entityId equals to
                https://st.keycloack.axway.int/ will be chosen to authenticate the user
agent.
           Note: Only one of these way can be done.
        -->
```

```
        <IdentityProviderResolution>
            <!--
                 Identity provider mapping using a query parameter. The name of query
parameter resolution will be
                 searched for in request parameters during runtime and its value should match
to the value attribute of a
                 Mapping element. If both query parameter name and value match, then
corresponding entityId is used to
                 select Identity Provider.
                 Examples:
                         1) https://localhost/?idp_id=keycloackIdp in the below case will
match the
                            Identity provider with entityId=https://st.keycloack.axway.int/
                         2) https://localhost/?idp_id=shibbolethIdp in the below case will
match the
                            Identity provider with entityId=https://st.shibboleth.axway.int/
                         3) https://localhost/?idp_id=kerbIdP in the below case will match
the
                            Identity provider with entityId=kerberos
             -->
            <QueryParameter name="idp_id">
                <!-- Note: The name of the query parameter/header should match the value of
the ST configuration option
                           LoginSettings.EndUser.SSO.idpResolverKey. -->
                <!-- Note: Ensure the name of the query parameter/header to be different than
ST configuration option
                           LoginSettings.EndUser.SSO.localIdpId in order to be able to
configure selection of ST as
                           local authentication provider and SSO Identity Providers. -->
                <Mapping value="keycloackIdp" entityId="https://st.keycloack.axway.int/" />
                <Mapping value="shibbolethIdp" entityId="https://st.shibboleth.axway.int/" />
                <Mapping value="kerbIdP" entityId="kerberos" />
            </QueryParameter>
        </IdentityProviderResolution>

        <!--
            This element is optional.
            Tenant resolution provides support for choosing the right Identity Provider based
on
            configuration and run-time metadata.
            If both tenant resolution and identity provider resolutions are present, then
tenant resolution takes precedence.
            Tenants are defined inside Identity Providers so resolving the IdP in turn will
resolve a tenant.
            The supported ways to do that are by:
            1) QueryParameter
                <QueryParameter name="idp_id">
                    <Mapping tenant="Axway" entityId="https://st.keycloack.axway.int/" />
                    <Mapping tenant="Sopra" entityId="https://st.shibboleth.axway.int/" />
                    <Mapping tenant="Apple" entityId="kerberos" />
                </QueryParameter>
            2) Header (example below)
            Notes:
                  1) Only one of these way can be done.
                  2) Header evaluation takes precedence on query parameter one.
                  3) If mapping is not present, IdentityProviderResolution is used.
                     If IdentityProviderResolution is not present first listed IdP is used.
        -->
        <TenantResolution>
            <!-- Note: The name of the query parameter/header should match the value of the
ST configuration option
                           LoginSettings.EndUser.SSO.idpResolverKey. -->
```

```
            <!-- Note: Ensure the name of the query parameter/header to be different than ST
configuration option
                          LoginSettings.EndUser.SSO.localIdpId in order to be able to
configure selection of ST as
                          local authentication provider and SSO Identity Providers. -->
            <Header name="idp_id">
                <Mapping tenant="Axway" entityId="https://st.keycloack.axway.int/" />
                <Mapping tenant="Sopra" entityId="https://st.shibboleth.axway.int/" />
                <Mapping tenant="Apple" entityId="kerberos" />
            </Header>
        </TenantResolution>
    </ServiceProvider>

    <!-- Identity provider definitions. Configures various aspects of interaction with
identity providers. -->
    <IdentityProviders>
        <!--
        Main attributes:
            entityId - Sets the unique identifier of the service provider. This identifier is
sent to the Identity Provider
                       so it can know who is requesting an authentication or a logout.
                       Add here EntityDescriptor entityID value, from the idpMetadata.xml
            metadataUrl - Specify the relative location of the metadata file.
                       Specifies a relative location of the metadata file to sso-enduser.xml
file.
                       NOTE: ST does not support the metadata URL to be a HTTP site.
            configurationUrl - should be an absolute path to Kerberos configuration file.
                       NOTE: ST does not support the configuration URL to be a network path.
                         verifyAssertionExpiration - Turn on/off verification of the validity
period of assertions. Consider to set to false if
                                          service provider and identity provider times are not
synchronized. Default: true.
            sign - If set to true, all SAML messages and their assertions sent by the service
provider will be signed.
                   There are a couple of features (see below) for fine-grained control of
signing. Optional - if not present, default value is false.
            userNameAttribute - Sets the name of the identity provider attribute that
provides the user name.
         -->

         <!-- Sample Keycloak Identity provider definition. -->
        <SamlIdentityProvider
                entityId="https://st.keycloack.axway.int/"
                metadataUrl="./keycloack-idp-metadata.xml"
                verifyAssertionExpiration="false"
                sign="true">

            <!-- Mappings tag is optional -->
            <Mappings>
                <!-- Filter mapping is optional. -->
                <FilterMapping>
                    <Filter>(department=426 AXW RD SOFIA)</Filter>
                    <OutputAttribute name="role">Developer</OutputAttribute>
                </FilterMapping>

                <!-- With this mapping, you can rename an attribute from the Identity
Provider, keeping its value. -->
                <!--
                     A system attributes mapping.
                     For email, UID, GID and homeDir ST expects the following renaming.
                -->
                <RenameMapping source="email" target="fdxEmail" />
                <RenameMapping source="uid" target="fdxUid" />
```

```
                <RenameMapping source="gid" target="fdxGid" />
                <RenameMapping source="homeDir" target="fdxHomeDir" />
                <!-- Custom attributes mapping examples (optional).-->
                <RenameMapping source="department" target="department" />
                <RenameMapping source="username" target="username" />
                <RenameMapping source="full name" target="fullName" />
                <RenameMapping source="last name" target="lastName" />
            </Mappings>

            <!-- Features control specific behavior of SAML message processing. -->
            <Features>
                <!-- Allows interaction with the IdP by plain HTTP. Default: false. -->
                <Feature key="saml-allow-http-connection" value="false"/>

                <!-- Allows unsigned assertions in messages received from the Identity
Provider. Default: false. -->
                <Feature key="saml-allow-unsigned-assertion" value="false"/>

                <!--
                    Enable or disable the signature verification of the metadata file and the
certification
                    path of the certificate used to sign. Set to false if metadata file is
not signed.
                    Default: true.
                -->
                <Feature key="saml-verify-metadata-signature" value="false"/>

                <!--
                    Enable or disable signing of Authentication Request messages. Presence of
this feature and its
                    value overrides the meaning of sign attribute of IdentityProvider element
above.
                -->
                <Feature key="saml-sign-authnrequest" value="true"/>

                <!--
                    Enable or disable signing of Logout Request messages. Presence of this
feature and its value
                    overrides the meaning of sign attribute of IdentityProvider element
above.
                 -->
                <Feature key="saml-sign-logoutrequest" value="true"/>

                <!--
                    Enable or disable signing of Logout Response messages. Presence of this
feature and its value
                    overrides the meaning of sign attribute of IdentityProvider element
above.
                -->
                <Feature key="saml-sign-logoutresponse" value="true"/>

            </Features>
        </SamlIdentityProvider>

        <!-- Sample Shibboleth Identity provider definition. -->
        <SamlIdentityProvider
                entityId="https://st.shibboleth.axway.int/"
                metadataUrl="./shibboleth-idp-metadata.xml"
                verifyAssertionExpiration="false"
                userNameAttribute="urn:oid:0.9.2342.19200300.100.1.1"
                sign="true" >

            <!-- Mappings tag is optional -->
```

```
            <Mappings>
                <!-- Filter mapping is optional. -->
                <FilterMapping>
                    <Filter>(department=R&amp;amp;D)</Filter>
                    <OutputAttribute name="rolename">Developer</OutputAttribute>
                </FilterMapping>
                <!-- A system attributes mapping. -->
                <RenameMapping source="email" target="fdxEmail" />
                <RenameMapping source="uid" target="fdxUid" />
                <RenameMapping source="gid" target="fdxGid" />
                <RenameMapping source="homeDir" target="fdxHomeDir" />
                <!-- Custom attributes mapping (optional).-->
                <RenameMapping source="department" target="department" />
                <RenameMapping source="username" target="username" />
                <RenameMapping source="full name" target="fullName" />
                <RenameMapping source="last name" target="lastName" />
            </Mappings>

            <!-- Features control specific behaviour of SAML message processing. -->
            <Features>
                <!-- Allows interaction with the IdP by plain HTTP. Default: false. -->
                <Feature key="saml-allow-http-connection" value="false"/>

                <!-- Allows unsigned assertions in messages received from the Identity
Provider. Default: false. -->
                <Feature key="saml-allow-unsigned-assertion" value="false"/>

                <!--
                    Enable or disable the signature verification of the metadata file and the
certification
                    path of the certificate used to sign. Set to false if metadata file is
not signed.
                    Default: true.
                -->
                <Feature key="saml-verify-metadata-signature" value="false"/>

                <!--
                    Enable or disable signing of Authentication Request messages. Presence of
this feature and its
                    value overrides the meaning of sign attribute of IdentityProvider element
above.
                -->
                <Feature key="saml-sign-authnrequest" value="true"/>

                <!--
                    Enable or disable signing of Logout Request messages. Presence of this
feature and its value
                    overrides the meaning of sign attribute of IdentityProvider element
above.
                 -->
                <Feature key="saml-sign-logoutrequest" value="true"/>

                <!--
                    Enable or disable signing of Logout Response messages. Presence of this
feature and its value
                    overrides the meaning of sign attribute of IdentityProvider element
above.
                -->
                <Feature key="saml-sign-logoutresponse" value="true"/>

            </Features>
        </SamlIdentityProvider>
```

```
        <!-- A Kerberos IdP sample definition. -->
        <KerberosIdentityProvider
                entityId="kerberos"
                configurationUrl="C:/Axway/SecureTransport/STServer/conf/sso/krb5-
login.conf">

        </KerberosIdentityProvider>

    </IdentityProviders>
</SSOConfiguration>
```

Multiple Identity Provider configuration

SecureTransport supports the multiple Identity Provider configuration. First, you should configure every Identity Provider. Then, for every Identity provider follow the steps described for administrators and end-users. Refer to *Enable Single Sign-On (SSO) for administrators* on page 432 and *Enable Single Sign-On (SSO) for end-users* on page 440.

Note For both administrators and end-users every Identity Provider should be in a different Identity Provider element. For a SAML-based Identity Provider you should use the `<SamlIdentityProvider>` element and for Kerberos you should use the `<KerberosIdentityProvider>` element.

Note For every Identity Provider defined in either the `sso-admin.xml` or `sso-enduser.xml` file, you should have a different Identity Provider metadata file. When uploading the SSO Configuration Files ZIP file, make sure that all required files are in the ZIP.

Note The client name has to be the same on all Identity Providers, SecureTransport only supports one service provider per component (Administrator and End-user).

Identity Provider resolution

Identity Provider resolution provides support for choosing the right Identity Provider based on server configuration and run-time metadata. If such resolution is not present, the first Identity Provider is selected by default, among ones specified under `<IdentityProviders>` element.

For both `sso-enduser.xml` and `sso-admin.xml` configuration files the element to edit is `<IdentityProviderResolution>` element under `<ServiceProvider>` element. The mapping can be done in either one of the following ways:

1. Query parameter – Identity provider mapping using a query parameter. The name of query parameter resolution will be searched for among request parameters during runtime and its value should match to the Identity Provider alias.
2. Header - Header value provided by a user request. The name of header resolution will be searched for among request header during runtime and its value should match to the Identity Provider alias.

Note Only one of these mappings can be done.

Note By default, if no Identity Provider ID is set, the first listed Identity Provider in the configuration file is used.

Identity Provider resolution for administrators

To configure Identity provider resolution for administrators, open the `sso-admin.xml` and edit the `<IdentityProviderResolution>` element under the `<ServiceProvider>` element.

Query parameter resolution example:

```
<QueryParameter name="idp_id">
    <Mapping value="keycloakIdp" entityId="https://st.keycloak.axway.int/" />
    <Mapping value="shibbolethIdp" entityId="https://st.shibboleth.axway.int/" />
</QueryParameter>
```

Note The value of name attribute in the `<QueryParameter>` element should match the Server configuration option `LoginSettings.Admin.SSO.idpResolverKey` value. The default value is `idp_id`.

Note Ensure the name of the query parameter/header is different than SecureTransport configuration option `LoginSettings.Admin.SSO.localIdpId` to be able to configure the selection of SecureTransport as local authentication provider and SSO Identity Provider. For more information on how to use SecureTransport as Identity provider, refer to *SecureTransport as an Identity Provider* on page 455.

In the example above we already have 2 identity providers defined in `sso-admin.xml` file.

Suppose we have the following request:

- `https://<ST>/?idp_id=shibbolethIdp`, where `<ST>` is the IP of the running SecureTransport instance.

SecureTransport will choose the Identity Provider with an `entityId='https://st.shibboleth.axway.int/'`. If no such Identity Provider is found, the login will be effectively rejected.

Header resolution example:

```
<Header name="idp_id">
    <Mapping value="keycloakIdp" entityId="https://st.keycloak.axway.int/" />
    <Mapping value="shibbolethIdp" entityId="https://st.shibboleth.axway.int/" />
</Header>
```

Note The value of name attribute in `<Header>` element should match the SecureTransport configuration option `LoginSettings.Admin.SSO.idpResolverKey` value. The default value is `idp_id`.

Note Ensure that the name of the query parameter/header is different than SecureTransport configuration option `LoginSettings.Admin.SSO.localIdpId` in order to be able to configure selection

of SecureTransport as local authentication provider and SSO Identity Provider. For more information on how to use SecureTransport as Identity Provider, refer to *SecureTransport as an Identity Provider* on page 455.

In the example above we already have two Identity Providers defined in the `sso-admin.xml` file.

Suppose we have request to the running SecureTransport instance that contains the following Header:

- `keycloakIdp : https://st.keycloak.axway.int/`

SecureTransport will choose the Identity Provider with an `entityId= 'https://st.keycloak.axway.int/'`. If no such Identity Provider is found, the login will be effectively rejected.

Identity provider resolution for end-users

To configure Identity provider resolution for end-users, open the `sso-enduser.xml` file and edit the `<IdentityProviderResolution>` element under the `<ServiceProvider>` element.

Query parameter resolution example:

```
<QueryParameter name="idp_id">
    <Mapping value="keycloakIdp" entityId="https://st.keycloak.axway.int/" />
    <Mapping value="shibbolethIdp" entityId="https://st.shibboleth.axway.int/" />
    <Mapping value="kerbIdP" entityId="kerberos" />
</QueryParameter>
```

Note The name attribute in `<QueryParameter>` element should match the SecureTransport configuration option `LoginSettings.EndUser.SSO.idpResolverKey` value. The default value is `idp_id`.

Note Ensure that the name of the query parameter/header is different than SecureTransport configuration option `LoginSettings.EndUser.SSO.localIdpId` to be able to configure selection of SecureTransport as local authentication provider and SSO Identity Provider. For more information on how to use SecureTransport as Identity provider, refer to *SecureTransport as an Identity Provider* on page 455.

In the example above we already have three Identity Providers defined in `sso-enduser.xml` file.

Suppose we have the following request:

- `https://<ST>/?idp_id=shibbolethIdp`, where `<ST>` is the IP of the SecureTransport instance.

SecureTransport will choose the Identity provider with an `entityId='https://st.shibboleth.axway.int/'`. If no such Identity Provider is found, the login will be effectively rejected.

Header resolution example:

```
<Header name="idp_id">
```

```
    <Mapping value="keycloakIdp" entityId="https://st.keycloak.axway.int/" />
    <Mapping value="shibbolethIdp" entityId="https://st.shibboleth.axway.int/" />
    <Mapping value="kerbIdP" entityId="kerberos" />
</Header>
```

Note The name attribute in `<Header>` element should match the Server configuration option `LoginSettings.EndUser.SSO.idpResolverKey` value. The default value is `idp_id`.

Note Ensure that the name of the query parameter/header is different than SecureTransport configuration option `LoginSettings.EndUser.SSO.localIdpId` in order to be able to configure selection of SecureTransport as a local authentication provider and SSO Identity Provider. For more information on how to use SecureTransport as Identity Provider, refer to *SecureTransport as an Identity Provider* on page 455.

In the example above we already have 3 identity providers defined in the `sso-enduser.xml` file.

Suppose we have request to SecureTransport instance that contains the following Header:

- `keycloakIdp : https://st.keycloak.axway.int/`

SecureTransport will choose the Identity Provider with an `entityId= 'https://st.keycloak.axway.int/'`. If no such Identity provider is found, the login will be effectively rejected.

Tenant resolution

Tenant resolution provides a support of multiple identity providers. If not present tenant is null. The supported ways to do that are by:

1. Query parameter
2. Header

Note Only one of the ways can be completed.

The tenant resolution is the same as the Identity Provider resolution in terms of syntax and meaning. However, the precedence is to take the Identity Provider resolution first. Header mapping, if present, will always be the first choice, no matter where it is placed; in the Identity Provider resolution or in tenant resolution.

Configure Single Sign-On (SSO) for streaming

You can configure Single Sign-On (SSO) functionality when using streaming setups.

Here are the basic steps:

1. Setup streaming. Refer to *Configure SecureTransport Server to Edge streaming communication on page 239*.
2. On backend, configure the SSO for end-user. Refer to *Enable Single Sign-On (SSO) for end-users on page 440*.

3. On the SecureTransport Edges, navigate to **Authentication > Login Settings** and set **Required** for *SSO* for end-users.

Note The `sso-enduser.xml` file is taken from backend, corresponding to SSO Service Provider Entity ID value in the **Setup > Network Zones**.

For the backend network zones:

- Private zone - SSO Service Provider Entity ID should be the Entity ID pointing to the Server.
- Streaming zone - SSO Service Provider Entity ID should be the Entity ID pointing to the Edge.

Note In a SecureTransport deployment, consisting of both backend and edge nodes, SSO must be configured and enabled on all nodes even if users are only logging in through the edge. On the edge, you must enable SSO for end-users. All backend and edge nodes in streaming must have the same setting for SSO Authentication (Disabled or Required).

Configure Single Sign-On (SSO) for clusters

To configure SSO in clusters, refer to *Enable Single Sign-On (SSO) for administrators* on page 432 and *Enable Single Sign-On (SSO) for end-users* on page 440.

SecureTransport as an Identity Provider

Using SecureTransport as an Identity Provider allows users to use SecureTransport as an Identity Provider. The users who authenticate using SecureTransport as an Identity Provider, will authenticate using the password stored internally in the SecureTransport, and will be treated as local accounts.

The following configuration options are added in the Server Configuration options:

- For administrators:
  - `LoginSettings.Admin.SSO.idpResolverKey`
  - `LoginSettings.Admin.SSO.localIdpId`
- For end-users:
  - `LoginSettings.EndUser.SSO.idpResolverKey`
  - `LoginSettings.EndUser.SSO.localIdpId`

The `idpResolverKey` corresponds to the Query Parameter name attribute and can be used in both a query parameter as well as a header. This is the key that will be used when requesting local or Identity Provider authentication. The default value is: `idp_id`.

The `localIdpId` corresponds to the value in the mapping that will force SecureTransport to not trigger the SSO flow and continue with local authentication. The default value is: `ST_IDP`.

The value of these options is configurable. For more information about how to edit the Server Configuration options, refer to *Update configuration files* on page 336.

Example for using SecureTransport as a Identity Provider:

```
https:// <ST IP>/?idp_id=ST_IDP
```

Note Options for using SecureTransport as an Identity Provider can be used either as query parameters or for using requests with header.

Single Sign-On SSO authentication flows

The supported Single Sign-on (SSO) and Single Logout (SLO) authentication flows for SecureTransport are:

- *Service Provider initiated Single Sign-On (SSO) authentication flow* on page 456
- *Identity Provider initiated Single Sign-On (SSO) authentication flow* on page 456
- *Service Provider initiated Single Logout (SLO) authentication flow* on page 456
- *Identity Provider initiated Single Logout (SLO) authentication flow* on page 457

Service Provider initiated Single Sign-On (SSO) authentication flow

The Service Provider initiated Single Sign-On (SSO) authentication flow describes the following behavior, when administrator or end-user access the protected resource directly on a SecureTransport site without being logged on. The user account is not managed on the SecureTransport side, it's managed by a third-party Identity Provider (IdP). The SecureTransport sends an authentication request to the Identity Provider. Practically, the authentication for the admin or user is done by an external agent. After authentication the IdP sends the response to SecureTransport, containing the result of the authentication process and mapped user-specific attributes. SecureTransport uses these attributes in various scenarios.

Identity Provider initiated Single Sign-On (SSO) authentication flow

The Identity Provider initiated flow describes the behavior when administrator or end-user chooses the Identity Provider that will be used for the authentication flow. This is configurable through the browser accessing the URL which is provided from the IdP. For that purpose, the Identity Provider first should be configured with specialized links that refer to the desired Service Provider (SecureTransport in our case).

Service Provider initiated Single Logout (SLO) authentication flow

The Service Provider initiated flow describes the behavior when the administrator or end-user, has already accessed Service Provider (SecureTransport) and the authentication is successful. In this case, SecureTransport initiates logout request that is sent to the IdP. After receiving the logout request from

SecureTransport, the Identity Provider sends a logout response to every other Service Provider which was connected, forcing them to effectively logout the current user. After successful logout, the admin or end-user is redirected to the original IdP login page.

Identity Provider initiated Single Logout (SLO) authentication flow

The Identity Provider initiated Single Logout (SLO) flow describes the behavior when the administrator or the end-user, has already accessed the Service Provider (SecureTransport) and the authentication is successful. Then, IdP request for the logout is executed, and sends a logout request to every other Service Provider which was connected. The logout operation is performed and the administrator or end-user is successfully logged out. After successful logout, the admin or end-user is redirected to the original IdP login page.

Configure Kerberos as an Identity Provider in SecureTransport

This topic describes the configuration of Kerberos as an Identity Provider and the configuration of SecureTransport as a Service provider. In order to setup Kerberos to work with SecureTransport you need to have three files; the Kerberos configuration file, the `.keytab` file, and the `sso-enduser.xml` file.

Generate a Kerberos keytab file on Windows

For information about how to configure the Kerberos for UNIX-like systems, refer to the official Kerberos documentation.

A `keytab` file contains pairs of Kerberos principals and encrypted keys that are derived from the Kerberos password. You can use a `keytab` file for resource authentication without entering a password.

1. Open CMD on your Domain Controller.
2. Use the `ktpass` command to set up an identity mapping for the service principal:

```
ktpass -princ HTTP/<FQDN>@<realm> -mapuser <SPN> -pass <password> -out
<PATH_TO_KEYTAB_FILE> -ptype KRB5_NT_PRINCIPAL -crypto AES256-SHA1
```

Where:

- FQDN - Fully qualified domain name of the Microsoft Windows Server machine (the command is case-sensitive, therefore make sure you precisely enter the FQDN of your Windows machine)
- Realm - Domain Name (**with UPPERCASE letters**). In case there are multiple domains that have Active Directory replication, use the primary domain name instead of the sub domain.
- SPN - Service Principal Name (the user name of a user created in the Active directory of your Domain Controller)

Path to keytab file – Enter a valid path on your Domain Controller Windows machine for the `keytab` file to be generated to. Example: `C:\key.keytab`

The selected encryption may be different.

After the command is successfully executed, the `keytab` file is generated in the specified location.

Create a Kerberos configuration file

```
com.sun.security.jgss.krb5.accept {
    com.sun.security.auth.module.Krb5LoginModule required
    principal="<HTTP/FQDN@REALM>"
    keyTab="ABSOLUTE_PATH_TO_KEYTAB_FILE"
    useKeyTab=true
    storeKey=true
    isInitiator=false
    doNotPrompt=true;
};
```

Note In a Standard Cluster or Enterprise Cluster environment, you have to specify the login modules for all Cluster nodes inside the Kerberos configuration file.

Main attributes:

- FQDN - Fully qualified domain name of the Microsoft Windows Server machine on which SecureTransport is installed (as entered in the `ktpass` command)
- Realm - Domain Name (as entered in the `ktpass` command)
- `keyTab` - The path to which the SSO configuration files are uploaded in SecureTransport. If SecureTransport is installed in C drive of your Windows machine, the path would be: `C:/Axway/SecureTransport/STServer/conf/sso/key.keytab`

Edit the sso-enduser.xml file

In order to use Kerberos as IdP for end-users you need to edit and upload the `sso-enduser.xml` file.

Verify that it contains Kerberos Identity Provider. Example:

```
<KerberosIdentityProvider
entityId="kerberos"
configurationUrl="C:/Axway/SecureTransport/STServer/conf/sso/krb5Login.conf">
</KerberosIdentityProvider>
```

`configurationUrl` - The absolute path to the `krb5Login.conf` file location after the SSO configuration files are uploaded to SecureTransport. If SecureTransport is installed on the C drive of your Windows machine, the path would be the same as the one in the example.

Configure supported browser authentication

For a list of SecureTransport supported browsers, refer to *Browsers* on page 37.

To use Kerberos with SecureTransport the browsers will need some fine tuning.

For example:

For Microsoft Edge:

1. The SecureTransport instance should be added to the *Trusted sites* list.
2. Navigate to **Internet Options > Security tab > Trusted sites > Custom level**. Scroll all the way to bottom under *User Authentication* and *Logon*, select **Automatic logon with current user name and password**.

   Note This action will prevent the additional authentication pop up.

For Firefox:

Navigate to the **about:config** URL in Firefox and set *http:///,https://* to the following properties:

`network.negotiate-auth.trusted-uris`

`network.automatic-ntlm-auth.trusted-uris`

For Chrome:

Google Chrome in Windows will use the Internet Explorer settings, so configure within Internet Explorer's Tools.

For Safari:

Mac OS supports SPNEGO with Kerberos as an authentication mechanism if Mac OS is joined to the Active directory.

Verify SSO authentication

To verify end-user SSO authentication, log onto a Windows machine joined to the same domain used as a realm in the SSO configuration, with a user existing in the Active directory. The Windows machine should be different from the one SecureTransport is installed to. Configure your browser for Kerberos authentication, and enter the URL of the ST Web Client (STWC) using the fully qualified domain name. You should be logged into STWC automatically with the same user account that you logged onto the Windows machine.

Pluggable authentication

A custom authentication plug-in allows you to implement your own authentication logic and override the SecureTransport authentication mechanism. The custom authentication has priority over the rest authentication types except for the Single sign-on (SSO). Only one authentication plug-in can be enabled at a

time.

Note A collection of authentication plug-ins maintained by Axway is available in the Amplify Repository.

Pluggable Authentication features:

- End-user authentication over the HTTP, FTP, and SSH.
- Administrator authentication over HTTP only.
- Supports multiple authentication methods for administrators and end-users: basic authentication (user name/password), certificates, and dual authentication (both username/password and a certificate)
- The plug-in custom configuration is stored in the Server Configuration and can be exported and imported with it.
- Error and messages from the plug-in are displayed in SecureTransport Server log.

The following conditions apply to Pluggable authentication:

- The deployment of multiple authentication plug-ins is not supported.
- To be authenticated by a plug-in, a SecureTransport account must be created with the "Password is stored locally" option disabled.
- On an Edge server, Pluggable authentication can be configured for administrators only.
- Log-in Restriction Policies work if an IP address is used, and may not work when a username is used for a policy (due to the fact that the plug-in may accept one username and return a completely different one).

Before your custom plug-in can be configured and used, it must be deployed, registered, and then enabled in the Server Configuration.

Plug-in deployment

The custom authentication logic (plug-in) is packaged as .jar file that follows the set of conventions described in the Developer Guide.

To deploy an authentication plug-in, place its JAR file in the `/<st_dir>/plugins/authentication/` directory, and restart the Admin and the TM daemons.

In a cluster environment, the plug-in should be deployed on all nodes before they are added to the cluster, and the Admin and Transaction Manager services should be restarted on all nodes.

Plug-in registration

SecureTransport identifies the plug-in by the name of its JAR file. Plug-ins are discovered and registered at the Admin daemon start. Each authentication plug-in is added to two configuration registries in the *Server Configuration* page:

- `Plugins.Authentication.Admin.Registry`
- `Plugins.Authentication.EndUser.Registry`

If the plug-in has a custom configuration, its configurable options are added in the Server Configuration page in the following format:

- `Plugins.Authentication.<plugin_name>.<config_option>`

Note The plug-in configuration options are exported upon server configuration export.
Before importing a server configuration with custom plug-in configuration options, the relative plug-ins must be deployed. Otherwise, their configuration options will not be imported.

Plug-in activation

After being registered, the authentication plug-ins are added to the admin and the end-user registries in the Server Configuration, but they are disabled (have a hash symbol in front of their names). SecureTransport will not automatically activate a newly registered plug-in. To activate a plug-in, remove the # symbol from its name.

Only one plug-in can be enabled per registry at a time. Otherwise, the authentication fails.

If no plug-in is enabled, users are authenticated by using the SecureTransport internal authentication logic.

Note Plug-in activation does not require service restart.

Plug-in management

To undeploy a plug-in:

1. Delete the JAR file from the `/<st_dir>/plugins/authentication/` directory.

2. Restart the Admin and TM daemons.
The plug-in name is then removed from the registries along with its configuration options.

When you uninstall SecureTransport, the plug-ins JAR files are also removed.

To redeploy or update a plug-in:

1. Undeploy the existing plug-in.

2. After the Admin and TM daemon restart, go to the Server Configuration registries and make sure the plug-in is removed from the registries.

3. Deploy the new plug-in (version).

After the restart, the new plug-in is added to the admin and end-user authentication plug-ins list.

Plug-in configuration

Successful plug-in usage depends on both the authentication methods that are supported by the plug-in and the correct configuration in **Login Settings**.

Example scenarios and troubleshooting

If you have enabled a plug-in that supports Basic authentication, you must configure the users to log in with a username and a password.

If the enabled plug-in supports authentication with a certificate, you must configure certificate authentication.

To configure dual authentication, you must enable a plug-in that supports both authentication methods.

In case the plug-in is enabled but not configured correctly, and you cannot log in to the administration tool to reconfigure, undeploy the plug-in.

In a Standard Cluster, if the jar file is not uploaded to the secondary node, the configuration will not be considered correct, and an error message will be displayed in the Server Log at startup.

Pluggable authentication status

The Login settings page displays the Pluggable authentication status and provides a link to the server configuration option related to the end-user authentication plug-ins list.

Messages and errors:

- "*Plugin <plugin name> enabled.*"- an authentication plug-in with that name is enabled.
- "*No plugins registered.*"- there are no registered plug-ins.
- "*No authentication plugin enabled.*"- there are registered plug-ins, but none of them is enabled.
- "*Invalid configuration. Several authentication plug-ins are enabled.*"- multiple plug-ins are enabled.

Note Plug-in authentication cannot be applied for a specific user class. When enabled, it applies to all user classes.

Plug-in authentication notifications

On each login request, the following messages are displayed in the Server log:

- INFO: "`Authentication call to <plugin_name>.`"- notification of an authentication request sent to an enabled authentication plug-in.
- INFO: "`<plugin_name> successfully authenticated user <username>. Authentication result: SUCCESS, '<plugin_message>'`"- notification of a successful authentication by a plug-in
- INFO: "`<plugin_name> failed to authenticate user. Authentication result: FAILURE, '<plugin_message>'`"- notification of an unsuccessful authentication
- INFO: "`<plugin_name> was unable to authenticate user. Authentication result: CONTINUE, '<plugin_message>'`"- the plug-in does not recognize the user and indicates that an SecureTransport internal authentication must be executed.

Note All data sent/ received to/from a plug-in is available on a DEBUG log level.

User mapping

If the authentication process is successful, user mapping is performed.

End-user only

An end-user can log in to SecureTransport as a virtual account, account template, or an external account. In the first two cases, the user is authenticated based on the user properties pre-defined in SecureTransport; In order for an external user to be successfully authenticated, the plug-in must provide the following attributes: login name, UID, GID, home directory.

1. The user logs in - the user class is determined by the user's properties (login name, GID, UID, Address or the custom expression).

2. SecureTransport searches for an account with the same login name and an externally stored password in its database. If such account is found, the user is mapped.

3. If there is no virtual account with this login name, SecureTransport tries to map the user to an account template (based on the user class).

4. If there is no matching account template, the user logs in as an external user. To determine the user, SecureTransport uses the attributes returned by the plug-in. If the plug-in does not return those attributes, the authentication fails.

Admin only

1. If an admin account with that login name and an externally-stored password is not present, the authentication will fail.
2. For certificate or dual authentication, the respective checkboxes on the Admin Settings page must be enabled.
3. If the plug-in successfully authenticates an administrator but their account is locked or it is of a 'dbsetup' type, the authentication will fallback to the internal SecureTransport logic.

System attributes

- Session attributes

Attribute Name	Expression Syntax
email	${sess.STSESSION_PLUGIN.email}
UID	${sess.STSESSION_PLUGIN.UID}
GID	${sess.STSESSION_PLUGIN.GID}

Attribute Name	Expression Syntax
homeDir	${sess.STSESSION_PLUGIN.homeDir}
username	${sess.STSESSION_PLUGIN.username}

- The first element in a custom attribute `<attribute>`

```
${sess.STSESSION_PLUGIN.additionalAttributes['<attribute>'][0]}
```

- Advanced routing environment attributes

Attribute Name	Expression Syntax
email	${plugin.email}
UID	${plugin.uid}
GID	${plugin.gid}
homeDir	${plugin.homeDir}
userName	${plugin.userName}

- The first element in the attribute `<attribute>`

```
${plugin.attributes['attribute'][0]}
```

Usage in User Class custom expressions

You can use plug-in attributes in the User Class custom expressions.

For example:

- PLUGIN.UID
- PLUGIN.GID
- PLUGIN.email
- PLUGIN.homeDir
- PLUGIN.pluginName

will return the corresponding attribute value from plug-in data.

For PLUGIN.attributes, you can use the built-in memberOf function (see the **Access Control** -> **User classes** -> **Custom expressions**, also on this page update the possible values for `DXAGENT_USERLOGINTYPE`).

When a custom authentication plug-in is used, the `DXAGENT_USERLOGINTYPE` value is `PLUGIN`; the

`DXAGENT_AUTHN_PLUGIN_NAME` value is the name of the plug-in that performed the authentication for the current user.

Login settings

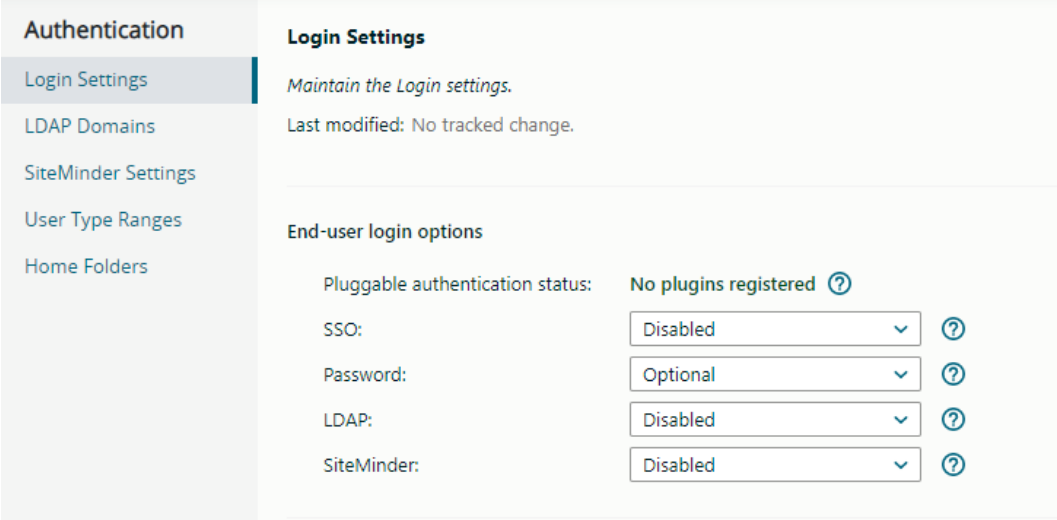
You can use the *Login Settings* page to disable or require Single Sign-On (SSO), enable or disable certificate authentication and to specify client certificate authentication for administrators, enable or disable password authentication and set LDAP and SiteMinder authentication levels for end-users.

The following topics describe the end-user and administrator login options:

- *End-user login options* on page 465
- *Administrator login options* on page 468

End-user login options

You can use the end-user login options to disable or require end-user Single Sign-On (SSO), check the current pluggable authentication status, enable or disable password authentication, or specify for which classes it will be applied, and set LDAP and SiteMinder authentication levels.



Note The end-user login options Password, LDAP, and SiteMinder are not available on the SecureTransport Edges.

Disable or require end-user Single Sign-On (SSO)

Use the following procedure to disable or require end-user Single Sign-On (SSO).

1. Select **Authentication > Login Settings**.

   The *Login Settings* page is displayed.

2. Under *End-user login options* in the **SSO** drop-down menu, select one of the following values:
   - **Disabled** - SSO will not be used.
   - **Required** - Redirection to Identity provider will always be performed. If the authentication with the Identity Provider (IdP) fails, the login will be rejected. This state will override any of the existing authentication methods via HTTP(s) for end-user - Certificate for HTTPS, LDAP with HTTP(s) and SiteMinder.

     Note Requires Transaction Manager service restart on back-end.

     Note SSO login option is applicable only for HTTP(s).
3. Click **Save**.

Note In order to configure SSO go to: *Server Configuration Files* edit page.

For information on editing and updating the server configuration files, refer to *Single Sign-On (SSO) and Single Logout (SLO)* on page 426.

Pluggable authentication

Shows the Pluggable authentication status and provides a link to the Server Configuration option related with the end-user's available deployed custom plug-ins list.

For information, refer to *Pluggable authentication* on page 459.

End-user password authentication

This option allows you to specify whether the end users need to provide a password in addition to authenticating with a certificate. The *Client Certificate* option can be configured per server. For information, refer to *Server control* on page 259.

Use the following procedure to specify password authentication for end-users.

1. Select **Authentication > Login Settings**.

   The *Login Settings* page is displayed.
2. Under *End-user login options*, select one of the three options in the **Password** drop-down menu to specify password authentication for end-users.

   Optional - A password is required only if a certificate is not presented.

   Required - A password is required in addition to certificate authentication option for users from all user classes.

   Required for user classes - A password is required in addition to certificate authentication option only for one or more comma-separated user classes inserted in the text field. User classes are case sensitive.

Note User classes with a comma inside their 'Class Name' should not be used due to the comma being a delimiter.

Note To enable or disable password authentication for Specific user classes from REST API you must update both configuration options.

```
LoginSettings.Certificate.RequirePassword and
LoginSettings.Certificate.RequirePassword.UserClasses.
```

Example: To enable password authentication for specific user class "VirtualClass" you must update

```
LoginSettings.Certificate.RequirePassword with value "true" and
LoginSettings.Certificate.RequirePassword.UserClasses with value
"VirtualClass".
```

3. Click **Save**.

For more information about how-to instructions for creating User Classes, see *User classes* on page 771.

Enable or disable LDAP

Use the following procedure to enable or disable LDAP.

1. Select **Authentication > Login Settings**.

   The *Login Settings* page is displayed.

2. Under *End-user login options* in the **LDAP** drop-down menu, select one of the following values:

   Disabled - The LDAP will not be used.

   Optional - SecureTransport searches the SecureTransport database before it searches the LDAP databases in the default domains. If no such user is found in SecureTransport and LDAP databases, then the login will be rejected.

   Required - An LDAP user will be required. If no such user exists, the login will be rejected.

   For details, see *LDAP logins* on page 476.

3. Click **Save**.
4. Restart the TM Server.

You must create one or more domains before SecureTransport can use LDAP to authenticate users. For information on creating LDAP domains, refer to *Create an LDAP domain* on page 479.

Enable or disable SiteMinder

Use the following procedure to enable or disable SiteMinder.

1. Select **Authentication > Login Settings**.

   The *Login Settings* page is displayed.

2. Under *End-user login options* in the **SiteMinder** drop-down menu, select one of the following values:

   Disabled - The SiteMinder configuration will not be used.

   Optional - The SiteMinder configuration may be used.

3. Click **Save**.

You must configure SiteMinder before you can use SiteMinder to authenticate users. For information on configuring SiteMinder, refer to *SiteMinder integration configuration* on page 470.

Administrator login options

You can use the administrator login options to enable or disable administrator Single Sign-On (SSO), check the current pluggable authentication status, and to enable or disable administrator certificate requirements.

Enable or disable administrator Single Sign-On (SSO)

Use the following procedure to enable or disable administrator Single Sign-On (SSO).

1. Select **Authentication > Login Settings**.

   The *Login Settings* page is displayed.

2. Under *Administrator login options* in the **SSO** drop-down menu, select one of the following values:
   - **Disabled** - SSO will not be used.
   - **Required** - Redirection to the Identity Provider (IdP) will always be performed. If the authentication with the IdP fails, the login will be rejected.
3. Click **Save**.

Note For more information of how to configure SSO for Administrators, refer to *Single Sign-On (SSO) and Single Logout (SLO)* on page 426.

For information on editing and updating the server configuration files, refer to *Update configuration files* on page 336.

Pluggable authentication

Shows the Pluggable authentication status and provides link to the Server Configuration option related with the end-user's available deployed custom plug-ins list.

For information, refer to *Pluggable authentication* on page 459.

Configure administrator certificate requirement and level

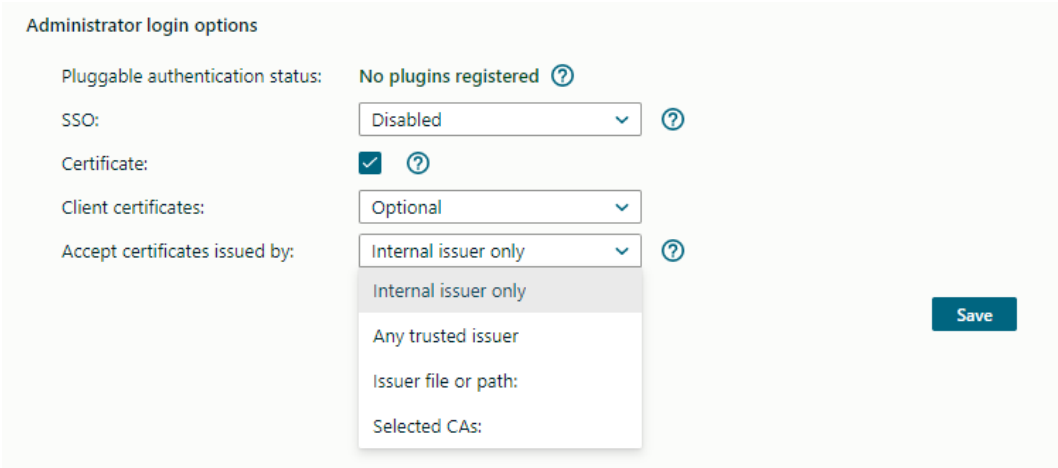
Set the certificate settings to allow administrators to log in by using a client certificate or to use dual authentication with both a certificate and a password. You can enable the ability to login with a client certificate, determine whether the certificate is optional or required, select the certificate issuer, and set the certificate chain limit.

When you want to access the Administration Tool and you have enabled client certificates, you are prompted to select the certificate you are using. Once the certificate is verified, you are logged in unless dual authentication is required. If certificates are optional and you do not select one, the login page is displayed. If SecureTransport cannot verify the certificate, or certificates are required and you did not select one, a connection error displays and you cannot log in.

If you are unable to successfully log in when using a certificate, clear the browser's SSL state, or close the browser and try again with a new browser instance.

1. Select **Authentication > Login Settings**.
2. The *Login Settings* page is displayed.
3. Under *Administrator login options*, select **Certificate** to allow administrators to log in using client certificates.

   The *Client Certificate Settings* pane and the remaining fields are displayed.



4. Select either **Optional** or **Required** in the **Client certificates** drop-down menu. If you select **Optional**, administrators do not need a certificate. If you select **Required**, each administrator must have a client certificate set up. If certificates are required, all administrators must be mapped to a certificate, and all users must present a valid trusted certificate to gain access to the login page.
5. Select one of the following choices from the **Accept certificates issued by** drop-down menu:
   - **internal issuer only** – The certificate must be issued by the internal CA. See *Manage the internal CA* on page 57.
   - **any trusted issuer** – The certificate must be issued by any of the trusted CAs (**Setup > Certificates > Trusted CA Certificates** tab). See *Manage trusted CAs* on page 55.
   - **issuer file or path** – The certificate must be issued by a CA whose certificate is in a file you specify.
   - **selected CAs** - Select one or more trusted CAs from the drop-down list. Only certificates from the selected CAs will be accepted.
6. If you select **issuer file or path:**, the following fields are displayed:
   - A field that you use to specify the location of the certificate PEM-encoded (`.pem`) file or a directory that contains the PEM-encoded files.

     You can type either a fully qualified file or path names or a file or path names relative to `<FILEDRIVEHOME>`. Do not put the PEM-encoded files in the keystore directory, `<FILEDRIVEHOME>/lib/certs/issuers`, because the certificates in that directory are regenerated from the database when servers start.

- A **Limit certificate chain depth to** field. Type a number that sets the maximum number of levels for SecureTransport to go through in validating the certificate up to a trusted root. For example, if you set the chain depth to 1, then only a certificate issued directly by a trusted root is allowed and a certificate issued by an intermediate CA is rejected.

7. Click **Save**.
8. Restart the Administration Tool server using the `stop_admin` and `start_admin` commands. If you are running on Windows, you can also use the **Services** console to restart the `admin` service.

If you choose to use certificates for administrator logins, a **Certificate DN** field displays in the *New Administrator* and *Edit Administrator* pages where you must provide the certificate domain name information. For more information, see *Add an administrator account* on page 702.

Note The Client Certificate Settings option will be set to **Disabled** when administrator SSO login option is set to **Required**.

SiteMinder integration configuration

SiteMinder is a third-party application that controls user access to secured applications and provides a Single Sign-On (SSO) portal. A SSO portal is a Web gateway or proxy that enables users to access multiple secured Web applications using a single user name and password they provide once at the start of the user session.

SecureTransport can be integrated into a SiteMinder SSO environment and use SiteMinder to SSO authenticate and authorize resource access using only HTTP or HTTPS.

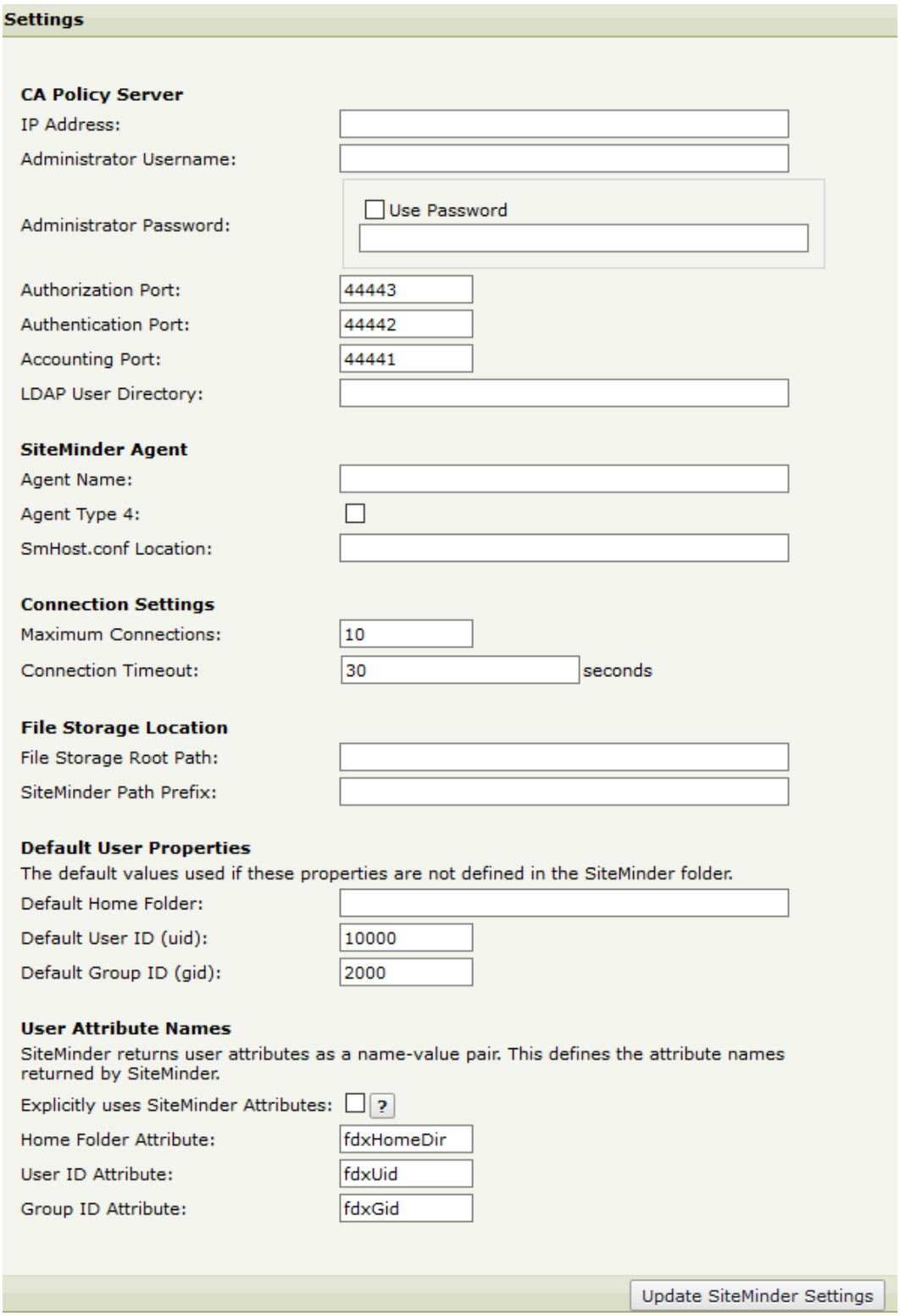
Note Before configuring SiteMinder settings, be sure to read *SiteMinder integration* on page 412.

Before using SecureTransport with SiteMinder, you must configure the SiteMinder settings using the SecureTransport Administration Tool.

Note If SecureTransport is deployed in a secure perimeter network (DMZ) configuration, configure the SiteMinder settings on SecureTransport Server as described in this topic. The *SiteMinder Settings* page is not available on SecureTransport Edge.

1. Select **Authentication > SiteMinder Settings**.

   The *CA SiteMinder Setting* page is displayed.



2. Provide the information as described in the following table:

Name	Description	Required/ optional
IP Address	The network address of the SiteMinder Policy Server.	Required
Administrator Username	The user name used to connect to the SiteMinder database.	Optional
Administrator Password	If a password is required, select Use Password and enter it in the field provided. **Note** Exported configuration from SecureTransport 4.x.y systems does not include the SiteMinder administrator password.	Optional
Authorization Port	The authorization port for the SiteMinder Policy Server.	Required
Authentication Port	The authentication port for the SiteMinder Policy Server.	Required
Accounting Port	The accounting port for the SiteMinder Policy Server.	Required
LDAP User Directory	Name of the SiteMinder user directory used to retrieve the home folder, user ID, and group ID.	Optional
Agent Name	The name for the SiteMinder agent that SecureTransport should use when connecting to the SiteMinder Policy Server.	Required
Agent Type	For SiteMinder protocol version 4 the shared secret used to communicate with the SiteMinder Policy Server. For version 5, the path to `SmHost.conf`.	Required
Shared Secret	The password for the SiteMinder agent that SecureTransport uses to connect to the Policy Server.	Required
Maximum Connections	The maximum number of SiteMinder connections that SecureTransport can have open simultaneously. This does not limit the number of users who can log in to the SecureTransport Server using the Site Minder SSO portal.	Required
Connection Timeout	The amount of time (in seconds) that a SiteMinder connection can be idle before it is closed. The default is 30 seconds. This is independent of user session timeout.	Required

Name	Description	Required/ optional
File Storage Root Path	The segment of the absolute URI that is removed before it is submitted to the SiteMinder Policy Server for authorization. If the entire absolute URI is submitted for authorization, type `/` in this field.	Required
SiteMinder Path Prefix	After the File Storage Root Path is removed, but prior to SiteMinder authorization, this entry is prefixed to the absolute URI. For example, if the absolute URI is `/mnt/ab/user1`, the File Storage Root Path is `/mnt/ab`, and the SiteMinder Path Prefix is `/root`; then `/root/user1` is sent to SiteMinder for authorization. If this box is left blank, no prefix is applied to the URI prior to authorization. **Note** When SiteMinder is enabled, all SecureTransport users must have GET access to the path specified in the SiteMinder Path Prefix to successfully log in. If this setting is left blank, then users must have GET access to /. The SiteMinder administrator must set up the SiteMinder Policy Server accordingly.	Optional
Default Home Folder	The absolute URI of the default home folder of the local user. The default home folder is used when a home folder is not supplied by the SiteMinder Policy Server. **Requirements:** • The folder must be created manually on the machine. • On Windows, the folder should not use shared storage. • On Linux, the permissions of the folder must be set to "777" on all nodes in the cluster.	Required
Default Local User ID	The numeric user ID (UID) of a user that has full read/write access to the directory specified as the File Storage Root Path and its subdirectories. This default is used only if a UID is not supplied by SiteMinder. **Note** On Windows, type the name of the respective virtual user. Windows does not support UIDs.	Required

Name	Description	Required/ optional
Default Local Group ID	The numeric group ID (GID) of a user that has full read/write access to the directory specified in the File Storage Root Path and its subdirectories. If no UID or GID are supplied by SiteMinder, these defaults are used for all file operations (including ownership of new files) performed by SecureTransport for users authenticated by SiteMinder.	Required
Explicitly uses SiteMinder Attributes	When selected, SecureTransport uses the values specified in the *User Attribute Names* section. If not selected, and • if the user is assigned to an account template, the User ID, Group ID, and Home Folder are determined from the template. • if the user is not assigned to an account template, the default home folder, local user ID, and local group ID are used. The state of the checkbox has no effect on SiteMinder users mapped as virtual users.	Optional
Home Folder Attribute	SiteMinder returns information about the user as name=value pairs. The attributes define the name part of the pair which must be added manually. In `Home Folder Attribute`, you should provide the name of the SiteMinder attribute which value is the absolute URI of the user's home folder. **Requirements:** • The folder must be created manually on the machine. • On Windows, the folder should not use shared storage. • On Linux, the permissions of the folder must be set to "777" on all nodes in the cluster. **Note** When changing `Home Folder Attribute`, the Transaction Manager should be restarted in order for the changes to be applied. **Note** If `Home Folder Attribute`, `Group ID Attribute` and `User ID Attribute` are left blank, and the attributes therefore not defined, the default Home Folder, Local User ID, and Local Group ID are used.	Optional
User ID Attribute		Optional
Group ID Attribute		Optional

3. Click **Update SiteMinder Settings**.

To use one or two more SiteMinder servers for failover, specify server configuration parameters that correspond to the fields described above. The names of the parameters for the second server start with `Siteminder.PolicyServers.Second.PolicyServer`. The names for the third server start with `Siteminder.PolicyServers.Third.PolicyServer`. The final parts of the names are given in the following table:

Field	Sever configuration parameter
Enable SiteMinder Module	`enable`
IP Address	`host`
Administrator Username	`adminUsername`
Administrator Password	`adminPassword`
Authorization Port	`authorizationPort`
Authentication Port	`authenticationPort`
LDAP User Directory	`ldapUserDirectory`
Maximum Connections	`maxConnections`
Connection Timeout	`timeout`

Note If more than one SiteMinder server is configured in a server that is upgraded from SecureTransport 4.x.y or in configuration that is imported from a SecureTransport 4.x.y server, set the `Siteminder.PolicyServers.Second.PolicyServer.enable` and the `Siteminder.PolicyServers.Third.PolicyServer.enable` system configuration parameters to `true` as required.

There are also parameters for `minConnections` and `connectionsStep` which are not set in the *CA SiteMinder Setting* page.

LDAP integration

You can configure Axway SecureTransport to use Lightweight Directory Access Protocol (LDAP) servers to authenticate users and provide information it uses to set up the user session.

The SecureTransport LDAP integration includes:

- Support for LDAP versions 2 and 3.
- Support for Secure LDAP, also know as LDAP over SSL/TLS or LDAPS.
- Search over multiple LDAP domains that provide authentication information and user attributes for different groups of users.

- Multiple, redundant LDAP servers in a domain for backup when an LDAP server is down or inaccessible.
- One or more default LDAP domains that SecureTransport searches when a user does not specify a domain name on login.

Note You cannot configure both LDAP integration and SiteMinder integration.

The following topics provide LDAP connections, binds, and searches information and logins, agents, domains, home folders, and use type ranges for LDAP:

- *LDAP connections, binds, and searches* on page 476 - Describes LDAP connections, binds, and searches.
- *LDAP logins* on page 476 - Describes LDAP logins and provides how-to instructions for logging into LDAP.
- *LDAP domains* on page 478 - Describes the LDAP domains.
- *LDAP home folders* on page 496 - Describes the LDAP home folders.
- *LDAP user type ranges* on page 498 - Describes the LDAP user type ranges.

LDAP connections, binds, and searches

To configure backup LDAP servers in case a server is not accessible or not responding, you can list two or more LDAP servers for any domain. SecureTransport attempts to connect to the servers and bind to their LDAP databases in the order you specify in the server list. SecureTransport uses the first LDAP database it can bind to. If SecureTransport does not find a record for the user in the first available LDAP database, it does not try to connect to other servers in the sequence. So for each login attempt, SecureTransport searches at most one LDAP database in a domain.

You can configure SecureTransport to bind to an LDAP database anonymously or using a bind DN and password.

To locate a DN in the LDAP database, SecureTransport searches using partial DN information and the user's common name (CN), unique identifier (UID), or Active Directory account name (sAMAccountName). You must define the base DN as required by the server and select the search attribute. You can also define an alias query that is a filter that uses values from an email address used as a login user name.

LDAP logins

If you configure and enable LDAP, SecureTransport uses it as follows when users log in:

- If the user includes a domain name in the login name, *domain_name*/*user_name*, SecureTransport attempts to connect the LDAP servers configured for the named domain. If the first server SecureTransport connect to has record for the user, SecureTransport use it. If not, the login fails.
- If the user does not include a domain name in the login name, SecureTransport can still find the authentication information and user attributes in the LDAP databases of the default domains in the order on the *LDAP Domains* page. Depending on whether or not LDAP authentication is required, SecureTransport also searches other databases:

1. If LDAP authentication is optional, SecureTransport searches the SecureTransport database before it searched the LDAP databases in the default domains.
2. If the user specified by the login name is not in the SecureTransport database and login name and password match, SecureTransport searches the databases of the LDAP servers configured for the default domains in the order on the *LDAP Domains* page.
3. If LDAP authentication is optional and the authentication information is not in any of the databases of the LDAP servers in the default domains, SecureTransport searches the operating system if real users are enabled.

If SecureTransport does not find the user name in one of these locations, the login fails.

When SecureTransport finds an LDAP entry for the user name, it uses the password from the entry to authenticate the user. If authentication fails, the login fails.

- **User ID** (UID) (UNIX-based systems only) – `fdxUid`. This is the numeric value required by the UNIX system to identify the user. This is not the LDAP attribute `UID`, which represents an LDAP unique identifier.
- **Group ID** (GID) – `fdxGid`
- **Home folder** (HomeDir) – `fdxHomeDir`
- **User type** – `fdxUserType`
- **User shell** (UNIX-based systems only) – `fdxShell`
- **System user** (Windows only) – `fdxSysUser`. This is name of a local or domain user of the Windows server. SecureTransport uses this user's credentials to access the Windows files in the session. If this is a real user, you must add the user to a SecureTransport password vault before you specify the user as the System User in an LDAP record or as the default system user for a domain. See *Add a user to a password vault* on page 745.
- **Login by email** – `fdxAuthByEmail`. If this is enabled, the user can login using an email address as well as a user name if the login by email is enabled in the LDAP domain and the email attribute of the LDAP record has the correct value.

If the LDAP record SecureTransport finds does not include some of the user attributes, SecureTransport uses any enabled attributes maps, any enabled user type ranges, any enabled home folder entries, and the configured defaults for the domain to set the attributes. If any required attribute information is not available or not valid, the login fails.

SecureTransport performs the following actions to set the user attributes and other required session information:

1. Sets all attributes from LDAP record values based on any enabled attribute maps. For configuration, see *Define attribute mappings for a domain* on page 490.
2. On UNIX-based systems, if the `fdxUserType` attribute is not set and there is an applicable entry in the *User Type Ranges* page, sets the user type based on the value of the user ID. For configuration, see *LDAP user type ranges* on page 498.
3. For attributes that are not set, applies the default values for the domain. For configuration, see *Define LDAP user settings for a domain* on page 489.
4. Sets the user class. See *User classes* on page 771.

5. Checks any enabled DN filters configured for the domain. You can use DN filters to permit access to only certain sub-trees of the LDAP directory structure within the domain. If there is an enabled DN filter for the user class set in the previous step or for all users, denoted by asterisk (`*`), the DN from the LDAP record must match one of those filters. If there are enabled DN filters for the user class or for all users and no filter matches, the login fails. See *Manage DN filters for a domain* on page 490.
6. If the `fdxHomeDir` attribute is not set, sets it based on the user class using the entries in the *Home Folder* page. For configuration, see *LDAP home folders* on page 496.
7. Use the alphabetically first applicable account template. If there is an applicable account template, the values from the template replace any value already set. For details, see *Account templates and external users* on page 718.

If, after this process, any required user attributes are not set because there is no enabled attribute map, because the LDAP value for an enabled attribute map is not present in the LDAP record, or because the value was not set by a later step, the login fails.

SecureTransport real users authenticated using LDAP have the following limitations:

- They cannot use certificate authentication.
- They cannot change their passwords using a SecureTransport client.
- You can only subscribe them to an application if you do it in an account template or create a SecureTransport account that stores its password in the LDAP record.

To configure Active Directory in a SecureTransport LDAP domain, see *LDAP and Active Directory configuration* on page 496.

LDAP domains

Note If you setup LDAP in Windows without specifying a `sysuser`, the default configuration is applied.

The following topics describe the management and configuration of LDAP domains:

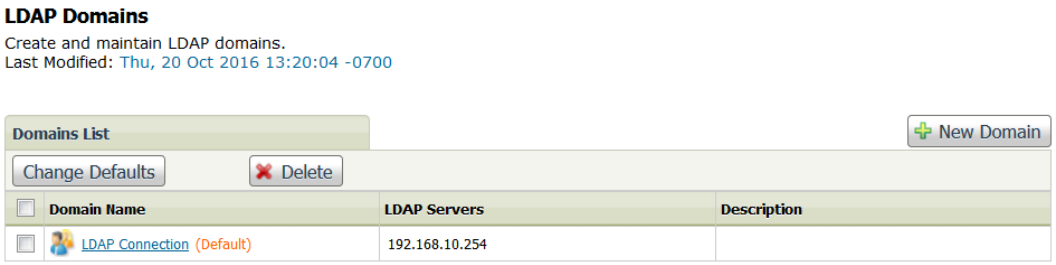
- *Create an LDAP domain* on page 479 - Provides how-to instructions for creating a domain.
- *Define LDAP search criteria for a domain* on page 482 - Provides how-to instructions for defining the LDAP search criteria for a domain.
- *Define LDAP user settings for a domain* on page 489 - Provides how-to instructions for defining LDAP user settings for a domain.
- *Define attribute mappings for a domain* on page 490 - Provides how-to instructions for defining domain attribute mappings.
- *Manage DN filters for a domain* on page 490 - Describes managing the DN filters for a domain.
- *Manage DN filters* on page 491 - Provides how-to instructions for managing DN filters.
- *Define Address Book settings for a domain* on page 492 - Provides how-to instructions for defining LDAP Address Book searches and Address Book attributes for a domain.
- *Edit a domain* on page 494 - Provides how-to instructions for editing a domain.
- *Delete domains* on page 494 - Provides how-to instructions for deleting domains.

- *Configure default domains* on page 494 - Provides how-to instructions for configuring default domains.
- *LDAP domains example* on page 495 - Provides examples of LDAP domains.
- *Secure LDAP* on page 495 - Describes securing the LDAP connection.
- *LDAP and Active Directory configuration* on page 496 - Describes using LDAP with an Active Directory configuration.

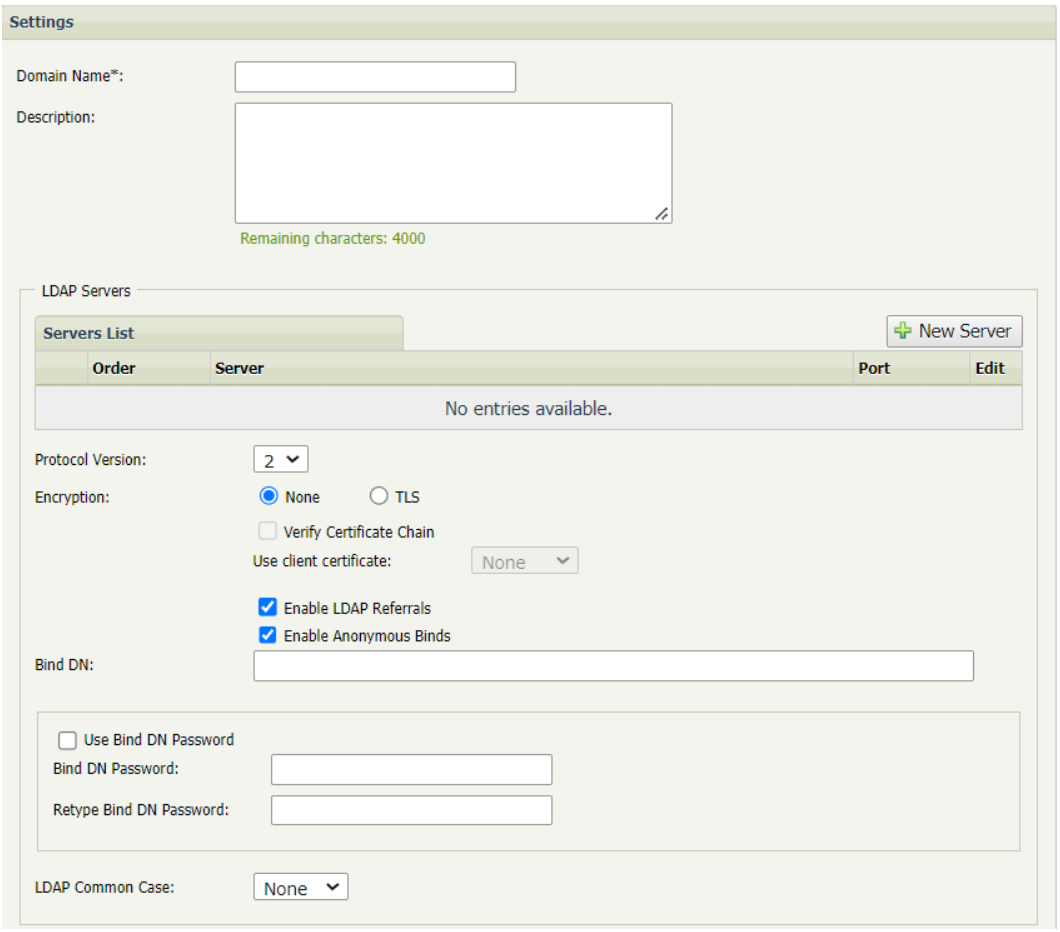
Create an LDAP domain

When you create a domain, you must give it a name and specify at least one LDAP server.

1. Navigate to **Authentication > LDAP Domains**.



2. Click **New Domain**.



3. Enter a **Domain Name**. The user must specify this name in the login name, *domain_name*/*user_name*, to select this domain if it is not configured as a default domain on the *LDAP Domains* page.
4. Enter a **Description**.
5. Under *LDAP Servers*, click **New Server**.

   A line is added to the *Servers List*.
6. In the **Server** field, enter the host name or IP address of the LDAP server.
7. In the **Port** field, enter the LDAP port number for the server. The default is 389.
8. To test the connection to the LDAP server, click the icon ( ) before the port number.
9. Click the **Save** icon ( ) in the **Edit** column.
10. Add backup LDAP servers, if any, to the *Servers List* by repeating the previous steps.
11. To change in which order SecureTransport tries the servers, click **Reorder**, update the numbers in the **Order** column, and click **Save**.
12. Under *LDAP Servers*, complete the following fields:

Field	Description	Valid values and notes
Protocol Version	Select the LDAP protocol version.	Values: 2; 3
Encryption	Enable Secure LDAP, also known as LDAP over SSL/TLS or LDAPS.	Values: None; TLS; StartTLS (for LDAP protocol 3)
Verify Certificate Chain	Configure whether SecureTransport implicitly trusts the LDAP servers in this domain or verifies the LDAP server certificates.	See *Secure LDAP on page 495*.
Use client certificate	To authenticate to the LDAP server or a reverse proxy with a client certificate, set the **Encryption** to *TLS* and select a certificate from the drop-down. If mutual authentication is required, you need to enable the **Verify Certificate Chain** option as well.	Values: None; certificates from the Local Certificates store represented by their aliases (e.g., `admind`) By default, the value is set to *None* and client certificate authentication is disabled.
Enable LDAP Referrals	Configure whether SecureTransport allows the LDAP server to refer a request to another LDAP server.	This option is required when the LDAP directory tree is distributed over a group of servers.
Enable Anonymous Binds	Configure whether SecureTransport uses a Bind DN to access the LDAP server.	You can select this option when LDAP servers support anonymous binding. If this option is not selected, the **Bind DN** field is required.
Bind DN	Enter the distinguished name of a user who is allowed access to the LDAP directory for user lookups.	For authorization purposes, this field is case-sensitive. If **Enable Anonymous Binds** is not selected, this field is required.

Field	Description	Valid values and notes
Use Bind DN Password	If a password is required to bind to the directory service on the LDAP server, select **Use Password** and enter the password in the fields provided.	
LDAP Common Case	Configure whether and how SecureTransport changes the case of the user name it receives from the LDAP database.	Values: None; Lower; Upper. If the value is *Lower* or *Upper*, SecureTransport maps the case of all letters in the user name to the specified case.

13. Click **Save.**

Note The *Address Book Settings* pane is displayed only if the Address Book feature is enabled (the value of the `AddressBook.Enabled` configuration option is set to **true**). For more information, refer to *Define Address Book settings for a domain on page 492* and *LDAP source on page 243*.

Define LDAP search criteria for a domain

For information about how SecureTransport uses the search criteria, see *LDAP connections, binds, and searches* on page 476.

Note A specific LDAP configuration is required for Active Directory. For details, see *LDAP and Active Directory configuration* on page 496.

If you do not have the *New LDAP Domain* page open, select **Authentication > LDAP Domains** and click the domain name in the Domains List to open the *LDAP Domain* page.

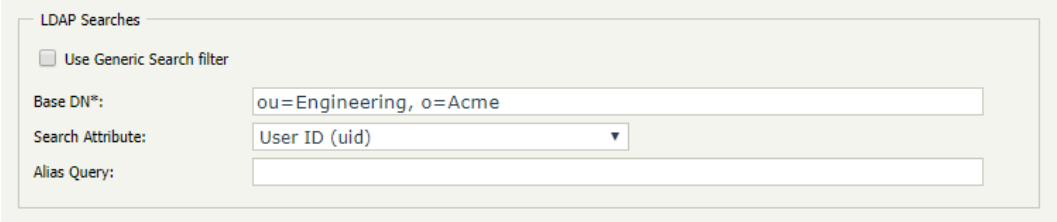
Under *LDAP Searches*, there are two interchanging ways to proceed:

- do not use the **Generic LDAP Search filter** (default) – uses the Alias Query for LDAP search
- use the **Generic LDAP Search filter** check-box – enables two options:
  - Generic Search Filter by any LDAP attribute without appending
  - Generic Search Attribute used for user configuration mapping

For more information, see the dedicated subtopics that follow.

Do not use Generic LDAP search

Add the LDAP search without selecting the **Use Generic Search filter** check-box.



Fill in the search criteria as described below:

Field	Description	Valid values and notes
Base DN	Define the base DN for the searches	A valid DN, such as, `OU=Sales`, `DC=ldaps1`, `DC=Example`, `DC=com`
Search Attributes	Select the LDAP attribute to use for the searches	User ID (`uid`), Common Name (`cn`), or SAM-Account-Name (`sAMAccountName`)
Alias Query	Define a filter using values from an email address used as a login user name	The query to be used with the LDAP search. Read further this subtopic for additional info on syntax and usage.

Once you are done, click **Save**.

The **Alias Query** field is used to perform real user look-up by email address to filter (limit) the search. If the **Alias Query** filter is selected, the search is performed not only by email but by email OR the specified filter in the **Alias Query** field. The real user look-up also returns only PERSON entities and no other object classes. So, you do not have to add attribute (`objectClass=Person`) in the field because SecureTransport inserts it in the filter. The value of the **Alias Query** field uses the search filter syntax described in *RFC 4515: Lightweight Directory Access Protocol (LDAP): String Representation of Search Filters (June 2006)*, (http://www.rfc-editor.org/rfc/pdfrfc/rfc4515.txt.pdf).

The basic syntax of a search filter is:

`(attribute search_operator value)`

For example:

`(buildingname>=alpha)`

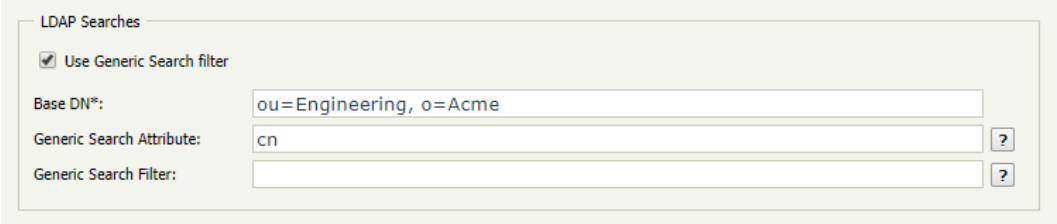
In this example, buildingname is the attribute, `>=` is the operator, and `alpha` is the value. You can also define filters that use different attributes combined together with logical operators.

Use Generic LDAP search filter

Select the **Use Generic Search filter** check-box and observe the following changes in the *LDAP Searches* options:

- the **Search Attribute** drop-down list is replaced with the **Generic Search Attribute** input text box
- **Alias Query** input text box is replaced by **Generic Search Filter** input text box

You can add any Generic Search attribute to use in the Generic search filter query. Note that the Generic LDAP search is used as it is and no additional attributes are appended.



Fill in the search criteria as described below:

Field	Description	Valid values and notes
Base DN	Define the base DN for the searches	A valid DN, such as, `OU=Sales, DC=ldaps1, DC=Example, DC=com`
Generic Search Attribute	Specify any LDAP property for user configuration mapping	Example values include: Common Name (`cn`), `displayName`, `givenName`, `sAMAccountName`, User ID (`UID`) Use the tooltip for example usage.
Generic Search Filter	Specify the query that will be used for performing LDAP search	Basic prefix logical operators (&, |, !) can be used. SecureTransport allows referencing the following values by replacing them: • %s – Complete email address. (SecureTransport replaces %s in the value with the complete email address.) • %u – User name. (SecureTransport replaces %u with the user name.). Use the tooltip to see the correct query syntax and the accepted values.

Once you are done, click **Save**.

Note The Generic LDAP search filter can access the Plugin Context, part of the `PluginUserData` object. Values from this plugin context are accessible using the syntax as in the following example: `(mail=%subjectAltNameEmail)`.
As `subjectAltNameEmail` is the key and if its value is "someemail@mail.com" the generic search filter evaluates to `mail=someemail@mail.com`

The following topics provide basic prefix logical operators, attribute names, search filter operators, special characters in search filters, special values, and LDAP configuration example using the Alias Query filter:

- *Basic prefix logical operators* on page 485
- *Attribute names* on page 485
- *Search filter operators* on page 486

- *Special characters in search filters* on page 487
- *Special values* on page 487
- *LDAP configuration using the Alias Query filter* on page 487

Basic prefix logical operators

Operator	Symbol	Description
AND	&	All specified filters must be true for the statement to be true. For example: `(&(filter)(filter)(filter)...)`
OR	|	At least one specified filter must be true for the statement to be true. For example: `(|(filter)(filter)(filter)...)`
NOT	!	The specified statement must not be true for the statement to be true. Only one filter is affected by the NOT operator. For example: `(!(filter))`

Logical expressions are evaluated in the following order: - Innermost to outermost parenthetical expressions first - All expressions from left to right. So, you can use parentheses (and) to specify the order of operations.

For example:

The following example returns users that match attribute `objectClass` value with `Person` and attribute `cn` matches `Babs J*` where `*` means matching of zero or more characters.

```
(&(objectClass=Person)(cn=Babs J*))
```

The following example returns entries containing attribute values that do not match the specified value.

```
(!(cn=Tim Howes))
```

Attribute names

Attribute names depend on the type of LDAP server. For Active Directory servers where the standard alias attribute is `proxyAddresses` an example for the filter is `(proxyAddresses=smtp\3A%s)`. For other LDAP servers an example can be `(|(cn=%u)(mail=%s))`. This will find all users with given primary email address or common name.

Examples for attributes:

uid User ID
cn Common Name
sn Surname
l Location
ou Organizational unit

- o Organization
- dc Domain Component
- st State
- c Country

Search filter operators

Search type	Operator	Description
Equality	=	Returns entries containing attribute values that exactly match the specified value. For example: `cn=Bob Johnson`

Substring	`=string*string`	Returns entries containing attributes containing the specified substring.
		For example:
		`cn=Bob*`
		`cn=*Johnson`
		`cn=*John*`
		`cn=B*John`
		The asterisk (`*`) indicates zero (0) or more characters.

Greater than or equal to	>=	Returns entries containing attributes that are greater than or equal to the specified value. For example: `buildingname >= alpha`
Less than or equal to	<=	Returns entries containing attributes that are less than or equal to the specified value. For example: `buildingname <= alpha`

Presence	=*	Returns entries containing one or more values for the specified attribute.
		For example:
		`cn=*`
		`telephonenumber=*`
		`manager=*`

Special characters in search filters

Use backslash (`\`) followed by two hexadecimal digits to specify any special character or non-ASCII UTF-8 characters.

Special character	Value with special character	Example filter
*	`Five*Star`	`(cn=Five\2aStar)`
\	`c:\File`	`(cn=c:\5cFile)`
()	`John (2nd)`	`(cn=John \282nd\29)`
NUL	`0004`	`(bin=\00\00\00\04)`
non-ASCII UTF-8 characters		`sn=Lu\c4\8di\c4\87)`

Special values

SecureTransport enables to reference the following values by replacing them:

`%s` – Complete email address. (SecureTransport replaces `%s` in the value with the complete email address.)

`%u` – User name. (SecureTransport replaces `%u` with the user name.)

`%d` – Domain name. (SecureTransport replaces `%d` with the domain.)

Examples:

```
(proxyAddresses=smtp\3A%s) - proxyAddresses
```

Active Directory attribute must be smtp: followed by the email address.

```
(proxyAddresses=smtp:testuser@domain.com) (|(cn=%u)(mail=%s))
```

Either the cn attribute is the user name or the mail attribute is the email address.

```
(|(cn=testuser)(mail=testuser@domain.com))
```

LDAP configuration using the Alias Query filter

1. Enable `Login by Email` in the *LDAP User Settings* pane on the *LDAP Domain Settings* page.
2. For two LDAP users a1 and b1, having the same settings, add new LDAP attributes:
   - For user a1 add attribute: `mobile = a1@st1.lab.sofi.axway.int`
   - For user b1 add attribute: `mail = b1@st1.lab.sofi.axway.int`
3. The users exist on same domain in SecureTransport. So, create two new attributes in the *Attributes List* section on the *LDAP Domain Settings* page and map them to the LDAP attributes:

SecureTransport attribute name	LDAP attribute name
fdxAuthByEmail	mail
fdxAuthByOwn	mobile

4. Alias Queries examples:
   - No matter what search criteria the administrator has to set in the **Alias Query** field, the LDAP user can always authenticate using his `uid` (User ID), `cn` (Common Name) and `sAMAccountName` (SAM-Account-Name) search attributes.
   - Using an `&` operation joins all filters together and all conditions should be true.
   - Using and `|` requires at least one true condition.
   - If we filter an e-mail with `%u` (username) only, the user with this attribute will be allowed to login only by providing a username. Try the following queries:
     - `(mail=%u)` - the result filter will be `(&(objectClass=Person)(mail=%u))`. User b1 has mail attribute which represents the full e-mail. So, it is different from the username (`%u`) and e-mail login for user b1 will fail.
     - `(&(mobile=%u)(mail=%u))` - the result filter will be `(&(objectClass=Person)(&(mobile=%u)(mail=%u)))`. User b1 has mail attribute and user a1 has mobile attribute different from the username, so e-mail login for users a1 and b1 will fail.
   - If we filter an e-mail with `%s` (full e-mail address) only, the user with this attribute will be allowed to login by providing both a username or a password. Try the following queries:
     - If **no query** is set, the result filter will be `(&(objectClass=Person)(mail=%s))`. The mail authentication of user a1, with only mobile attribute defined, fails, because we haven't set an e-mail login permissions for the mobile attribute in the search filter. Mail authentication for b1 is successful because by default SecureTransport searches by e-mail with filter `(mail=%s)`.
     - `(mobile=%s)` - We haven't provided additional filtering for mail attribute. The result filter will be `(&(objectClass=Person)(|(mobile=%s)(mail=%s))`. E-mail login for user b1 will be successful. In the filter we have `(mail=%s)` and user b1 have mail attribute references the full e-mail address. For user a1 we have mobile attribute which references the full e-mail address, so e-mail login for a1 will also be successful.
     - `(&(mail=%u)(mobile=%s))` - the result filter will be `(&(objectClass=Person)(&(mail=%u)(mobile=%s)))`. Mail attribute in the filter allows only username login, so e-mail authentication for user b1 which mail attribute references the full e-mail address will fail. Mobile attribute in the filter allows full e-mail authentication, so for user a1 the e-mail authentication will be successful.

- Filter with `%d` (domain name) identifies domain name:
  - `(mobile=%u@%d) equals (mobile=%s)` - the result filter will be `(&(objectClass=Person)(|(mobile=%u@%d)(mail=%s)))`. The search filter allows user a1 with mobile attribute to authenticate using the full e-mail. User b1 will be also authenticated successfully with his mail attribute.
  - `(|(mobile=%u@%d)(mail=%u))` - the result filter will be `(&(objectClass=Person)(|(mobile=%u@%d)(mail=%u)))`. The search filter allows user a1 with attribute mobile to be authenticated by full e-mail address and denies user b1 authentication with the full e-mail.
- If we have two different mail addresses for one account - for example: attribute `mail = x@a.b` and attribute `mobile = y@a.b` the login will be successful by default for `x@a.b`, unless we define the following query `(mobile=%s)`.

Note In Active Directory the `proxyAddress` is the only possible property that we can use for mail attribute (after the primary mail), so we can use filtering with the following query: `(proxyAddresses=smtp\3A%s)`.

Define LDAP user settings for a domain

SecureTransport uses these default values when the LDAP entry does not include the attribute and no mapping to the attribute is enabled in the *Default Attributes List*. For information, see *LDAP logins* on page 476.

1. If you do not have the *New LDAP Domain* page open, select **Authentication > LDAP Domains** and click the domain name in the Domains List to open the *LDAP Domain* page.
2. Under *LDAP User Settings*, complete the following fields:

Field	Description	Valid values and notes
Default GID	The value for the `fdxGid` attribute (group ID) for the LDAP user.	The default value is 10000.
Default UID	The value for the `fdxUid` attribute (user ID) for the LDAP user.	UNIX-based systems only. The default value is 10000.
Default User Shell	The value for the `fdxShell` attribute (user shell) for the LDAP user.	UNIX-based systems only. Valid user shell. The default value is `/bin/sh`.
Default User Type	The value for the `fdxUserType` attribute (user type) for the LDAP user.	`Real` or `Virtual`. The default value is `Virtual`.
Allow login by email	Controls whether the user can log in using an email address stored in the email attribute of the LDAP record.	`Enabled` or `Disabled`. The default value is `Disabled`.

3. Click **Save**.

Define attribute mappings for a domain

For information about how SecureTransport uses the default attribute mappings, see *LDAP logins* on page 476.

The session variables available depend on the attribute mappings:

- The following session variables are always available: `STSESSION_LDAP_AUTH_BY_EMAIL`, `STSESSION_LDAP_DN`, `STSESSION_LDAP_DOMAIN_ID`, and `STSESSION_LDAP_DOMAIN_NAME`.
- To enable `STSESSION_LDAP_fdxGid`, `STSESSION_LDAP_fdxHomeDir`, `STSESSION_LDAP_fdxShell`, `STSESSION_LDAP_fdxUid`, and `STSESSION_LDAP_fdxUserType`, select **Map to Schema** for the corresponding default attribute.
- If you do not select **Map to Schema** for any custom mappings, all LDAP attributes are mapped to session variables named `LDAP_DIR_` followed by the attribute name.
- If you add a custom mapping, only those attributes added with **Map to Schema** selected are mapped to session variables named `LDAP_DIR_` followed by the attribute name.

A multivalued LDAP attribute is mapped to several session variables. To use a multivalued LDAP variable, map it and check the SecureTransport session for the names of the session variables.

1. If you do not have the *New LDAP Domain* page open, select **Authentication > LDAP Domains** and click the domain name in the Domains List to open the *LDAP Domain* page.
2. Under *Attributes List*, for each SecureTransport attribute that will be mapped from an LDAP attribute, select **Map to Schema** to enable an attribute mapping.

You can modify a default attribute mapping.

1. Click the Edit icon () in the **Edit** column.
2. Type the new value in the **LDAP Attribute Name** column.
3. Click the Save icon () in the **Edit** column.

You can define a mapping for a custom LDAP attribute.

1. Click **New Attribute**.

   SecureTransport adds a line to the *Attributes List*.
2. Type the **Description**, **ST Attribute Name**, and **LDAP Attribute Name**.
3. Click the Save icon () in the **Edit** column.
4. Select **Map to Schema** to enable the mapping.

To delete a custom attribute mapping, click **X** in the first column of the table.

Manage DN filters for a domain

SecureTransport uses DN filters to accept users with matching DNs. For details, see *LDAP logins* on page 476.

Manage DN filters

The following topics describe how to manage DN filters:

Add a DN filter

Use the following procedure to add a DN filter.

1. If you do not have the *New LDAP Domain* page open, select **Authentication > LDAP Domains** and click the domain name in the Domains List to open the *LDAP Domain* page.
2. Under *DN Filter List*, click **New Filter**.

   A line is added to the *DN Filters List*.
3. In the **DN Filter** field, type a regular expression to match against the DN retrieved from the LDAP database. To specify only a portion of a DN, use wild cards. For example, to allow access to users from the organization acme, enter `.*O=acme.*` in this field. For more information about regular expressions supported by SecureTransport, see *Regular expressions* on page 1117.
4. In the **User Class** field, select a user class to apply the DN filter only to users in that class or asterisk (`*`) to apply the DN filter to all users.
5. Click the Save icon ( ) in the **Edit** column.

The status of the new DN filter is Disabled.

Enable or disable a DN filter

Use the following procedure to enable or disable a DN filter.

- In the entry in the *DN Filter List*, click **Enable** or **Disable**.

  The Status column is updated.

Edit a DN filter

Use the following procedure to edit a DN filter.

1. In an entry in the *DN Filter List*, click the Edit icon ( ) in the **Edit** column.
2. Make the required changes to the fields in the entry.
3. Click the Save icon ( ) in the **Edit** column.

Delete a DN filter

Use the following procedure to delete a DN filter.

- In the entry in the *DN Filter List*, click **X** in the first column.

  The entry is removed from the list.

Define Address Book settings for a domain

For information on LDAP Address Book sources, refer to *LDAP source* on page 243.

Note The *Address Book Settings* pane is only displayed if the Address Book feature is enabled (the value of the `AddressBook.Enabled` configuration option is set to **true**).

The following topics define Address Book LDAP searches and attribute mappings for a domain:

Define Address Book LDAP searches for a domain

Use the following instructions to configure LDAP search settings for the Address Book feature.

Note All search operations performed via this source will be case insensitive and wild card searches will be supported at the end of the phrase; for example, `(|(displayName=string*)(ou=string*)(mail=string*))`.

1. If you do not have the *New LDAP Domain* page open, select **Authentication > LDAP Domains** and click **New Domain** open the *New LDAP Domain* page.

2. In the *Address Book Settings* pane under *LDAP Searches*, complete the following fields:

Field	Description	Valid values and notes
Base DN	Define the base DN for the searches	A valid DN, such as, `OU=Sales, DC=ldaps1, DC=Example, DC=com`

Additional search query	Enter an LDAP query to specify the selection criteria for Address Book.
	The search behavior depends on the current selection of the **Use only additional search query** checkbox:
	• If selected, SecureTransport executes the exact search query entered in the **Additional search query** field.
	• If not selected, SecureTransport applies pre-defined filters that manage the LDAP server responses to the search query, entered in the **Additional search query** field, and executes the final query.
		Get all user entries with an email attribute and a surname equal to "smith":
		`&(sn=smith) (objectClass=user) (email=*)`
		Get all entries:
		`objectclass=*`

3. Click **Save**.

Define Address Book attribute mappings for a domain

Note Address Book supports only unique group names, so the LDAP server should not have two group entries with exact same value of the attribute which, is mapped to `displayName`.

Note Address Book classifies an entry as user or group based on the `objectClass` and `objectCategory` attribute values.

To map an Address Book attribute to a schema:

1. If you do not have the *New LDAP Domain* page open, select **Authentication > LDAP Domains** and click **New Domain** open the *New LDAP Domain* page.
2. Under *Address Book Attributes List*, for each Address Book attribute that will be mapped from an LDAP domain attribute, select **Map to Schema** to enable an attribute mapping.

You can modify a default attribute mapping.

1. Click the Edit icon ( ) in the **Edit** column.
2. Type the new value in the **LDAP Attribute Name** column.
3. Click the Save icon ( ) in the **Edit** column.

You can define a mapping for a custom Address Book attribute.

1. Click **New Attribute**.

   SecureTransport adds a line to the *Attributes List*.
2. Type the **Description**, **Entity Attribute Name**, and **LDAP Attribute Name**.
3. Click the Save icon () in the **Edit** column.
4. Select **Map to Schema** to enable the mapping.

To delete a custom attribute mapping, click **X** in the first column of the table.

Edit a domain

Use the following procedure to edit a LDAP domain.

1. Select **Authentication > LDAP Domains**.

   The *LDAP Domains* page is displayed.
2. Click the domain name in the Domains List.

   The *LDAP Domain* page is displayed.
3. Make the required changes.
4. Click **Save**.

Delete domains

Use the following procedure to delete domains.

1. Select **Authentication > LDAP Domains**.

   The *LDAP Domains* page is displayed.
2. Select the domains to delete in first column of the Domains List.
3. Click **Delete**.
4. Confirm the deletion.

   The selected domains are removed from the Domains List.

Configure default domains

For information about how SecureTransport uses the default domains, see *LDAP logins* on page 476.

1. Select **Authentication > LDAP Domains**.

   The *LDAP Domains* page is displayed.
2. Click **Change Defaults**.

   The Default column is displayed.
3. Click **Toggle** in the Default column to add a domain to or remove a domain from the default domains.

   The default domains are moved to the top of the Domains List and indicated in the Domain Name column.

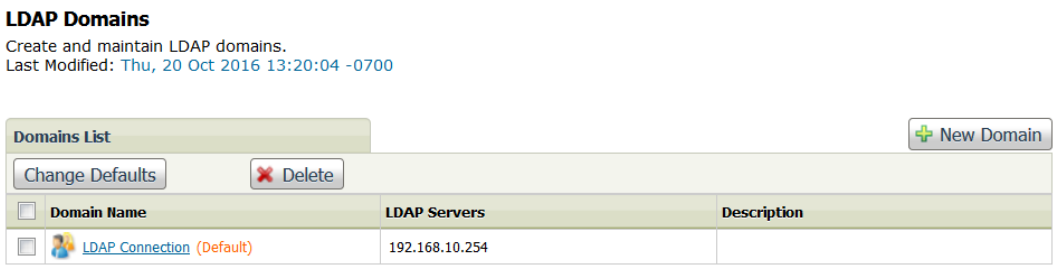
4. If you specify two ore more default domains, up and down arrows are displayed in a column before the Domain Name column.
5. Using the arrows, you can drag the default domain rows in the Domains List to the order you want SecureTransport to search the domains.
6. When the default domains are correct, click **Save Defaults**.

   The Default column and arrow columns are removed.

LDAP domains example

This example illustrates a scenario with three LDAP domains, one each for Engineering, Operations, and Security. The configuration has the following features:

- Each domain has a primary LDAP server and backup LDAP server.
- The Engineering and Operations domains are default domains. Users with login credentials in those LDAP databases do not need to specify a domain to log in.
- The Engineering domain is searched first. Users who have the same login credentials in both the Engineering and Operations domain will get the attributes stored in the entry in the Engineering domain unless they specify the Operations domain when they log in.
- The Security domain is not a default domain. Users with login credentials in the Security domain LDAP database must specify a domain name to log in.



Secure LDAP

For secure communication between SecureTransport and an LDAP server to work correctly, a trust must be established between the two parties.

When **Verify Certificate Chain** is selected, SecureTransport must trust the CA certificates used to sign the LDAP servers' certificate for encrypted connections. You must add these certificates to the SecureTransport trusted certificate store. For more information, see *Import a local certificate* on page 53.

SSL and TLS support have the following limitations based on the SSL protocol and TLS LDAP server implementation:

- SecureTransport cannot connect to the i-Planet Directory Server, v5.0 using TLS. SecureTransport fails to connect after a few minutes, displays an error message in the client, and makes an entry in its server log.
- The OpenLDAP server might incorrectly report an error when closing a TLS connection. The TLS connection closes properly even though the error is reported.

LDAP and Active Directory configuration

If you use LDAP with Active Directory, you must consider the following requirements for the LDAP server configuration in SecureTransport:

- Under LDAP Searches:
  - Specify a CN value in the **Base DN** field (for example, `cn=users`). The CN value is required for SecureTransport LDAP authentication to work, even though the OpenLDAP `ldapsearch` command generically does not require it when run from a command line interface.
  - Select **Common Name (cn)** or **SAM-Account-Name (sAMAccountName)** for the **Search Attribute**. Use the SAM-Account-Name parameter instead of the CN parameter to log in using the Windows domain login name.
- Disable **Anonymous Binds**.

LDAP home folders

You can define entries in the *Home Folder* page that SecureTransport uses to set the home folder (`fdxHomeDir` attribute) for an LDAP user when the attribute is not set by the other actions listed in *LDAP logins* on page 476. If there is an entry for the user's user class or for all users, SecureTransport uses the configured prefix. For example, if the prefix is `/home/users/partners` and the user name is `suplco`, SecureTransport set the home folder to `/home/users/partners/suplco`.

When SecureTransport is running under Windows, you can use a local file path, such as `D:\home\users\partners` or a UNC path for a share such as `\\NAS2\home\users\partners`. The permissions for the share must permit the SecureTransport Administration Tool service, which runs on Windows with a local system user as its owner, to create the folder. If the permissions granted for the share are not sufficient to create the subfolder for the LDAP user's home folder, SecureTransport refuses the connection.

Note Because operating systems do not accept angle brackets (`< >`) and quotation marks (`"`) in file names, LDAP users with any of those characters in their user name cannot log in to SecureTransport and get a default home directory. You must map such users to a properly configured account template.

You can define a user class based on values from the LDAP entry. See *User classes* on page 771.

If there is no entry for the user class, SecureTransport uses the entry for all users indicated by an asterisk (`*`) in the **User Class** field.

For more information about how SecureTransport uses the entries on the *Home Folder* page during LDAP logins, see *LDAP logins* on page 476 In particular, if there is an applicable account template, the home folder defined in the account template replaces any home folder set from configuration on the *Home Folders* page.

The following topics describe managing LDAP home folders:

- *Create a home folder entry* on page 497 - Provides how-to instructions for creating a home folder entry.
- *Enable or disable home folder entries* on page 497 - Provides how-to instructions for enabling or disabling home folder entries.

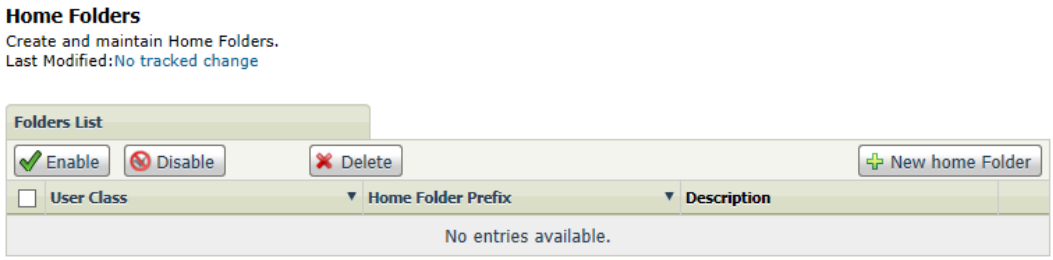
- *Edit a home folder entry* on page 497 - Provides how-to instructions for editing a home folder entry.
- *Delete home folder entries* on page 498 - Provide how-to instructions for deleting home folder entries.

Create a home folder entry

Use the following procedure to create a home folder entry.

1. Select **Authentication > Home Folders**.

   The *Home Folders* page is displayed.



2. Click **New home Folder**.

   A line is added to the *Folders List*.

3. In the **User Class** field, select a user class to apply the home folder prefix only to users in that class or asterisk (*) to apply the home folder prefix to all users.
4. In the **Home Folder Prefix** field, type the path name for the prefix.
5. In the description field, type any notes you want to record about the entry.
6. Click the Save icon ( ) in the **Edit** column.

The status of the new entry is disabled.

Enable or disable home folder entries

Use the following procedure to enable or disable home folder entries.

1. In the first column of the *Folders List*, select the entries to change.
2. Click **Enable** or **Disable**.

   The status icon in the **User Class** column is updated.

Edit a home folder entry

Use the following procedure to edit a home folder entry.

1. In an entry in the *Folders List*, click the Edit icon ( ) in the **Edit** column.
2. Make the required changes to the fields in the entry.
3. Click the Save icon ( ) in the **Edit** column.

Delete home folder entries

Use the following procedure to delete home folder entries.

1. In the first column of the *Folders List*, select the entries to delete.
2. Click **Delete**.
3. Confirm the deletion.

   The entries are removed from the list.

LDAP user type ranges

On UNIX-based systems, you can define entries in the *User Type Ranges* page that SecureTransport uses to set the user type for an LDAP user when the user type is not set by the other actions listed in *LDAP logins* on page 476. You specify a range of values for the user ID (UID) and the user type for SecureTransport to assign to users with values in that range. Every LDAP user on a UNIX-based system has a user ID. See *LDAP logins* on page 476 for the actions that set the user ID.

The following topics provide how-to instructions for managing LDAP user type ranges:

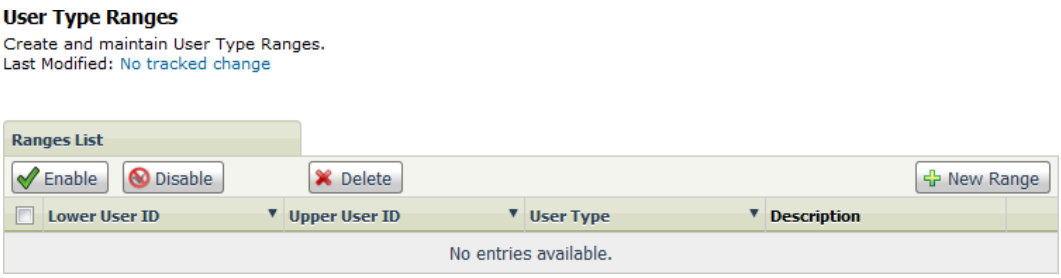
- *Create a user type range entry* on page 498 - Provides how-to instructions for creating a user type range entry.
- *Enable or disable user type range entries* on page 499 - Provides how-to instructions for enabling or disabling user type range entries.
- *Edit a user type range entry* on page 499 - Provides how-to instructions for editing a user type range entry.
- *Delete user type range entries* on page 499 - Provides how-to instructions for deleting user type range entries.

Create a user type range entry

Use the following procedure to create a user type range entry.

1. Select **Authentication > User Type Ranges**.

   The *User Type Ranges* page is displayed.

2. Click **New Range**.

   A line is added to the *Ranges List*.

3. Type the **Lower User ID** and the **Upper User ID**.
4. Select a **User Type**.
5. Click the Save icon () in the **Edit** column.

   The entry is added to the to the *Ranges List*.

The status of the new entry is disabled.

Enable or disable user type range entries

Use the following procedure to enable or disable user type range entries.

1. In the first column of the *Ranges List*, select the entries to change.
2. Click **Enable** or **Disable**.

   The status icon in the **Lower User ID** column is updated.

Edit a user type range entry

Use the following procedure to edit a user type range entry.

1. In an entry in the *Ranges List*, click the Edit icon () in the **Edit** column.
2. Make the required changes to the fields in the entry.
3. Click the Save icon () in the **Edit** column.

Delete user type range entries

Use the following procedure to delete user type range entries.

1. In the first column of the *Ranges List*, select the entries to delete.
2. Click **Delete**.
3. Confirm the deletion.

   The entries are removed from the list.

Accounts 11

Use the pages of the **Accounts** menu to create and manage login accounts for users, administrators, and partners. In order for login by email to function properly, all user accounts must be assigned unique email addresses. Additionally, the client password reset feature will not work if emails assigned to users accounts are not unique.

Types of Accounts

An Axway SecureTransport *account* contains information about a user or an internal system that processes SecureTransport file transfers. SecureTransport supports two types of accounts: *user* and *service*.

User accounts

A *user account* is typically external to your enterprise. It consists of settings such as the account name, contact information, login information, *subscription* information (the *applications* this user account uses to process file transfers), *transfer site* information, and *certificate* information. User accounts connect to SecureTransport through subscriptions to one or more applications. For details, see *User accounts on page 501*.

Service accounts

A *service account* is used to represent processes on systems internal to your enterprise.

In contrast to user accounts, service accounts are internal to the system and function only with the Standard Router application that creates a connection and transfers files between a service account (internal system) and a standard user account. See *Standard Router application on page 850*.

Service accounts do not subscribe to applications using subscriptions; instead, they are referenced in the application configuration. For this reason, the *Service Accounts* page does not contain a **Subscriptions** tab. Apart from that, service accounts are defined and managed in almost the same way as user accounts: the account settings, per-account transfer sites definition, and per-partner certificates are defined and managed identically to those of user accounts.

Subscriptions, transfer sites, and certificates

A subscription defines how files are submitted to and received from applications. For details, see *Manage subscriptions on page 664*.

A transfer site is a location such as a local folder or protocol server used by SecureTransport to pull data from or send data to during a transfer. For details, see *Transfer sites on page 540*.

A certificate can be one of three types: login, partner, or private. Each certificate type can be used by the account for different purposes. For details, see *Manage login certificates on page 527*.

Applications

An application is an instance of an *application type*. An application type is a workflow definition that is triggered by either data arrival or a scheduled event. An application is created when you configure an application type, name it and save it. For more information, see *Applications on page 817*.

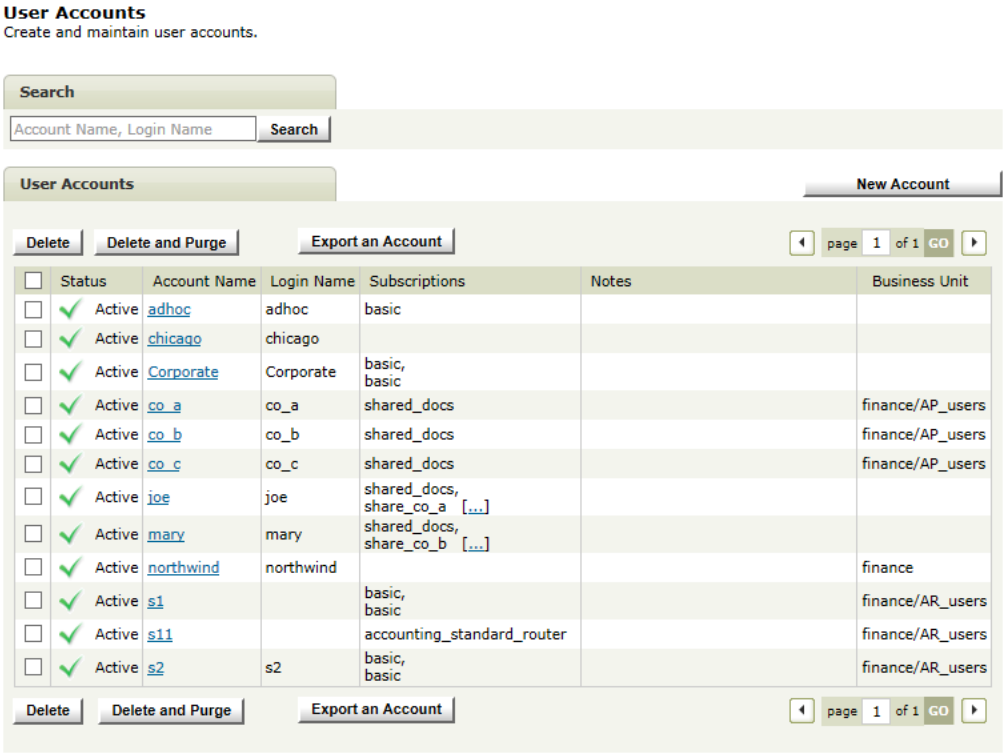
User accounts

Use the *User Accounts* page to display, create, modify, and delete user accounts.

See available user accounts

The *User Accounts* page displays a list of accounts. You can display a list of all user accounts, or search the list based on account name or login name.

- Select **Accounts > User Accounts**. The *User Accounts* page is displayed.



The *User Accounts* page lists 100 account entries per page by default.

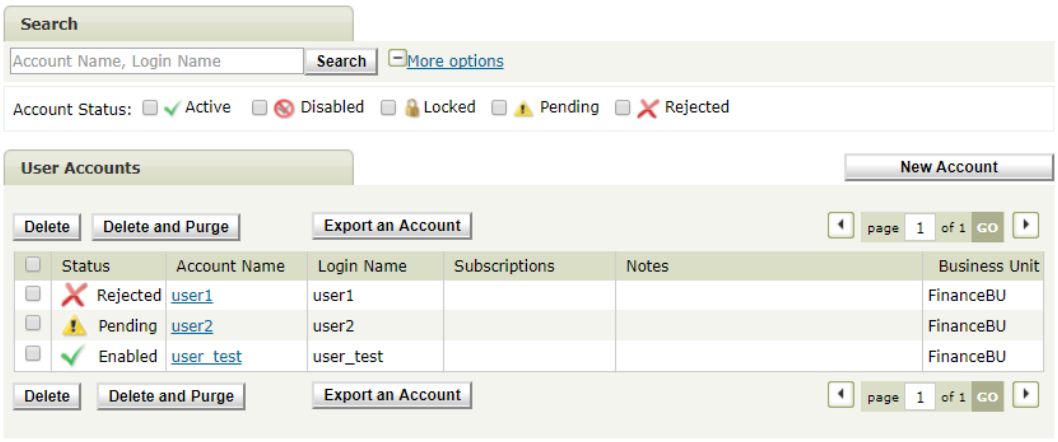
Note You can change the default number of records per page by editing the `LinesPerPage` parameter in the file `<FILEDRIVEHOME>/tomcat/admin/webapps/coreadmin/WEB-INF/web.xml`.

Search for a user account

You can search the user accounts database and display the results on the *User Accounts* page.

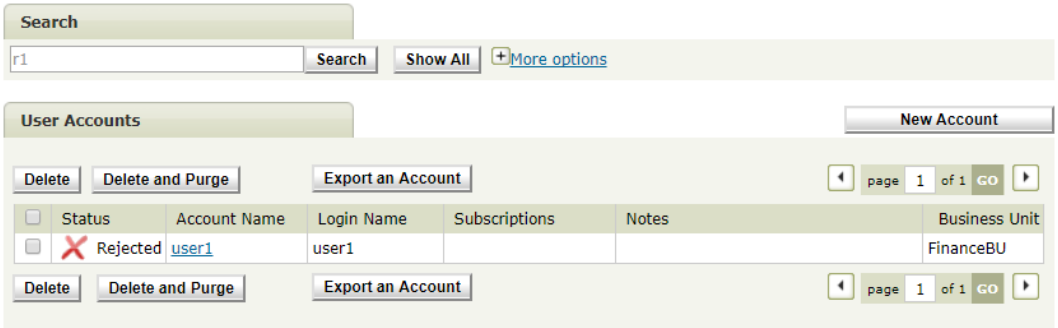
1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. In the *Search* pane, type all or part of a user account name and click **Search**. Wildcards are not accepted.

   The search results are displayed on the *User Accounts* page.



3. (Optional) Click **More Options** to expand the Search panel with Account status checkboxes as additional search filters.
   Note that the **Pending** and **Rejected** account statuses are introduced as part of the Maker-Checker flow.
4. (Optional) Narrow your list of search results. In the *Search* pane, append or prepend your search string.

   For example, adding the string "1" to the end of the search string "s" in the previous example yields the following result.



After you perform a search, the **Show All** button is displayed.

5. To display all accounts, click **Show All**.

View account settings

Use the *Settings* pane to view and modify account status and settings. When an account is defined for an external authenticated user, such as an LDAP or SiteMinder user, the external UID, GID, and home folder attributes are replaced with the values taken from the account information while the external user is logged into SecureTransport.

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click the name of the account whose settings you want to view.

   The *Settings* pane is displayed with details for the selected account.

The following topics describe managing user accounts:

- *Create a user account* on page 503
- *Edit user account settings* on page 515
- *Change how long user account information is cached in memory* on page 516
- *Disable or enable a user account* on page 516
- *Lock or unlock a user account* on page 517
- *Delete or purge a user account* on page 517
- *Manage user account passwords* on page 518
- *Export a single user or service account* on page 519
- *Protected folders and accounts* on page 525
- *Unlicensed users* on page 520

Create a user account

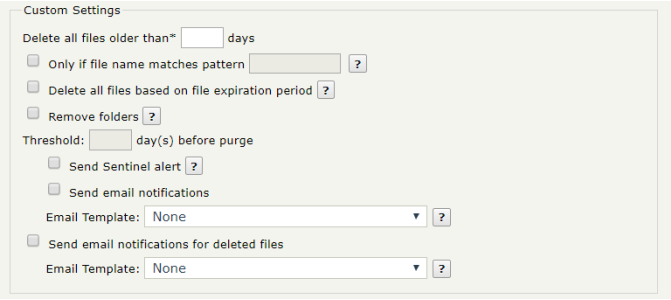
Go to **Accounts > User Accounts** to open the *User Accounts* page and click **New Account**.

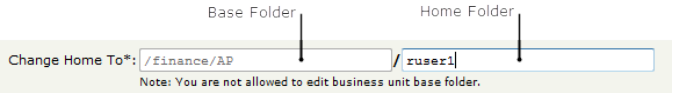
When you create a user account, all the tabs except **Settings** are disabled.

Some of the options are initially hidden. The following table provides detailed information about the account creation options.

Name / Type	Description
Account Name* *text box*	The account name must be unique for the system. If an account with the name you specify already exists, SecureTransport prompts you to enter another name. Account names cannot contain more than 80 characters. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514. **Note** If this user account is to be used for PeSIT transfers, the Account Name may need to follow the naming convention described in *PeSIT configuration overview* on page 601.
Email Contact *text box*	This field is required for using any of the following features of ST Web Client: Address Book, Mailbox, or Share Folder. SecureTransport uses the email address to determine the recipient of an AdHoc file transfer email sent from ST Web Client or an Axway email plug-in. If the user is allowed to log in by email, this is the value used in the **User ID** field on the login page. A user account must be assigned a unique email address in order for the login by email and password reset features to function properly.
Phone contact *text box*	Contact phone number.
Account Type *drop-down list*	Use this parameter to differentiate between accounts that transfer files internally and those that transfer files between partners. Choose from the following: • **Unspecified** – Default value. All accounts created using versions of SecureTransport that do not have this option have this value. • **Internal** – Transfers for this account occur within a single organization. • **Partner** – Transfers for this account occur between organizations.
Business Unit *drop-down list*	Select the business unit the current account will belong to. The default setting is "No Business Unit". **Note** All Business Unit-level policies apply to all accounts in the particular unit.
HTML Template *drop-down list*	This option determines which template is displayed when the user logs in to SecureTransport Web Client. The **Default HTML Template** is the one selected in **Setup > Miscellaneous> HTML Template**. For details, see *Select a default HTML template* on page 201.

Name / Type	Description
PeSIT Routing Mode *drop-down list*	This option controls how SecureTransport behaves when it is an intermediate partner in a PeSIT transfer directed to this account. • If the account is to use Advanced Routing, always select **Ignore**. In this case, the file is received but its final destination (PI 62) is ignored and it is submitted to Advanced Routing for processing. The processed file is then sent to the receiver specified in the AR Send To Partner step. • If the account will not use Advanced Routing, use either **Accept** or **Reject** to choose an action to take when the account has no PeSIT Default transfer site configured and none of its transfer site (FTP, HTTP, PeSIT, SSH, or Connect:Direct) matches the routing destination. **Reject** (default) – the file transfer is rejected before it starts. **Accept** – the file is received successfully and retained locally for reuse.
PeSIT ID	When the server configuration option `Pesit.UsePesitIds` is set to `true`, this value is used for defining a PeSIT partnership. Otherwise, the Account Name is used. For more details, see *PeSIT configuration overview* on page 601
Encrypt Mode *drop-down list*	This field allows you to enable repository encryption for this user. • **Unspecified** (default) – Repository encryption is enabled based on the `EncryptClass` user class evaluation. • **Enabled** – Repository encryption is enabled for this user account.
Subscription Folder Discovery *drop-down list*	This field determines the subscription folder discovery mode. For accounts with multiple subscriptions, the number of subscriptions and the target folder depth may impact performance. • **Iterable** (default) – Subscription folder discovery is performed by iteration over all of the account's subscriptions while trying to match the target folder. Tip: Choose this mode when the number of subscriptions is small and the target folder depth is large. • **Recursive** – Subscription folder discovery is performed by recursive traversal of the target folder hierarchy - the target folder is checked first and if no match is found, then its parent folder is checked. The process continues until a match is found or there are no more folders to check. Tip: choose this mode when the number of subscriptions is large and the target folder depth is shallow.

Name / Type	Description
File archiving policy *drop-down list*	Determines the File archiving policy for the current user account based on the selected option: • **Default** - If the account is assigned to a business unit, it will inherit the business unit policy. Otherwise, the global archiving policy will be applied. • **Enabled** - File archiving is enabled for the account. • **Disabled** - File archiving is disabled for the account. **Note** This setting cannot be modified when the global file archiving policy is disabled or if this account is assigned to a business unit with **Allow File Archiving Policy modifying** option deselected.
File Maintenance policy *drop-down list*	Determines the File Maintenance policy for the current user account based on the selected option: • **Default**- If the account is assigned to a business unit, it will inherit the business unit policy. Otherwise, the global file maintenance policy will be applied. • **Disabled** - File Maintenance is disabled for this account. • **Custom** - When you select this option, the panel expands with a **Custom settings** pane that allows you to modify the existing *File Maintenance application* on page 834. The customized policy applies to this account only.  **Note** This option cannot be modified when the global file maintenance policy is disabled or the account is assigned to a business unit with the **Allow File Maintenance Policy modifying** checkbox deselected.
UID* *text box*	Enter the numeric user ID of the user in the **UID** field. This field is mandatory on UNIX and Linux platforms. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514. On Windows, this field is named **Real User** and is optional.
GID* *text box*	Type the numeric group ID for the user account in the **GID** field. The account uses the system access rights and privileges valid for this user group on the system. You cannot enter spaces-only values in this field.

Name / Type	Description
Change home to *text boxes*	You must assign a **Current Home** folder for your user. Enter a valid home folder in the **Change Home To** field for the account as an absolute path. SecureTransport validates the directory path you specify and prompts you for a new path if necessary. This field is mandatory. You cannot enter spaces-only values in this field.  Add a base folder path in the field to the left of the forward slash (`/`) and add the home folder in the field to the right of it. You can add multiple levels, such as `/home/dev3/test`, but the parent directories must be typed in the field to the left of the slash. Only the final child directory should be in the field to the right of the slash. When you select a business unit, a base folder for the business unit is automatically added. The base folder must be the business unit base folder. You cannot change the base folder for a user account if a business unit is selected unless the business unit has the option **Allow Base Folder modifying** selected. Although you can use the `/` when adding parent directories to a home folder, you cannot use the following characters in the home folder name: `* < > ? " / \	:` **Note** If you change the home folder when editing a user account, any subscription folders the account has are reinitialized. In other words, the subscription folders are created again under the new home folder of the account. None of the other folders created by the user will be moved and the user will no longer have access to them. This also happens if the user is moved from one business unit to another. When SecureTransport is running under Windows: • Account home folder can be specified in UNC format, pointing to a local or a remotely shared folder over the network. • When a network share is used as a home folder for an account, you must manually create a directory with proper access settings. SecureTransport cannot create the home folder because the SecureTransport services run on Windows as service accounts with a local system user as an owner. You must either use SecureTransport impersonation functionality or use permissions sufficient for the network share to be accessed by local system users. For more information, see *Real users on Windows* on page 742. • Transaction Manager agents must use the Windows impersonation functionality (mapping virtual users to real users) as needed to access directories on a network share (that is, directories in UNC format or on mapped drives).

Name / Type	Description
Home Folder Access Level *drop-down list*	The home folder access level determines whether and which other accounts are able to publish to the home folder of the current account. • **Private** – The access level is private. Only the current account is able to publish files to its home folder. • **Business Unit** – Account home folder access is limited to the account's business unit. The current account and all accounts in the current account's business unit can publish to this account's home folder. • **Public** – Access to the account is public. All accounts are able to publish to this account's home folder. **Note** Access level is applicable only when Advanced Routing feature is used. For more information see *Advanced Routing* on page 864
Notes *text box*	Enter a text description of the user account in the **Notes** field.

Name / Type	Description
Delivery Method *drop-down list*	*AdHoc settings* This value controls the options that ST Web Client displays in the *User Access* window. • **Disabled** – The user cannot send files using ad hoc file transfers. • **Default** – Use the delivery method specified in the account template, if any, or in the **Default Package Delivery Method** field of the *AdHoc Setting* page. • **Anonymous** – The sender can choose **Send attachment link only** or **Protect attachment link with security question**. • **Account Without Enrollment** – The sender can choose **Send attachment link only**, **Protect attachment link with security question**, or **Send to existing users only**. • **Account With Enrollment** – The sender can choose **Send attachment link only**, **Protect attachment link with security question**, **Send to existing users only**, **Allow recipients to enroll as restricted users (receive and reply to messages only)**, or **Allow recipients to enroll as unrestricted users**. (Elsewhere the Administration Tool refers to restricted users as unlicensed users and unrestricted users as licensed users.) • **Custom** – Select the allowed enrollment types in the **Enrollment Types** field. The sender can chose any of the selected enrollment types. For a custom delivery method, select one or more of the allowed enrollment types in the **Enrollment Types** field: ◦ **Anonymous** – The ad hoc file recipient receives a link to retrieve the files and is not enrolled as a user. The ST Web Client option is **Send attachment link only**. ◦ **Challenge** – The ad hoc file recipient receives a link and must answer correctly a challenge question specified by the sender to retrieve the files. The recipient is not enrolled as a user. The ST Web Client option is **Protect attachment link with security question**. ◦ **Existing Account** – Do not enroll ad hoc file recipients. Only existing users can receive files. The ST Web Client option is **Send to existing users only**. ◦ **Enroll Unlicensed** – If the ad hoc file recipient does not have a user account, the recipient must enroll and create an account before retrieving the files. The ad hoc file recipient becomes a restricted user who can only reply once to the email and retrieve the files. Other user attributes are defined by the enrollment template. The ST Web Client option is **Allow recipients to enroll as restricted users (receive and reply to messages only)**. ◦ **Enroll Licensed** – If the ad hoc file recipient does not have a user account, the recipient must enroll and create an account before retrieving the files. The ad hoc file recipient becomes a SecureTransport user with all the

Name / Type	Description
	attributes specified in the default enrollment template. The ST Web Client option is **Allow recipients to enroll as unrestricted users** When the value of the **Delivery Method** field is not **Default**, the **Implicit Enrollment Type** value controls which option ST Web Client selects initially in the *User Access* window. The choices depend on the enrollment types enabled by the **Delivery Methods** and **Enrollment Types** fields. **Challenge** is not an option because the Axway Email Plug-in do not include the challenge question and answer function.
Address Book Settings *group of options*	*Address Book Settings* (Optional) When the Address Book feature is enabled, the *Address Book Settings* are displayed. To configure the user account Address Book settings: 1. Select the Address Book source. • **Default** - The account inherits either its business unit Address Book policy or the global Address Book policy. • **Custom** - A custom Address Book policy configuration will be set for this account only and the following will be configurable: a. Enable or disable Address Book sources for the account. b. Specify the parent groups for Address Book sources. c. Specify the domain for LDAP Address Book sources. d. Specify **All Business Units** or **User's own business unit** for local and custom Address Book sources. • **Disabled** - The Address Book policy is set to disabled for this account. 2. Specify whether or not to allow collaboration with non-Address Book recipients. If Address Book functionality is disabled, this setting does not affect user collaboration. • When **checked**, the account will be allowed to send email packages and share folders with users that do not exist in the defined Address Book. • When **unchecked**, the account will be allowed to send email packages and share folders only with users that exist in the defined Address Book. This account setting overrides the business unit or global Address Book Policy setting for collaboration. For additional Address Book account level configuration information, see *Address Book account level configuration* on page 249.

Name / Type	Description
Bandwidth Limits Policy *drop-down list*	*Bandwidth limits* Select a **Bandwidth Limits Policy** to apply: • Default – the current user account inherits their bandwidth limits from the parent business unit or the global bandwidth • Custom – the panel expands with two additional options for you to configure: **Inbound limit** and **Outbound limit** (both values in kb/s per user) • Disabled – no bandwidth limits are applied to the users assigned to the current business unit

Name / Type	Description
Login Settings *check-box-controlled group of options*	In the *Login Settings* area: select **Allow this account to log in to SecureTransport Server** to allow the new account to log in to SecureTransport. This setting is enabled by default. Disabling the option restricts access of this account to the SecureTransport Server. If you enable this option, the following options are enabled. • Enter a **Login Name** for the account. This is the unique name with which the account is identified by the SecureTransport Server. Login names cannot contain the following characters: +, :, or [. Login Names cannot start with the following character: *. • Select the **Login Restriction Policy**. The Login Restriction Policy defines rules for allow or deny login to users based on the client IP or host and other conditions. For additional information, see *Login restrictions* on page 810. ◦ If a Login Restriction Policy is selected as the global default policy, it will be the inherited default selection for the user account. ◦ If a Login Restriction Policy is not selected as the global default policy and the Business Unit has a Login Restriction Policy selected, it will be the inherited default selection for the user account. ◦ If neither a global default Login Restriction Policy or a Business Unit Login Restriction Policy is selected, then the policy selected for the users account will be in effect. **Note** The default inherited Login Restriction Policy can be overridden by selecting a Login Restriction Policy from **On Account**. • Select **Allow this account to login by email** to allow the user to log in using with the value of the **Email Contact** field as well as the **Login Name**. • Select **Allow this account to submit transfers using the Transfers RESTful API** to enable calls from the SecureTransport REST file transfer API authenticated with the credentials from this account. When this option is selected, the account will be allowed to trigger server initiated transfers using the Transfers RESTful API resource and retrieve the tracking information for these transfers. • Select **Password is stored locally (not in external directory)** to store the password locally in the system. SecureTransport stores the passwords of real, LDAP, SiteMinder, and SSO users in an external directory, and the passwords of virtual users are stored in the SecureTransport database. **Note** The **Password is stored locally (not in external directory)** option can only be used for a user account that has a virtual user associated with it. If the user associated with the account is a real, LDAP, SiteMinder, or SSO user, then the password cannot be stored locally in the database and this option is unusable.

Name / Type	Description
	◦ Enter a **New Password** for the account. ◦ **Re-enter Password** for the account. ◦ Select **Require user to change password on next login** to require the user to change their password on the next login. ◦ Select **Require user to set new secret question on next login** to require the user to select and answer a new secret question. When this option is selected, the user must select and answer a new secret question on their next login. For information on configuring the secret question feature, see *Configure a secret question* on page 683. ◦ Complete the **Require user to change password every X days** field to require the user to change their password every specified number of days. If the number of days is unspecified, the user will not be required to change their password every "X" number of days. ◦ Complete the **Lock account after X failed login attempts** field to lock the account after the specified number of failed login attempts. If you don't supply a value, the value of the `Users.DefaultLockoutLimit` configuration option will be applied. If the value is set to 0 [zero], an infinite number of failed login attempts are allowed. **Note** Certificate authentication attempts over HTTP are not counted as failed login attempts. ◦ Complete the **Lock account after X failed ssh key authentication login attempts** field to lock the account after the specified number of failed ssh key authentication login attempts. If you don't supply a value, the value of the `Users.DefaultSshKeyLockoutLimit` configuration option will be applied. If you set the value to 0 [zero], infinite number of failed login attempts via SSH keys will be allowed. ◦ Complete the **Lock account after X successful logins** field to lock the account after the specified number of successful logins. If you don't supply a value, infinite number of successful login attempts will be allowed. The `GlobalLoginThreshold` configuration option is a percentage value that will allow additional successful logins after reaching the threshold specified in the *Account* page (**Lock account after X successful logins**). **Note** A user account's *Last Login* date and time are updated upon a successful manual attempt to log in (including changing the password), as well as upon performing a server- or client-initiated transfer.

Name / Type	Description

Add Attribute
group of options
	Additional Attributes
	
	Add an attribute
	• Click **Add Attribute** to enable input in the **Attribute** and **Value** fields.
	• Enter the respective values and click the Save ( ) icon.
	Repeat the process to add more attributes, if necessary.
	
	Delete an attribute
	• Select the corresponding check-box of the (one or more) attribute to remove and then click **Delete**.
	Delete an attribute
	You can also access these custom properties using any fields in Advanced Routing. See the following examples:
	`${account.attributes['userVars.1']}`
	`${account.attributes['userVars.2']}`
	For example, the `account.attributes` is the selector for attributes of the account used to execute the current route - it has to be written exactly as shown.
	The `userVars.` prefix must be prepended to attribute name.
	All this should be written as an EL expression: `${...}`
	Click the **Save** ( ) icon.
	For more information, see *Additional attributes* on page 759.

After you add all your changes, click **Save**.

The user account information is saved and displayed in the *Settings* tab of the user account.

Once you have saved the account settings, you can select the **Subscriptions**, **Routes**, **Transfer Sites**, or **Certificates** to further define the new account. For more information, see *Manage subscriptions* on page 664, *Manage Routes* on page 881, *Transfer sites* on page 540, and *Manage login certificates* on page 527.

Spaces in required fields

Some fields in SecureTransport require that you enter a value. When you enter a value in such a field, SecureTransport trims any leading or trailing spaces and then determines whether the field is empty. This means you cannot enter space-only values in required fields because those fields are treated as empty.

Maker-Checker user creation

Delegated administrators with Maker and Checker rights have two separate complementing roles:

- Maker creates the user account and submits it for approval
- Checker approves or rejects the pending user account

Create and submit user

As a Maker, you can create a user account that will remain in Pending verification status. Your user will not have access until a Checker administrator *approves* their particular account.

In order to submit the account for approval, go to **Accounts > User Accounts** and on the *Settings* tab click **Submit for approval**.

Approve user

As a Checker administrator, you can only view and approve or reject users in pending Account verification status.

If you reject a pending account, you can type in the reason for rejection.

Edit user account settings

Use the following procedure to edit user account settings. If you are using a Firefox browser, disable the auto complete function before you begin editing.

1. Select **Accounts > User Accounts**.
2. Click the name of the account whose settings you want to edit.
3. Click **Edit Account Settings**.
4. Make your changes. For information about the fields on the *Edit Accounts Settings* page, see *Create a user account on page 503*.
5. Click **Save**.

Change user and group ownership

Changing UIDs and GIDs should be done with due care because it can cause problems with file access and permissions. It's important to know that when changing the UID or GID of an existing user account, the new value affects only content created after the change. SecureTransport does not automatically change the ownership on the home folder or any existing files and folders previously owned by the user to the new UID or GID. However, this change can be done manually. Follow the procedure below to correctly update UIDs and GIDs and fix file and folder ownership:

1. Edit the user account configuration to update the UID or GID.
2. Click **Save** to save the new values.
3. In the user account's **Settings** tab, under **Account Settings**, click **Change File/Folder Ownership**.

A dialog opens and presents you with the change ownership mode options: recursive or non-recursive.

4. Examine the options:

Recursive	resets the UID/GID of the user home folder and all its files and subfolders to the integers specified in the account configuration
Non-Recursive	resets the UID/GID of the user home folder only to the integers specified in the account configuration

5. Click the corresponding button for the mode you would like to use.

Change how long user account information is cached in memory

You can set how long SecureTransport keeps the user account information cached in memory by editing the `TransactionManager.AccountContextAgent.cacheTimeout` parameter on the *Server Configuration* page. The default value is `900` seconds. Change the value to keep the cached information for a longer or shorter period of time. You can also set it to null or a negative value to not cache the account information. Changes you make to an account do not show up until after the cache is refreshed.

Disable or enable a user account

Use the following procedure to disable or enable a user account.

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click **Disable Account** or **Enable Account**, depending on the current status of the account.
3. SecureTransport asks you whether to change the owner of that account's home directory and files. If you have changed the account's UID or GID, you have to reassign the ownership of the account home directory to the new UID/GID pair. Select from the following actions:
    - Press **Recursive** to change the ownership on the home directory and its content and save all changes made to the account configuration.
    - Press **Non-Recursive** to change the ownership on the home folder only and save all changes made to the account configuration.
    - Press **No Change** to keep the current UID/GID on the home folder and save the current account configuration.
    - Press **Cancel** to close the prompt and discard all changes.

If an account is disabled:

- Scheduled subscriptions for the account are not triggered.
- The user associated with the account cannot log in the system or perform any transfers.

Lock or unlock a user account

Use the following procedure to lock or unlock a user account.

1. Select **Accounts > User Accounts**.

   The *User Accounts* page is displayed.

2. On the Click the name of the account that you want to lock or unlock.

   The *Settings* pane is displayed with details for the selected account.

3. Click **Lock Account** or **Unlock Account**, depending on the current status of the account.

   Note An account can be locked or unlocked only if it has a user associated with it. When an account is locked, the user associated with it cannot log in to the SecureTransport Server. However, if an account is locked, server-initiated transfers associated with that account are not affected.

Accounts are locked when:

- When an administrator locks an account using the lock user account procedure.
- When the number failed login attempts exceeds the configured number of allowed attempts.
- When the number failed ssh key authentication login attempts exceeds the configured number of allowed attempts.
- When the number successful login attempts exceeds the configured number of successful attempts.
- When the day count exceeds the configured number of days between password changes.

  Note Change password failures are counted as failed login attempts.

Delete or purge a user account

When you delete a user account, all its subscriptions to applications are also deleted. However, the home folder for that account is kept on the server.

Delete an account

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Select the checkbox for the user account you want to delete. You can select multiple accounts.
3. Click **Delete**. SecureTransport prompts you to confirm that you want to delete the account.
4. Click **OK** to confirm deletion.

Purge an account

To delete an account and remove the home folder and any subdirectories, use the **Delete and Purge** button.

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Select the checkbox for the user account you want to delete you can select multiple accounts.
3. Click **Delete and Purge**. SecureTransport prompts you to confirm that you want to delete the account.
4. Click **OK** to confirm deletion.

Manage user account passwords

This subsection describes procedures for changing and expiring passwords of user accounts, as well as applying configuration

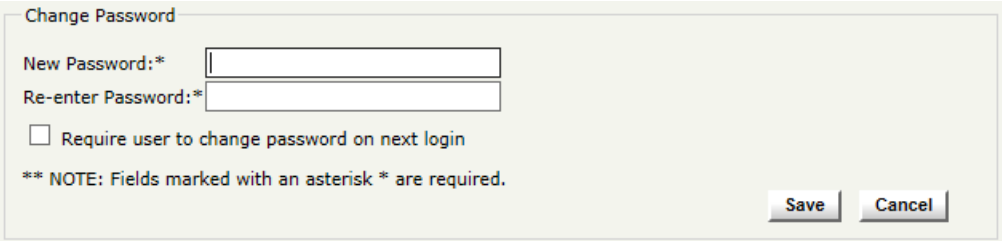
Change a user account password

Use the following procedure to change a user account password.

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click the name of the account whose password you want to change.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click **Change Password**.

   The *Change Password* pane is displayed.



4. Type the new password.
5. Re-type the password to confirm it.
6. (Optional) Select the **Require user to change password on next login** checkbox.
7. Click **Save**.

Note Password policy restrictions apply.

Expire a user account password

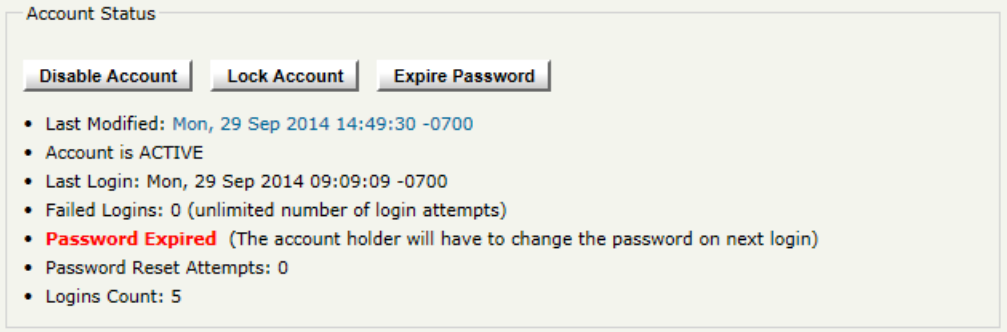
When you expire the password of an account, the user is prompted to change the password on next login. The new password must follow the password policy you configured for SecureTransport. For more information, see *Set password policy* on page 205.

Note If you change the password policy of a user while that user is logged in to SecureTransport through a web client, the old password policy is displayed until the user logs out and logs in again.

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click the name of the account whose password you want to expire.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click **Expire Password**.

   The *User Account* page is displayed with the red text in the *Account Status* pane stating that the password is expired.



Web password compatibility

If a user created a password that contained the plus symbol (+) while running a web client in a previous version of SecureTransport, you must change the value of the `Http.Password.Compatability.Mode` server configuration parameter to `on` to allow passwords with the + symbol.

When its value is set to `on`, this parameter allows users who previously changed their password to a string containing the plus character through a web client to log in and change their password. This option affects only the web clients. Once the password is changed, the user can log in using any client or protocol.

If you have upgraded or performed a system import, and you did not already manually change the parameter, SecureTransport sets the value set to `on` to allow legacy users with a + symbol in the password to log on without an error.

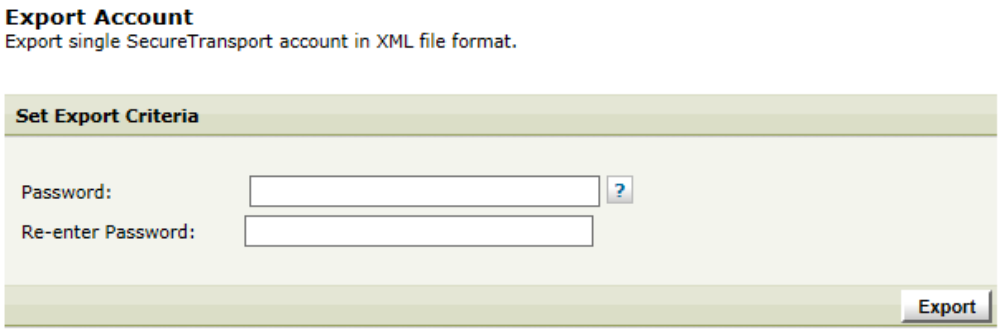
Export a single user or service account

SecureTransport exports a single user account to an XML file you can download to your local computer. During the export you specify a password that is used to encrypt the sensitive information contained in the account. You use this password when you import the account, to decrypt the sensitive information.

You can export a single user account to an XML file.

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Select the account you want to export and click **Export an Account**.

   The *Export User Account* page is displayed.



3. Choose a password to encrypt the sensitive information contained in the account.
4. Re-enter the password.
5. Click **Export**.

Export a single service account

You can export a single service account to an XML file.

1. Select **Accounts > Service Accounts**.
2. Select the account you want to export and click **Export an Account**.
3. Choose a password to encrypt the sensitive information contained in the account.
4. Re-enter the password.
5. Click **Export**.

To export and import all accounts, see *Export and import accounts* on page 686.

Unlicensed users

The recipient of an ad hoc human-to-human file transfer email or a system-to-human file transfer email does not need all SecureTransport user account functionality to retrieve the files. If the recipient does not need other SecureTransport functionality, the recipient can be an unlicensed user. As the name implies, unlicensed users do not require an account license to connect to SecureTransport to retrieve or reply to messages.

Unlicensed users can log in to SecureTransport using ST Web Client to retrieve files. They can reply once to the sender of a file transfer email but cannot create or forward messages with ad hoc file transfers. For unlicensed users, ST Web Client does not offer file transfer functions.

If the ad hoc human-to-human file transfer specifies enrollment as an unlicensed user or the System to Human transfer site specifies a security level of `Enroll User - Unlicensed`, the recipient becomes an unlicensed user on enrollment.

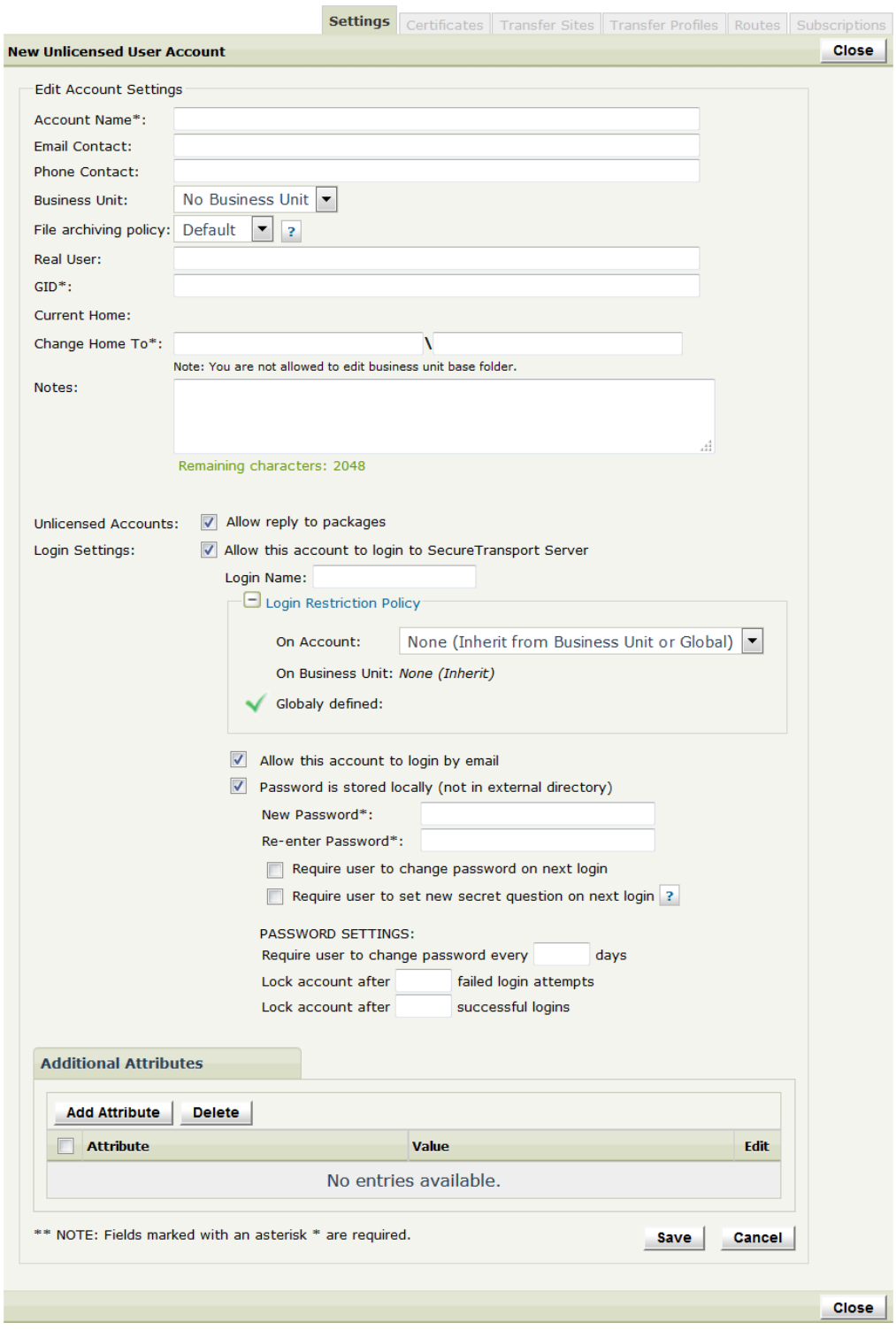
Create account templates for unlicensed users. Configure the account template that applies to unlicensed users in the business unit or in the **Default enrollment account template** field on the *AdHoc Setting* page. See *Manage account templates* on page 719, *Create or edit a business unit* on page 1, and *Configure AdHoc file transfers* on page 87.

You can also create an unlicensed user account from the *Unlicensed User Accounts* page.

1. Select **Accounts > Unlicensed Users**. The *Unlicensed User Accounts* page is displayed.
2. Click **New Account**.

   The *Settings* pane of the *New Unlicensed User Account* page is displayed.

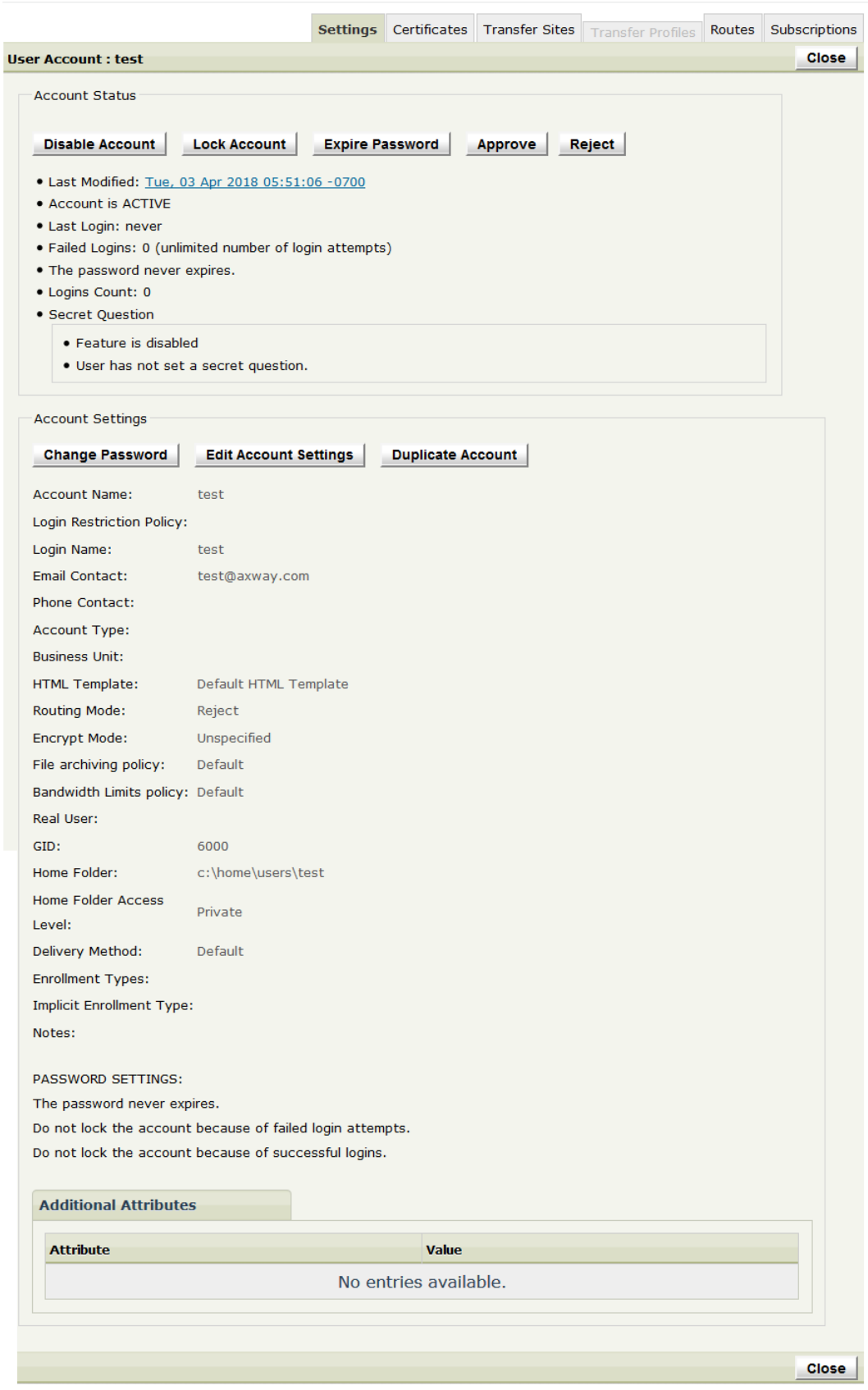
   Note The *Address Book Settings* are only displayed if the Address Book feature is enabled (the value of the `AddressBook.Enabled` configuration option is set to **true**). For Address Book account level configuration instructions, refer to *Address Book account level configuration* on page 249.



3. For the common fields, see *Create a user account* on page 503.
4. Select **Allow reply to packages** to permit the unlicensed user to reply once to a received package.

5. If **Allow this account to login to SecureTransport Server** is not selected, the following fields are not displayed.
6. Click **Save**.

   The user account information is saved and displayed in the *Settings* pane of the user account. The other user account tabs are not available for unlicensed users.


Settings
Certificates
Transfer Sites
Transfer Profiles
Routes
Subscriptions
User Account : test
Close
Account Status
Disable Account
Lock Account
Expire Password
Approve
Reject
• Last Modified: Tue, 03 Apr 2018 05:51:06 -0700
• Account is ACTIVE
• Last Login: never
• Failed Logins: 0 (unlimited number of login attempts)
• The password never expires.
• Logins Count: 0
• Secret Question
• Feature is disabled
• User has not set a secret question.
Account Settings
Change Password
Edit Account Settings
Duplicate Account
Account Name: test
Login Restriction Policy:
Login Name: test
Email Contact: test@axway.com
Phone Contact:
Account Type:
Business Unit:
HTML Template: Default HTML Template
Routing Mode: Reject
Encrypt Mode: Unspecified
File archiving policy: Default
Bandwidth Limits policy: Default
Real User:
GID: 6000
Home Folder: c:\home\users\test
Home Folder Access Level: Private
Delivery Method: Default
Enrollment Types:
Implicit Enrollment Type:
Notes:
PASSWORD SETTINGS:
The password never expires.
Do not lock the account because of failed login attempts.
Do not lock the account because of successful logins.
Additional Attributes
Attribute
Value
No entries available.
Close

You can convert an unlicensed user account to a licensed user account.

1. Select **Accounts > Unlicensed Users**. The *Unlicensed User Accounts* page is displayed.
2. Click the name of the account to convert to a licensed user account.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click **Convert to Licensed**.

   The Administration Tool displays the *Settings* pane for the new user account.

The other procedures are the same for licensed and unlicensed user accounts.

Protected folders and accounts

SecureTransport maintains a list of directories which you should not use for home folders for user or service accounts. This type of directory is called a *protected folder*. Protected folders are identified by a specific prefix in the path. The following table lists the prefixes used by default.

Virtual accounts can be purged using SecureTransport, provided these accounts are not in a protected folder.

SecureTransport provides the following precautions that are built-in to prevent accidental or malicious account deletion:

- Paths are converted to equivalent paths without any "." or ".." directories.
- The user home folder cannot directly, or indirectly through a symbolic link, refer to any of the protected directories.
- If the entry for a user home folder is not a directory, it is not purged.
- If the user home folder begins with any of the protected home folder prefixes, the account is not purged.

Platform	Protected home folder prefixes
Linux, Axway Appliance	`/audit /bin /boot /dev /etc /kernel /lib /lpp /mnt /modules /net /opt /platform /proc /root /sbin /stand /sys /tftp /usr /var /vol`
Windows (in Cygwin Format)	(none)

Note You can add to the list of protected folders by modifying the `UnsafePaths` server configuration option. When adding a folder name that contains spaces, use quotes around the path so the entire path is recognized, for example, "`/user 1/`". Do not remove any of the default protected folder prefix. Make the change on all servers in your Standard Cluster (SC) or Enterprise Cluster (EC).

User certificates

User certificates are managed on a per-account basis from the **Accounts** menu in the Administration Tool. They are generated, imported, exported, and deleted for the respective accounts. They fall into the following three types:

- **Login** – They do not have a private key and are used for logging to SecureTransport Servers. Their private key is exported during the generation of the certificate. The supported formats are X509 and SSH.
- **Partner** – They only have a public key and are used for encrypting PGP and AS2 data to an account and verifying the signature of data from the account. The supported formats are X509 and PGP.
- **Private** – They have a private key and are used for decryption and signing of PGP and AS2 data. The supported formats are X509, SSH and PGP.

Certificate uniqueness

SecureTransport provides server configuration options for controlling the certificate uniqueness per type. For instructions, refer to the dedicated topics for managing user certificates:

- *Manage login certificates* on page 527
- *Manage partner certificates* on page 530
- *Manage private certificates* on page 534

Overwrite settings

On user level, you can overwrite an existing certificate through the REST API by adding `"overwrite": true` in the POST request body. When generating a new certificate, set this property to false or remove it.

In the Administration Tool, you can overwrite a private or partner certificate by importing a new one with an alias that already exists.

Security considerations

By default, SecureTransport 5.5 does not allow you to import certificates that violate ITU-T X.690 standards. To be able to import such certificates, add the following Java option in `<FILEDRIVEHOME>/bin/start_admin`:

```
JAVA_OPTS="-Dorg.bouncycastle.asn1.allow_unsafe_integer=true
$JAVA_OPTS"
```

and restart the Admin service.
Note that Axway does not recommend the import of such certificates as they can cause encoding issues.

Manage login certificates

SecureTransport uses login certificates when the respective (currently active) account logs in to a SecureTransport Server using a certificate or SSH Key.

You can view, generate, import and delete login certificates for the active account from the **Login Certificates** tab. It displays automatically when you click the **Certificate** tab in the *User Account* page.

Certificate uniqueness

The `CertificateStores.UserCertificateStore.Keystore.uniqueCerts` server configuration option controls whether the same login certificate can be used by multiple accounts. It applies to X509 certificates and SSH keys, and not to PGP keys. SecureTransport determines if a certificate is unique by comparing its fingerprint and content against the other user accounts' certificates. The option is enabled by default, which means that the same login certificate cannot be used for multiple accounts. To allow multiple accounts to use the same certificate, set it to **false** and restart all services.

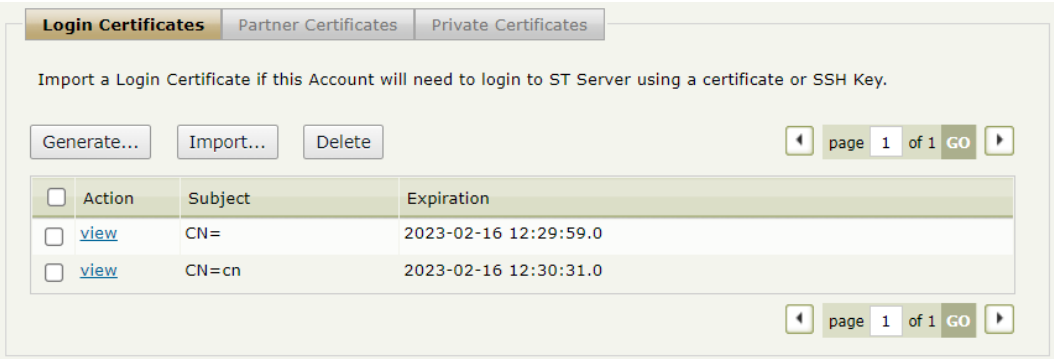
View and export a login certificate

Use the following procedure to export a login certificate.

1. Select **Accounts > User Accounts**.
2. Click the name of the account for which you want to view the certificate.

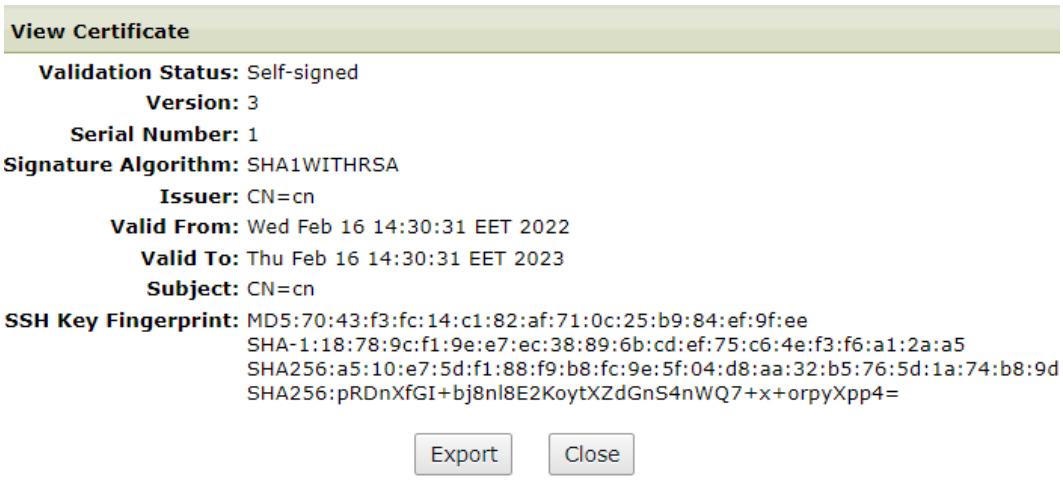
   The *User Account Settings* page is displayed with details for the selected account.
3. Click the **Certificates** tab and then click **Login Certificates**.

   A list of login certificates for the selected account is displayed on the *Login Certificates* page.



4. Click the **View** link corresponding to the certificate you want to export.

   Certificate information for the certificate you selected is displayed in a *View Certificate* dialog box.



5. In the *View Certificate* dialog box, click **Export**.

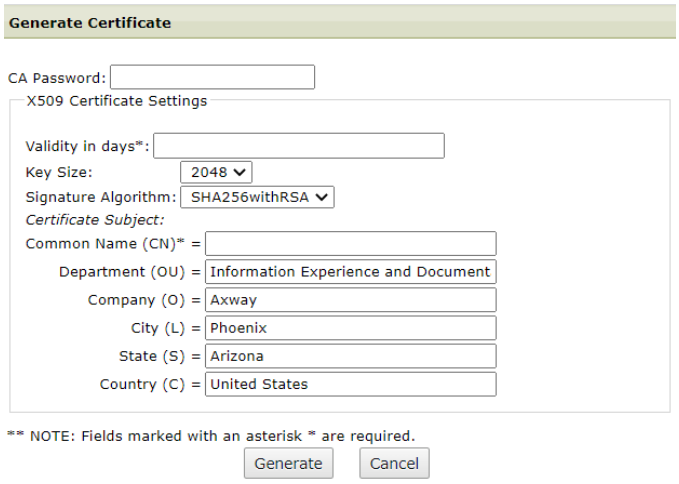
   The file is automatically downloaded and saved in your default download location.

Generate a login certificate

SecureTransport generates only X509 login certificates.

1. In the **Login Certificates** tab, click **Generate**.

   The *Generate Certificate* dialog box is displayed.



2. Type the necessary information in the *Generate Certificate* dialog box, and then click **Generate**.

   Validity in days and **Common Name (CN)** are required fields.

Import a login certificate

Before the user attempts to log in using an imported certificate, ensure that the CA referenced in the certificate is included in the trusted CAs for the SecureTransport Server. For details, see *Manage trusted CAs on page 55*.

SecureTransport generates only X509 login certificates, but you are allowed to import both X509 and SSH key certificates. When importing certificates, consider the following:

- The SSH login public key may not be unique. SecureTransport supports the use of DSA, RSA, ECDSA, and ED25519 SSH keys as SSH login public keys.
- By default, SecureTransport requires that the X509 login certificates be unique. See *Certificate uniqueness on page 527*.

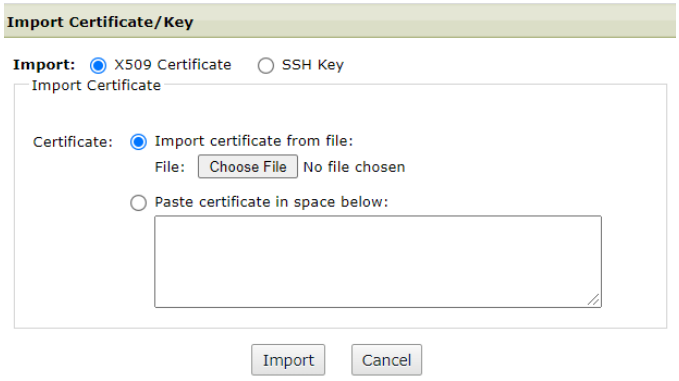
Use the following procedure to import an X509 login certificate or an SSH login public key:

1. Go to **Accounts > User Accounts**, and select an account.
2. On the *User Account* page, click the **Certificates** tab.

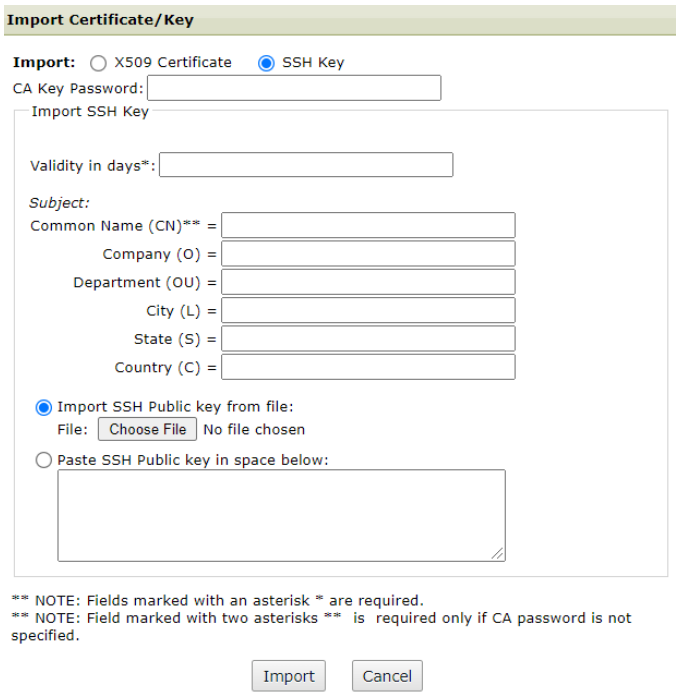
   The list of login certificates for the selected account is displayed.
3. Click **Import**.

   The *Import Certificate/Key* dialog box is displayed.

- To import an X509 certificate, select **X509 Certificate**. In the displayed *Import Certificate/Key* dialog box, paste the certificate content directly in the provided space, or import the certificate from a file. To import from a file, type in the file path or click **Browse** to browse for the file.



- To import an SSH login public key, select **SSH Key**. In the displayed *Import Certificate/Key* dialog box, type in the necessary information. **Validity in days** is a required field, and either **Common Name (CN)** or **CA Key Password** should be specified. Paste the SSH public key directly in the provided space, or import the SSH public key from a file. To import from a file, type in the file path or click **Browse** to browse for the file containing the key.



Note When an SSH key is imported without providing the internal **CA Key Password**, the key will be stored as an X509 certificate and signed with a temporarily generated certificate. As a result, the SSH key will be stored as an X509 self-signed certificate.

4. Click **Import**.

Delete one or more login certificates

Use the following procedure to delete one or more login certificates.

1. Go to **Accounts > User Accounts**, and click the account name whose certificate you want to delete.
2. On the *User Account* page, click the **Certificates** tab.

   The list of login certificates for the selected account is displayed.
3. Select the checkbox next to the certificate you want to delete, and then click **Delete**.
4. Click **OK** to confirm the deletion of the certificate. Otherwise, click **Cancel**. If **OK** is clicked, the selected certificate will be deleted and cannot be recovered.

Manage partner certificates

SecureTransport uses partner certificates as public certificates for encrypting PGP and AS2 data to the respective account and for verification of the signature of data from the account.

You can view, generate, import and delete partner certificates for the active account from the *Partner Certificates* tab page.

Enforce certificate uniqueness

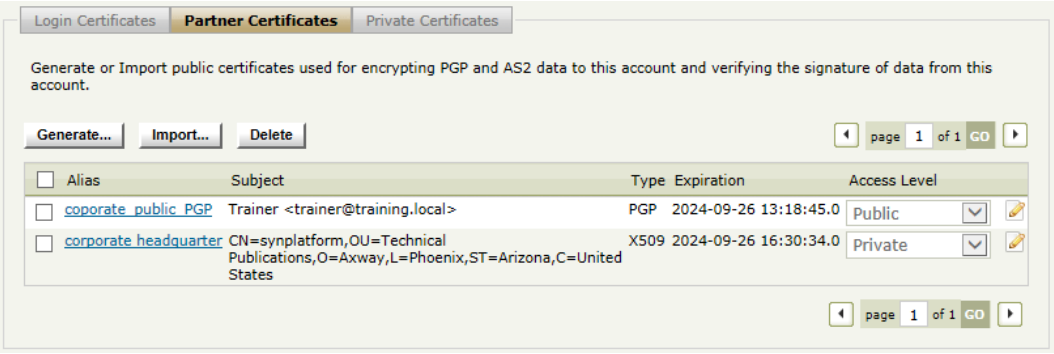
The `CertificateStores.PartnerCertificateStore.uniqueCerts` server configuration option controls whether the same partner certificate can be used by multiple accounts. It applies to X509 certificates and SSH keys, and not to PGP keys.
SecureTransport determines if a certificate is unique by comparing its content against the other user accounts' certificates. The option is not set by default, which means that the same partner certificate can be used for multiple accounts. To enforce certificate uniqueness, set it to **true** and restart all services.

View and export a partner certificate

1. Select **Accounts > User Accounts**.
2. Click the name of the account for which you want to view the certificate.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click the **Certificates** tab, and then click **Partner Certificates.**

   A list of partner certificates for the selected account is displayed on the *Partner Certificates* page.



4. Click the alias name of the partner certificate you want to view or export.

   Partner certificate information is displayed in the *View Certificate* dialog box.
5. Click **Export**.

Generate a partner certificate

SecureTransport generates X509 and PGP partner certificates.

Generate an X509 partner certificate

Use the following procedure to generate an X509 partner certificate.

1. Select **Accounts > User Accounts**.
2. Click the name of the account where you want to add the certificate.

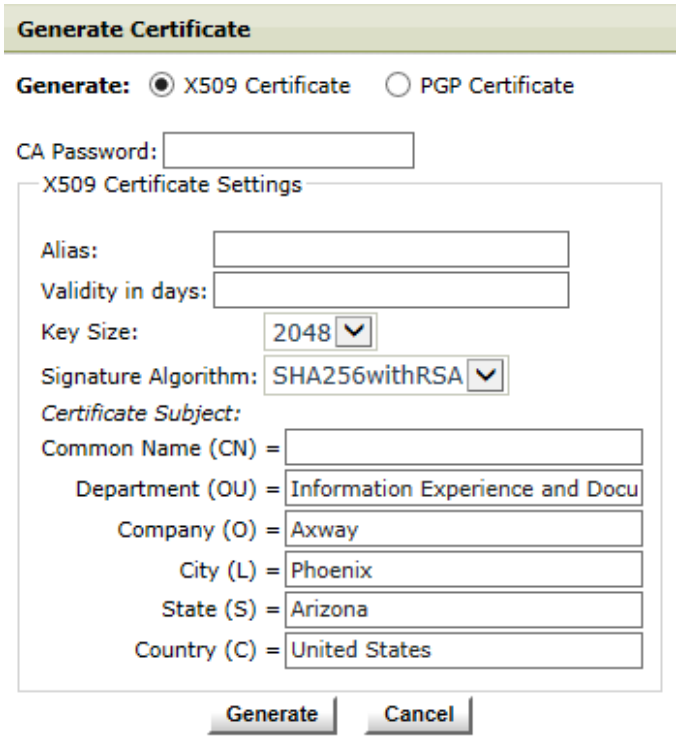
   The *User Account Settings* page is displayed with details for the selected account.

3. Click the **Certificates** tab, and then click **Partner Certificates.**

   A list of partner certificates for the selected account is displayed on the *Partner Certificates* page.

4. Click **Generate**.

   The *Generate Certificate* dialog box is displayed.



5. Select **X509 Certificate**.
6. Type the necessary information in the *Generate Certificate* dialog box.

   Alias, **Validity in days**, and **Common Name (CN)** are required fields. The value you specify in the **Alias** field cannot exceed 50 characters in length.

7. Click **Generate**.

Generate a PGP partner certificate

Use the following procedure generate a PGP partner certificate.

1. Select **Accounts > User Accounts**.
2. Click the name of the account where you want to add the certificate.

   The *User Account Settings* page is displayed with details for the selected account.

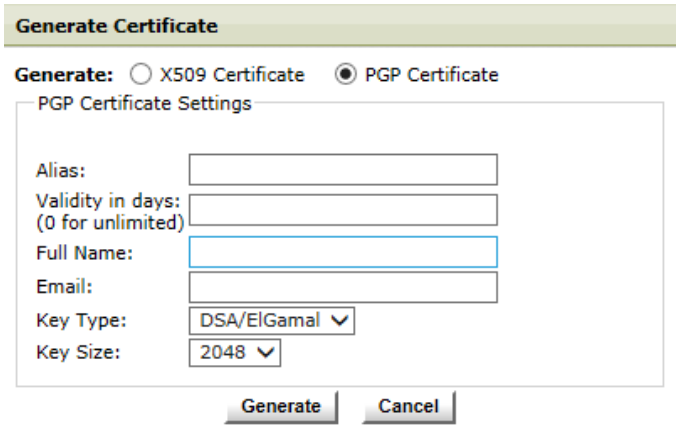
3. Click the **Certificates** tab, and then click **Partner Certificates.**

   A list of partner certificates for the selected account is displayed on the *Partner Certificates* page.

4. Click **Generate**.

   The *Generate Certificate* dialog box is displayed.

5. Select **PGP Certificate**.



6. Type the necessary information in the *Generate Certificate* dialog box.

   Alias, **Validity in days**, and **Full Name** are required fields. The value you specify in the **Alias** field cannot exceed 50 characters in length.

7. Click **Generate**.

Import a partner certificate

SecureTransport imports X509 and PGP partner certificates.

Use the following procedure to import a partner certificate.

1. Select **Accounts > User Accounts**.
2. Click the name of the account where you want to import the certificate.

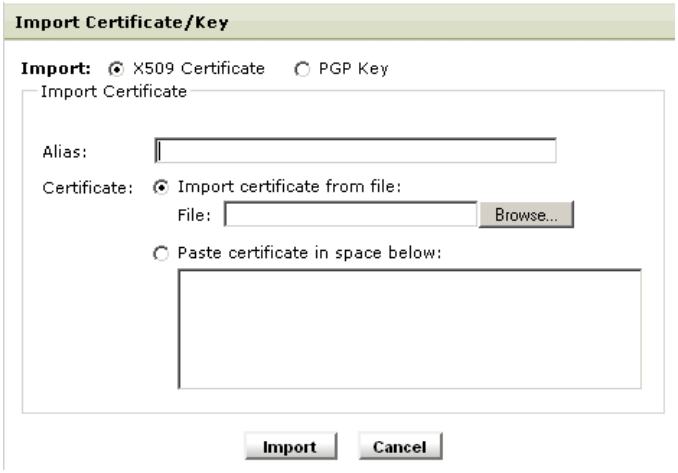
   The *User Account Settings* page is displayed with details for the selected account.

3. Click the **Certificates** tab, and then click **Partner Certificates.**

   A list of partner certificates for the selected account is displayed on the *Partner Certificates* page.

4. Click **Import** button.

   The *Import Certificate/Key* dialog box is displayed.



5. Select the certificate type of the certificate you want to import: **X509** or **PGP**.
6. Type the necessary information in the *Import Certificate/Key* dialog box.

   Paste the certificate content directly in the provided space, or import the certificate from a file. To import from file, type the file path or click **Browse** to browse for the file.
7. Click **Import**.

Note The PGP keys imported for use with an account must specify signing algorithms.

Delete one or more partner certificates

Use the following procedure to delete one or more partner certificates.

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click the name of the account containing the certificate.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click the **Certificates** tab, and then click **Partner Certificates.**

   A list of partner certificates for the selected account is displayed on the *Partner Certificates* page.
4. Select the checkbox next to the certificate you want to delete, and then click **Delete**.
5. Click **OK** to confirm the deletion of the certificate. Otherwise, click **Cancel**. If **OK** is clicked, the selected certificate will be deleted and cannot be recovered.

Manage private certificates

SecureTransport uses private certificates to log in to remote transfer sites for this account, as well as for decrypting and signing PGP and AS2 data.

You can view, generate, import and delete private certificates for the active account from the *Private Certificates* page.

Enforce certificate uniqueness

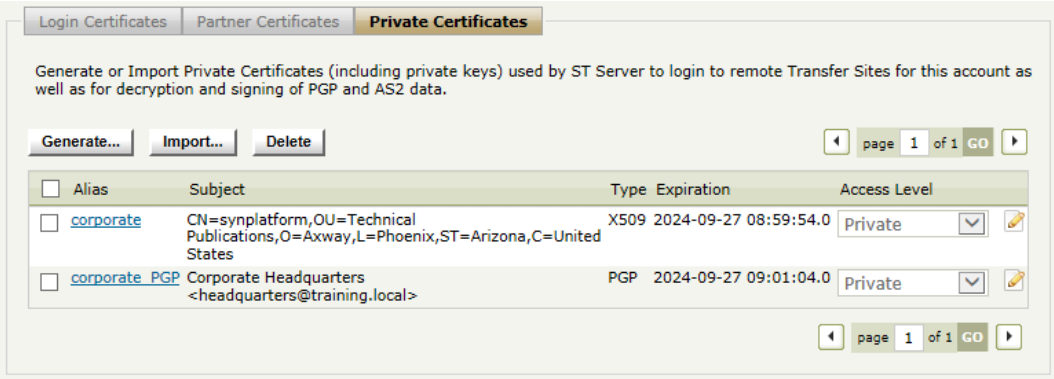
The `CertificateStores.AccountLocalCertificateStore.uniqueCerts` server configuration option controls whether the same private certificate can be used by multiple accounts. It applies to X509 certificates and SSH keys, and not to PGP keys.
SecureTransport determines if a certificate is unique by comparing content against the other user accounts' certificates. The option is not set by default, which means that the same private certificate can be used for multiple accounts. To enforce certificate uniqueness, set it to **true** and restart all services.

View a private certificate

1. Select **Accounts > User Accounts**.
2. Click the name of the account for which you want to view the certificate.

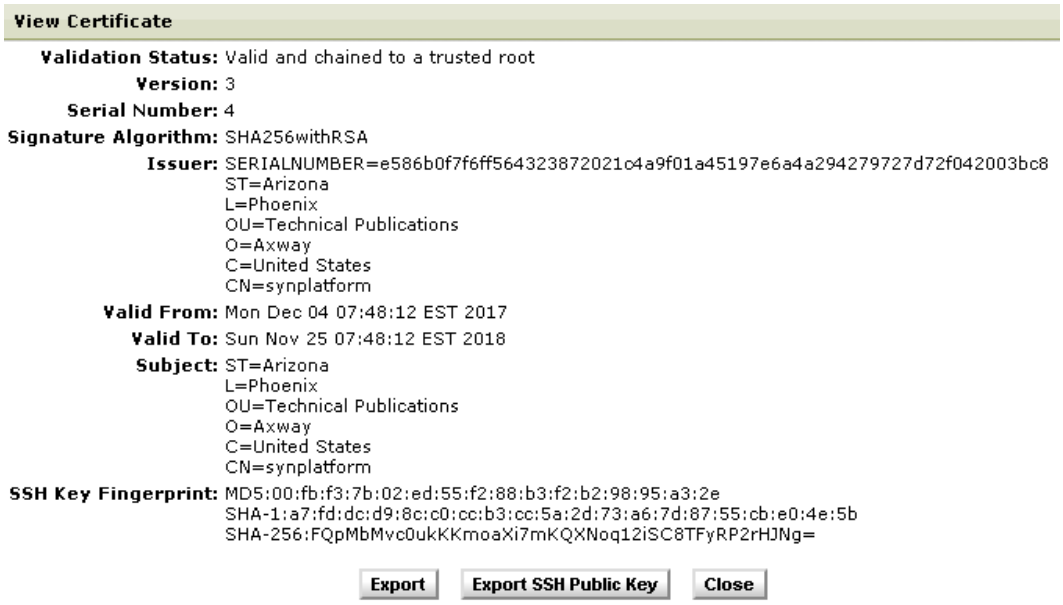
   The *User Account Settings* page is displayed with details for the selected account.
3. Click the **Certificates** tab, and then click **Private Certificates**.

   A list of private certificates for the selected account is displayed on the *Private Certificates* page.



4. Click the name of the certificate you want to view.

Private certificate information is displayed in the *View Certificate* dialog box.



Export the SSH public key of an X509 private certificate

Use the following procedure to export the SSH public key of an X509 private certificate.

1. Select **Accounts > User Accounts**.
2. Click the name of the account containing the certificate.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click the **Certificates** tab, and then click **Private Certificates.**

   A list of private certificates for the selected account is displayed on the *Private Certificates* page.
4. Click the name of the certificate you want to export.

   Private certificate information is displayed in the *View Certificate* dialog box.
5. Click **Export SSH Public Key**.
6. When prompted, save the file containing the key on your file system.

Note An SSH Key can be exported for each X509 certificate. In SecureTransport, all certificates are stored as X509 or PGP ones. Imported SSH Keys are also stored as X509 certificates.

Generate private certificates

SecureTransport generates X509, SSH, and PGP private certificates.

Generate an X509 private certificate

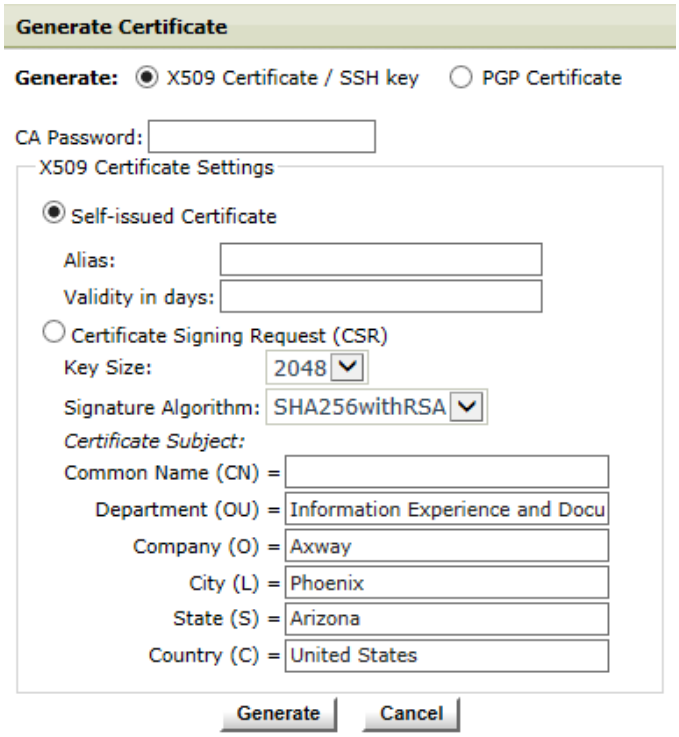
Use the following procedure to generate an X509 certificate.

1. Select **Accounts > User Accounts**.
2. Click the name of the account where you want to add the certificate.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click the **Certificates** tab, and then click **Private Certificates.**

   A list of private certificates for the selected account is displayed on the *Private Certificates* page.
4. Click **Generate**.

   The *Generate Certificate* page is displayed.
5. Select the **X509 Certificate / SSH key** radio button.



6. Select either a **Self-issued Certificate** or a **Certificate Signing Request (CSR)**.

   Note If you select **Certificate Signing Request (CSR)**, the field below the**Self-issued Certificate** option are disabled for editing.
7. Depending on your choice, type the necessary information.

   Alias, **Validity in days**, and **Common Name (CN)** are required fields. The value you specify in the **Alias** field cannot exceed 50 characters in length.
8. Click **Generate**.

Generate a PGP private certificate

Use the following procedure to generate a PGP private certificate.

1. Select **Accounts > User Accounts**.
2. Click the name of the account where you want to add the certificate.

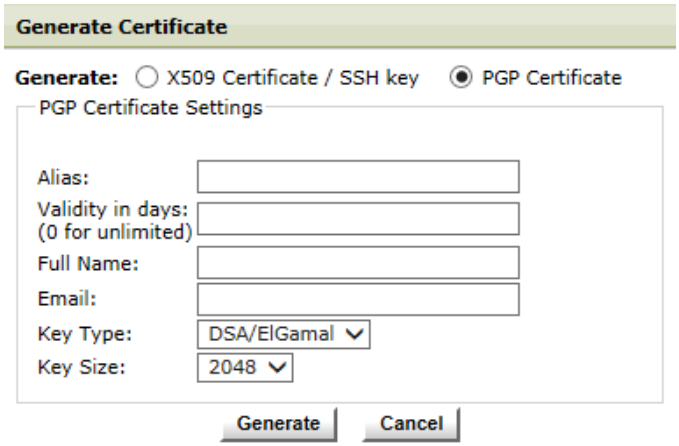
The *User Account Settings* page is displayed with details for the selected account.

3. Click the **Certificates** tab, and then click **Private Certificates.**

   A list of private certificates for the selected account is displayed on the *Private Certificates* page.

4. Click **Generate**.

   The *Generate Certificate* page is displayed.



5. Select the **PGP Certificate** radio button.
6. Type the necessary information and click **Generate**.

   Alias, **Validity in Days**, and **Full Name** are required fields. The value you specify in the **Alias** field cannot exceed 50 characters in length.

Import a private certificate

SecureTransport imports X509, PGP, and SSH private certificates.

Import an X509 or PGP private certificate

Use the following procedure to import an X509 or PGP private certificate.

1. Select **Accounts > User Accounts**.
2. Click the name of the account where you want to import the certificate.

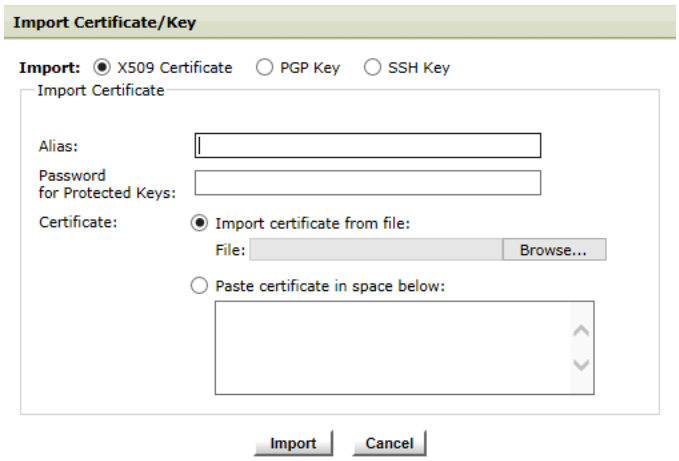
   The *User Account Settings* page is displayed with details for the selected account.

3. Click the **Certificates** tab, and then click **Private Certificates.**

   A list of private certificates for the selected account is displayed on the *Private Certificates* page.

4. Click **Import**.

   The *Import Certificate/Key* page is displayed.



a. Select the type of the certificate you want to import: **X509** or **PGP**.

b. Type the certificate **Alias**. The value you specify in the **Alias** field cannot exceed 50 characters in length.

c. Type the certificate **Password**, if one was specified during the certificate generation.

d. Paste the certificate content directly in the provided space, or import the certificate from a file. To import from file, type the file path or click **Browse** to browse for the file.

5. Click **Import**.

Note The PGP keys imported for use with an account must specify signing algorithms.

Import an SSH private certificate

Use the following procedure to import an SSH private certificate.

1. Select **Accounts > User Accounts**.
2. Click the name of the account where you want to import the certificate.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click the **Certificates** tab, and then click **Private Certificates**.

   A list of private certificates for the selected account is displayed on the *Private Certificates* page.
4. Click **Import**.

   The *Import Certificate/Key* page is displayed.
5. Select the certificate type of the certificate you want to import: **SSH Key**.
6. Type the **CA Key Password** specified during the certificate generation.

   Note **CA Key Password** is not a required field. When a SSH key is imported (without providing the internal CA key password), the key will be stored as X.509 certificate and signed with temporarily generated certificate. As a result, the SSH key will be stored as X.509 self-signed certificate.
7. Type the information necessary to import the key.

Alias and **Validity in days** are required fields. The value you specify in the **Alias** field cannot exceed 50 characters in length.

8. Paste the certificate content directly in the provided space, or import the certificate from a file. To import from file, type the file path or click **Browse** to browse for the file.
9. Click **Import**.

Delete one or more private certificates

Use the following procedure to delete one or more private certificates.

1. Select **Accounts > User Accounts**.
2. Click the name of the account containing the certificate.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click the **Certificates** tab, and then click **Private Certificates.**

   A list of private certificates for the selected account is displayed on the *Private Certificates* page.
4. Select the checkboxes next to the certificates you want to delete, and then click **Delete**.
5. Click **OK** to confirm the deletion of the certificate. Otherwise, click **Cancel**. If **OK** is clicked, the selected certificate will be deleted and cannot be recovered.

Transfer sites

A transfer site is a location such as a local folder or protocol server used when sending or receiving files during a server-initiated transfer. SecureTransport comes with pre-built transfer sites for file transfers based on AS2, Folder Monitor, FTP(S), Generic-HTTP(S), HTTP(S), PeSIT, SSH (SFTP), and System to Human. Support for other transfer protocols can be added using connectors. See *Pluggable Transfer Sites* on page 639.

Transfer sites are specified and managed on a per-account basis. When defining a transfer site as part of an account, you need to provide the information required for connecting to the site, authentication and sending or receiving files. The transfer protocol you select dictates the information required in the transfer site configuration.

In general, a site is used for server-initiated or AS2 transfers. In the case of a server-initiated transfer, a file is either uploaded to the site (when a subscription processes an uploaded file) or downloaded from the site using a schedule in a subscription.

An account can have zero or more transfer sites. An account can subscribe to zero or more applications, and an application can be triggered when a file is transferred using a site.

Common properties for all transfer sites

A transfer site in SecureTransport is defined with various properties. To avoid repetition, the *common properties* for all Transfer Sites are briefly described here. Properties which are custom to each Transfer Site are described in the dedicated for each available transfer site type (scroll to the bottom

of this page).

- **Site Name** – the name of the site. This name must be unique for the account.

  You cannot enter spaces-only values in the Site Name field.

- **Site Type** – indicates whether transfers are internal (within a single organization) or partner (between organizations). Reported to Axway Sentinel as `USERPARAMETER1` and displayed in the Sentinel event attributes.

- **Access Level** - transfer site access level determines whether and which other accounts could reuse this transfer site in the Send To Partner step.

  - **Private** – The access level is private. Only the current account is able to use this transfer site.
  - **Business Unit** – Access to the transfer site is limited to the account's business unit. The current account and all accounts in the current account's business unit can use this transfer site.
  - **Public** – Access to the transfer site is public. All accounts are able use this transfer site.

  Note Access level is applicable only when Advanced Routing feature is used. For more information see *Advanced Routing* on page 864.

- **Maximum parallel transfers** – If you enter a value greater than zero, SecureTransport executes only the specified number of transfers in parallel. If the value is null or zero, the maximum number of parallel transfers is limited by system capacity.

  The maximum number of parallel transfers limit is applied cluster wide. The limit for files transferred from the client will not be exceeded. Due to limitations in Standard Cluster communication mode, the parallel pulls limit can be exceeded when there are several connections. If you want to force the limit, then the `force.standard.cluster.sit.transfers.sync=true` system property should be added to the `start_tm_console`. Adding the property to the `start_tm_console` has a performance penalty due to increased cluster communication.
  Note that the `force.standard.cluster.sit.transfers.sync` value overrides the value of the `force.standart.cluster.sit.pulls.sync` property, used in previous SecureTransport versions for the same purposes.
  You can configure **Maximum parallel transfers** with the following transfer site protocols: AS2, FTP(S), HTTP(S), SSH, Generic-HTTP(S), SharePoint.

  Note When multiple transfer limitations are set, all of them apply but the strictest limitation takes priority over the rest. The following limitations may affect server initiated transfers:

  - Maximum Parallel Transfers - the limit per Transfer Site, described here.
  - Maximum number of parallel transfers - the limit for "Files Received from this Account or its Partners", specified in Basic Application or Advanced Routing Subscriptions
  - Maximum concurrent connections per host for outbound connections

- **Use Expression Language** - when checked, all possible checkboxes for boolean properties are switched to text fields. Applicable for Folder Monitor, FTP(S), HTTP(S), SSH (SFTP) and PeSIT Transfer Sites. In these text fields, you can type values or expressions for the required and the optional fields needed to define the transfer site.
  You can use expressions in the fields indicated by a **vertical yellow bar**.
- **Transfer Protocol** – one of the supported protocols: AS2, Connect:Direct, Folder Monitor, FTP (S), HTTP(S), SSH (SFTP), PeSIT, System to Human, or a protocol implemented using the file services interface. The protocol of an existing transfer site cannot be changed.
- **Additional attributes** – a group of fields that allows you to add (or remove) custom attributes as *attribute:value* pairs to use with your Transfer sites. This functionality is available at the bottom of the page on the *Add Transfer Site* page, regardless of the Transfer site type. See *Additional attributes* on page 759.
- **Custom properties** – these vary according to the selected transfer protocol. For more information about each protocol, see the respective subtopic:
  - *AS2 transfer sites* on page 542
  - *Connect:Direct transfer sites* on page 550
  - *Folder Monitor transfer sites* on page 557
  - *File services interface transfer sites* on page 554
  - *FTP(S) transfer sites* on page 562
  - *HTTP(S) transfer sites* on page 595
  - *PeSIT transfer sites* on page 603
  - *System to Human transfer sites* on page 624
  - *SSH transfer sites* on page 613
  - *Generic HTTP transfer sites* on page 568

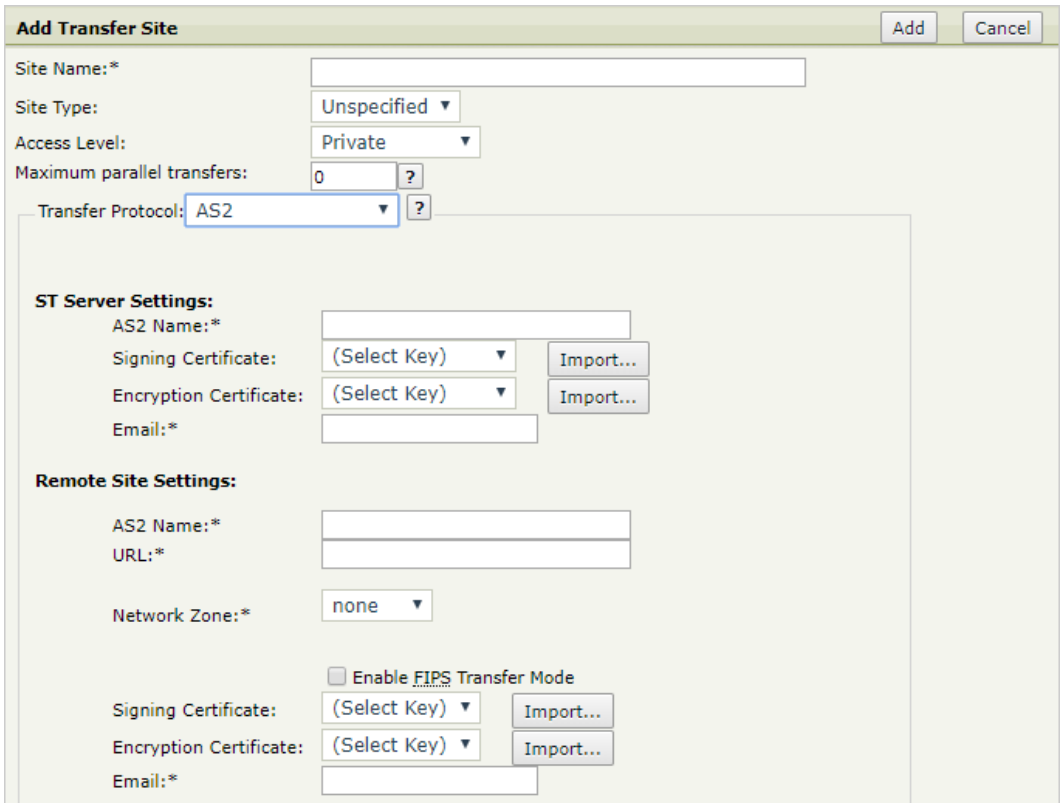
AS2 transfer sites

Although transfers use the AS2 protocol function in a different way than the other supported protocols, you can subscribe accounts with AS2 transfer sites to applications. Among the standard applications, the Site Mailbox and Standard Router applications are appropriate for an AS2 transfer site.

Unlike transfer sites for other transfer protocols, an AS2 transfer site is also used for transfers initiated by the remote AS2 site (considered client-initiated by SecureTransport). Only the fields marked with an asterisk (`*`) as required are needed to define the partnership to enable these transfers.

For detailed information about AS2 transfers, see *AS2 transfers* on page 1035.

The following is the *Add Transfer Site* page for a transfer site definition that uses the AS2 transfer protocol.



The following table describes the AS2 protocol options for defining a transfer site.

Field	Description
SecureTransport Server Settings	
AS2 Name*	The local partnership name, which the remote AS2 site uses to identify to this SecureTransport Server. Each AS2 transfer site for a user must have a unique AS2 Name. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Signing Certificate	(Optional) The alias that represents the server or partner certificate used to sign a message.
Encryption Certificate	(Optional) The alias that represents the server or partner certificate used to encrypt a message.
Email	The email address used to receive information from the remote AS2 site. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Remote Site Settings	

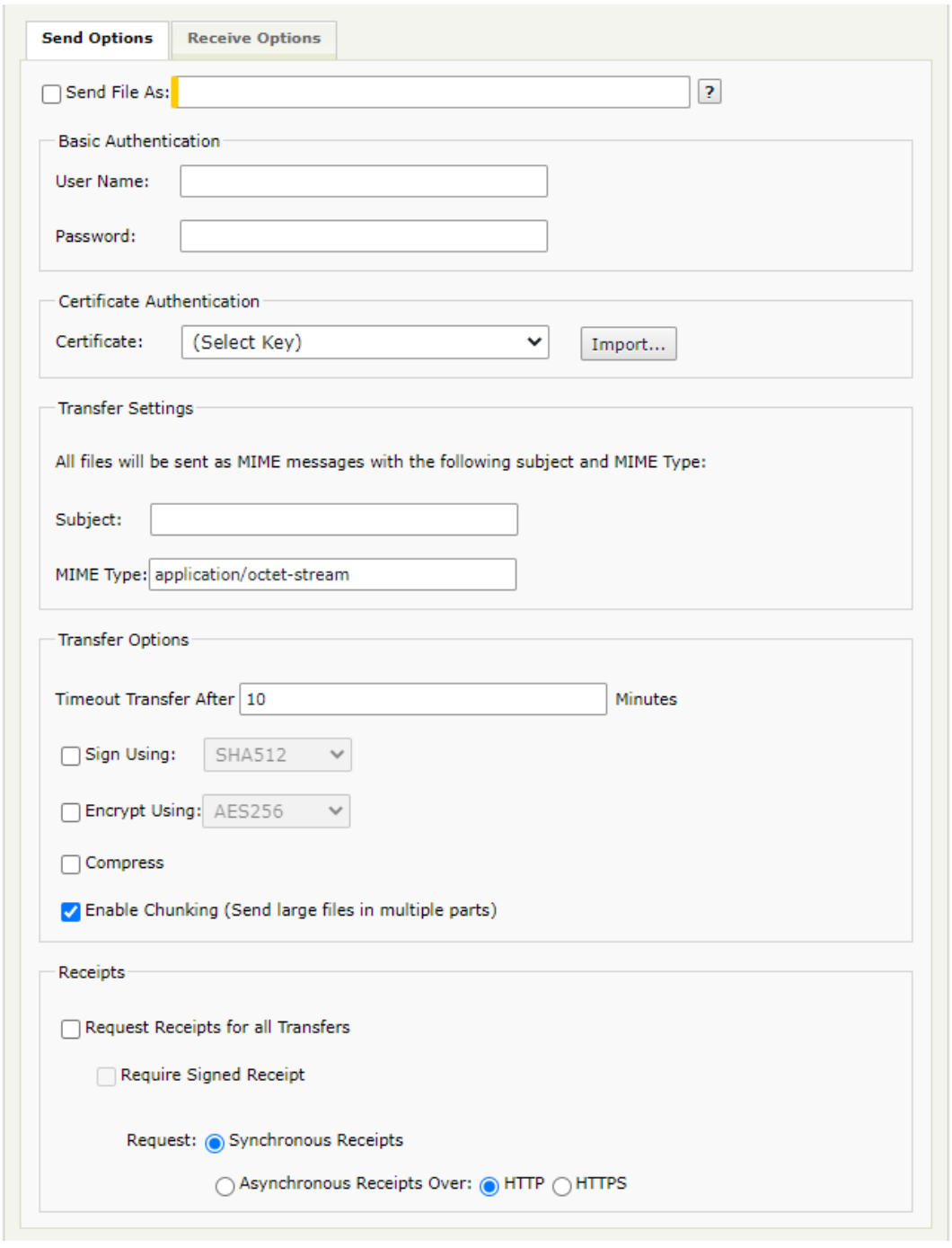
Field	Description
AS2 Name*	The remote partnership name, which the SecureTransport Server uses to identify to the remote AS2 site. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
URL	The URL used to access the remote site. For example, `https://as2.example.com:10443` ,`https://172.23.34.45:10443`, or `https://[FC00:1234:2345:3456::]:10443`. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Alternative addresses	The visibility of this option is controlled with the value set for the `TransferSite.AlternativeAddresses.retryPolicy` configuration option. It allows you to set a list of endpoints that act as backup alternatives to the configured URL and are particularly useful in cases of transfer failures. For mode details, see *Set Alternative addresses* on page 633.
Network Zone	The network zone that defines the proxies to use for transfers through this site. • Select **none** to connect directly to the partner AS2 server. • Select **any** to allow SecureTransport to select the proxy connection using a network zone that enables an HTTP proxy. • Select **Default** to use the default network zone proxy configuration. If no default network zone is defined, transfers from this transfer site fail. • Select a specific network zone to use the proxy configuration defined for that zone. For more information, see *Specify TM Server communication ports and IP address for protocol servers on SecureTransport Edge* on page 236.
Enable FIPS Transfer Mode	Restrict AS2 to use only FIPS 140-2 Level 1 certified cryptographic libraries. When you enable FIPS transfer mode, the panel expands with an additional field that lets you specify the desired set of cipher suites to be used in FIPS mode for server-initiated transfers through this site. By default, this set is populated with the cipher suites as defined in the `As2.FIPS.SIT.Ciphers` configuration option. You can add or remove cipher suites. The supported FIPS cipher suites from which you can select when adding a new one are listed in FIPS-compliant ciphers and cipher suites (login required). **Note** Both the sender and the recipient must use supported FIPS ciphers suites. Otherwise, the transfer will fail.
Signing Certificate	(Optional) The alias that represents the user or partner certificate used to sign a message from this site.
Encryption Certificate	(Optional) The alias that represents the user or partner certificate used to encrypt a message from this site.

Field	Description
Email	The email address used to receive information from SecureTransport Server. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.

* Each AS2 transfer site must have a unique combination of SecureTransport Server AS2 Name and Remote Site AS2 Name.

Transfer Settings: Send Options

This subtopic provides descriptions on the Send Options and Receive Options pages for AS2 transfer sites.



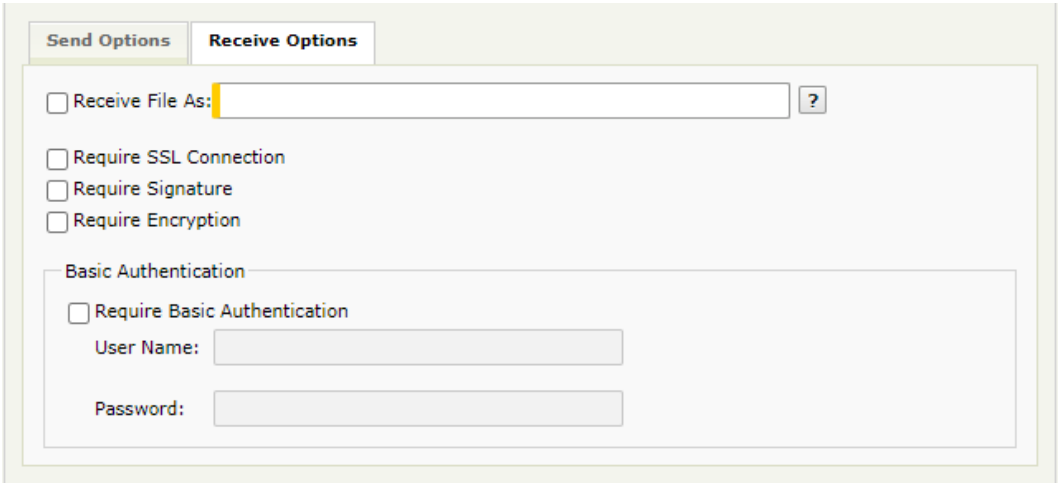
The following table describes the Send Options for an AS2 transfer site.

Field	Description
Send options	

Field	Description
Send File As	Select the checkbox to specify a file name. You can use the expression language to specify the criteria you want to match. The expression uses the criteria provided to create a new file name from the original file name. When you enter a new file name in this field, the AS2 message header uses the new name as the value for `original filename`.
Basic Authentication	Enter the username and password to be used for authentication to the remote AS2 server. To enable Basic Authentication when asynchronous MDNs are requested, in **Receive options** set up **Basic Authentication** with valid credentials that match those sent from the remote partner.
Client Certificate Authentication	A private certificate component for client authentication. You can either select a local or private certificate from the drop-down or import one. The certificate must be signed by a trusted CA. See *Import a trusted CA certificate* on page 56. If SecureTransport is sending files, you need to provide your trading partner with the public key corresponding to the private key specified here. If SecureTransport is receiving files, the partner's certificate must be imported either in the **Login** or the **Partner** tab of the account. The certificates in the **Partner** tab can be used for logins over AS2 and PeSIT only, while those in the **Login** tab can be used for all protocols but PeSIT. When asynchronous MDN are requested, trading partners must exchange their public keys. If both client certificate and basic authentication are enabled, SecureTransport will first match the transfer site according to AS2-From and AS2-To headers, then perform certificate authentication, and verify the user name and password.
Transfer Settings	
Subject	The MIME subject to be used for outgoing messages.
Mimetype	The MIME type to be used for outgoing messages. For example, `application/edi-x12`.
Transfer Options	
Timeout Transfer After `x` Minutes	The number of minutes after which a transfer is timed out if it is not successful.
Sign Using	The algorithm to be used to sign messages from this site.
Encrypt Using	The algorithm used to encrypt messages from this site. The RC2/40, RC2/64 and RC2/128 algorithms are not FIPS compliant.

Field	Description
Compress	Select this checkbox to enable compression.
Enable Chunking	Select this checkbox to enable chunking.
Receipts	
Request receipts for all Transfers	Select this checkbox to request receipts for all transfers.
Require Signed Receipt	If you select the **Request receipts for all transfers** checkbox, select the checkbox to require those receipts to be signed. In the User Name and Password fields, provide the credentials used
Request: Synchronous Asynchronous	Specify whether you want receipts to be synchronous or asynchronous. If you select asynchronous receipts, specify whether you want to receive those receipts via HTTP or HTTPS. If you request receipts via asynchronous HTTP and you specify an SSL connection in *Receive Options*, you receive receipts via HTTPS instead of HTTP.

Transfer Settings: Receive Options

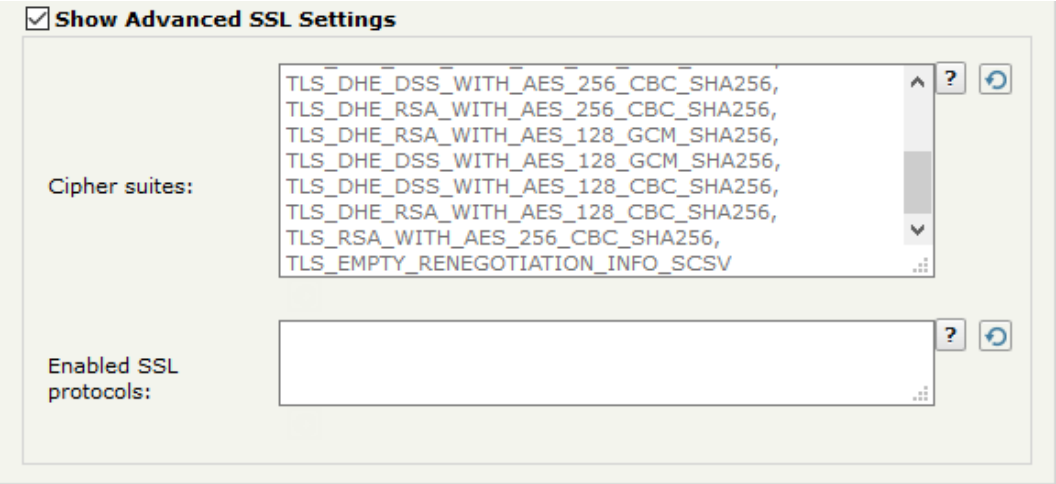


The following table describes the Receive Options for an AS2 transfer site.

Field	Description
Receive Options	

Field	Description
Receive File As	Select the checkbox to specify a file name. You can use the expression language to specify the criteria you want to match. The expression uses the criteria provided to create a new file name from the original file name when the transfer is received. You can use the SecureTransport-specific variable `${stenv.rawsource}` which takes the value from the `original filename` in the AS2 message header. See *Expression Language* on page 1104 for information on SecureTransport-specific variables.
Require SSL Connection	Select this checkbox to require an SSL connection for transfers received. If you request receipts via asynchronous HTTP and you specify an SSL connection, you receive receipts via HTTPS instead of HTTP.
Require Signature	Select this checkbox to require transfers received to be signed.
Require Encryption	Select this checkbox to require transfers received to be encrypted.
Require Basic Authentication	When selected, Basic Authentication is required for all incoming transfers and MDNs sent to this transfer site partnership. In the User Name and Password fields, enter the credentials to use for authentication.

Advanced SSL Settings



The following table describes the Advanced SSL Settings for an AS2 transfer site.

Field	Description
Show Advanced SSL Settings	

Field	Description
Cipher suites	The set of cipher suites available with the current AS2 transfer site for secure SIT connection. By default this set is populated with the cipher suites as defined in the `As2.SIT.Ciphers` configuration option. To reset to default values, click the button next to the tooltip.
Enabled SSL protocols	The available SSL protocols for secure SIT connection with the current AS2 transfer site. By default this option uses the SSL protocols as defined in the `As2.SIT.EnabledProtocols` configuration option. To reset to default values, click the button next to the tooltip.

Note Use a subscription to a Basic application or a Site Mailbox application to process files received by an AS2 transfer site.
When using asynchronous receipts for outgoing AS2 transfers, post-transmission actions execute, even if the AS2 transfer has failed. This occurs because the transfer initially reports success, triggering the post-transmission action. After the post-transmission action is triggered, an asynchronous failure message is returned, causing the transfer to fail.

Connect:Direct transfer sites

The Connect:Direct transfer sites are not built into SecureTransport. To be able to create and modify transfer sites that use the Connect:Direct protocol, you need to perform the following tasks:

1. Install CDJAI (Connect:Direct Application Interface for Java).
2. Enable file transfers via Connect:Direct.
3. Set the server configuration options for Connect:Direct file transfers.
4. Create a Connect:Direct transfer site.

Install CDJAI

The IBM Sterling Connect:Direct Application Interface for Java enables SecureTransport to connect to Connect:Direct servers.

To install it, you need the `CDJAI.jar` file, which is provided with the Connect:Direct Java API. Follow the steps:

1. Stop SecureTransport by running `<FILEDRIVEHOME>/bin/stop_all`.
2. Copy the `CDJAI.jar` file to the `<FILEDRIVEHOME>/lib/jars/external` directory on the server running SecureTransport.
3. Edit the `<FILEDRIVEHOME>/bin/start_tm_console` file to set the correct path to the `CDJAI.jar` file in the `CLASSPATH` parameter.
4. If you are using the embedded database, run `<FILEDRIVEHOME>/bin/start_db`.
5. Start the Administration Tool server by running `<FILEDRIVEHOME>/bin/start_admin`.

The certificate authentication to Connect:Direct servers requires CDJAI version 1.1.00 Fix 000026, as well as SecureTransport 5.5 October 2020 Update or later. If you're running an older CDJAI version, you must replace the jar file following the steps:

1. Stop all services.
2. Replace the jar file.
3. Restart all services.

Next, you need to enable the file transfers via Connect:Direct and set the server configuration options related to them.

Enable file transfers via Connect:Direct

1. Log in to the Administration Tool, and go to **Setup > TM Settings**.
2. Enable the `ConnectDirectTransfer` rule package.

Set the server configuration options for Connect:Direct file transfers

1. Go to **Operations > Server Configuration**.
2. Search for the `ConnectDirectTransferAgent` parameters.
3. Set `ConnectDirectTransferAgent.transfersFolder` to the full path of the directory for the SecureTransport Server to use for the Connect:Direct transfers. The directory must be shared between the SecureTransport and the Connect:Direct servers, and the path should be the same on both. Verify that SecureTransport has full permissions for the directory.

   Note The directory path is not relative to `<FILEDRIVEHOME>`. Specify a full absolute path from `/` (root) in UNIX or `C:\` or another volume on Windows.

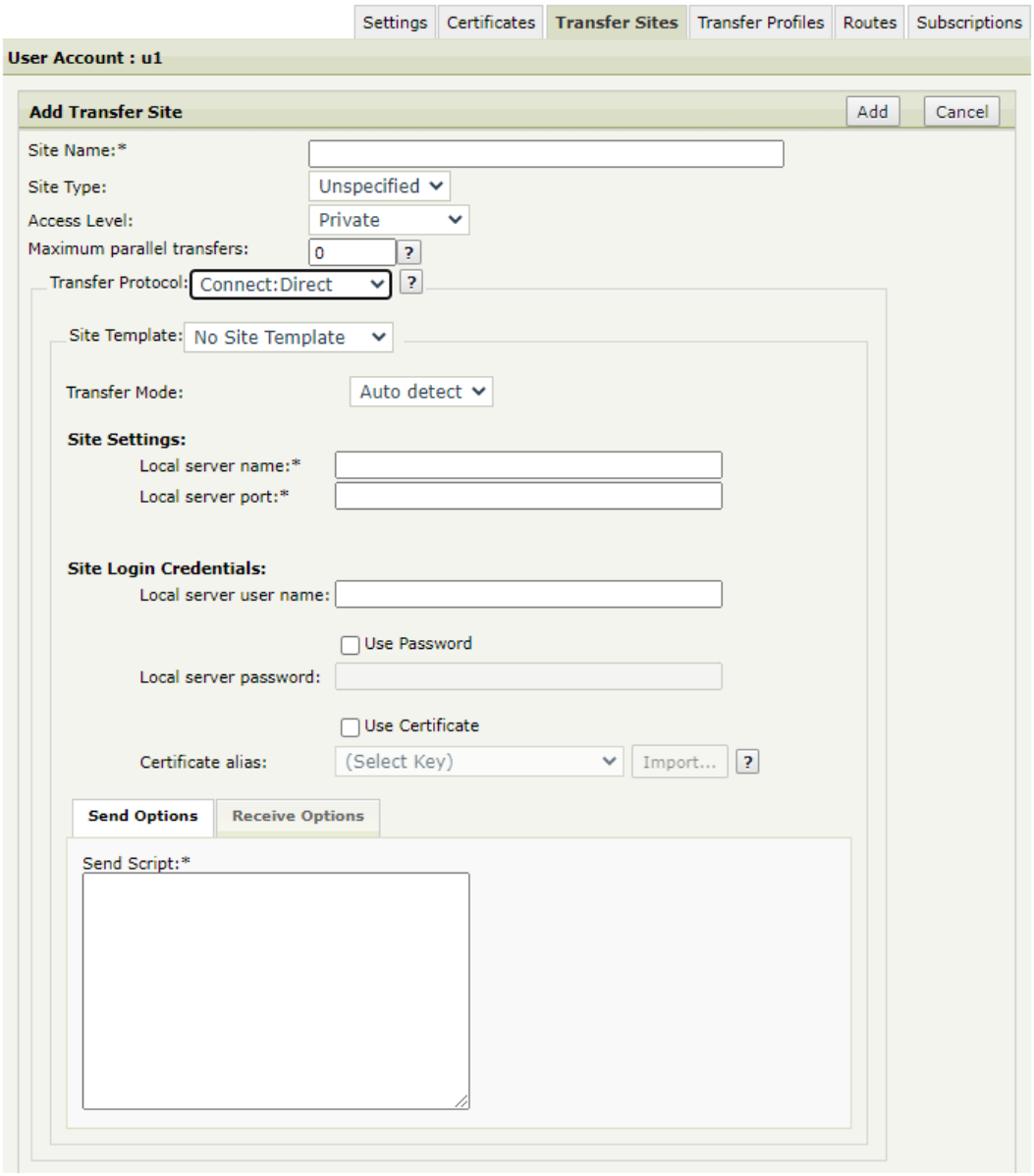
4. To manage purging of the Connect:Direct folder, use the `ConnectDirectTransferAgent.transfersFolder.purge` server configuration option. By default, it is set to **true**, which means that the folder used for the Connect:Direct transfers will be purged on Transaction Manager startup. When set to **false**, no purging is performed.
5. Set `ConnectDirectTransferAgent.commandTimeout` to the interval in seconds that SecureTransport waits before the transfer times out.
6. Start SecureTransport by running `<FILEDRIVEHOME>/bin/start_all`.

Note In order to obtain error messages for failed command execution, SecureTransport executes the SELECT STATISTICS command. The Connect:Direct user must have permissions to select statistics (`cmd.selstats:a`)

Create a Connect:Direct transfer site

In the transfer site definition, select Connect:Direct as Transfer Protocol. For more information, see *Manage transfer sites* on page 628. To create a Connect:Direct transfer site from a template, see *Use a site template to define a transfer site* on page 741.

The following is the *Add Transfer Site* page for a Connect:Direct transfer site.



The following table describes the Connect:Direct protocol options for a transfer site:

Field	Description
Transfer Mode	Sets the file transfer mode. Valid values: ASCII, Binary, and Auto-detect. The default value is Auto-detect, meaning SecureTransport automatically determines the proper transfer mode based on the file content type. For more information, see *Transfer mode for server-initiated transfers* on page 764.
Site Settings	

Field	Description
Local server name	Specifies the domain name or IP address of the local server. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Local server port	Specifies the port assigned to the local server. You cannot enter spaces-only values in this field.
Site Login Credentials	
Local server user name	The username used to log in to the local server.
Select the authentication method	• Password authentication If the local server uses a password, select **Use Password** and enter the password in the field. • Certificate authentication To configure a SecureTransport transfer site to connect to a Connect:Direct server by using a certificate: 1. Exchange the CA certificates between the SecureTransport and the Connect:Direct server: the trusted root certificate file of the Connect:Direct server should be exported, and imported as a trusted CA in SecureTransport; the trusted root certificate file of the SecureTransport server should be exported, and imported on the Connect:Direct server. 2. Generate a local or a private certificate in SecureTransport whose common name field matches the Connect:Direct local user that is going to be used to log in to the Connect:Direct server. 3. Import the certificate in the Connect:Direct server. 4. In the transfer site configuration, select the **Use Certificate** checkbox and specify the **Certificate Alias** used for connecting to the Connect:Direct server. You can either select the private certificate you generated at *Step 2* from the drop down or import a certificate. When certificate authentication is enabled, the connection uses the default TLS 1.2 protocol and compatible ciphers. **Note** After reverting the SecureTransport October 2020 Update, the Connect:Direct certificate-based authentication feature will not work, although the certificate placeholder remains visible on the transfer site definition page for sites created using a site template.
Send Options	

Field	Description
Send Script	Specifies the Connect:Direct process to execute when uploading a file to a remote site. You must provide a script for either the Send Options or the Receive Options. This field must contain a valid Connect:Direct process language script. You can use expression language variables such as `${stenv.target}` in the script. For example, you can use the script field to execute a copy command. The remote server you are calling must be identified by its alias in the script. To correctly identify the file name in a script you must use the variable `${cd_transfer_file}`. The variable is required because the file names might not be known at the time you write the script. When creating an upload script you must use `${cd_transfer_file}` instead of the file name of the file being uploaded. You cannot enter spaces-only values in this field.
Receive Options	
Receive Script	Specifies the Connect:Direct process to execute when downloading a file from a remote site. You must provide a script for either the Send Options or the Receive Options. This field must contain a valid Connect:Direct process language script. You can use expression language variables such as `${stenv.target}` in the script. For example, you can use the script field to execute a copy command. The remote server you are calling must be identified by its alias in the script. To correctly identify the file name in a script you must use the variable `${cd_transfer_file}`. The variable is required because the file names might not be known at the time you write the script. When creating a download script you must use `${cd_transfer_file}` to specify the directory where downloaded files are saved. When downloading a single file, use `${cd_transfer_file}<path_separator><file_name>`. For example, `${cd_transfer_file}/xls_sheet.xls`. You cannot enter spaces-only values in this field.

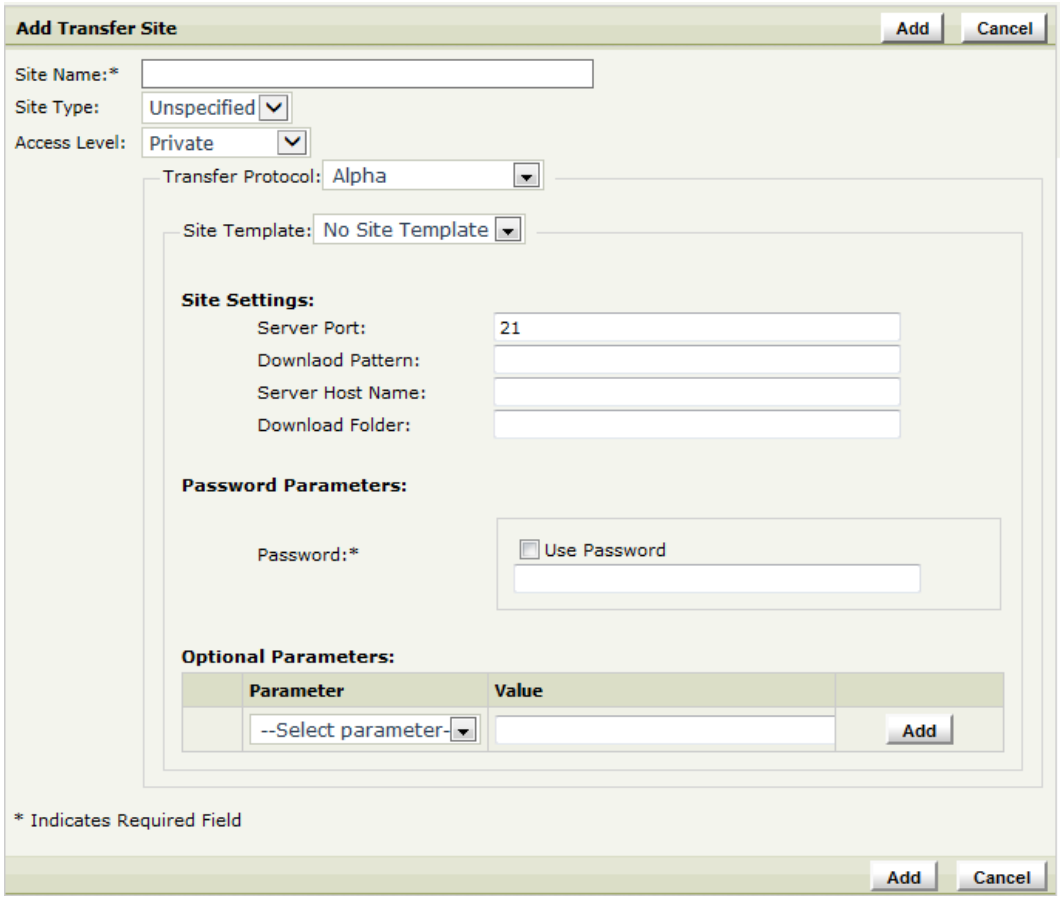
Note The **Send Script** and **Receive Script** accept regular expressions. For more information on writing Connect:Direct scripts, refer to the Connect:Direct documentation.

You can use a site template to define a Connect:Direct transfer site. For more information, see *Site templates* on page 737.

File services interface transfer sites

Each file services interface protocol can have different required and optional parameters as configured by the developer in the file services interface protocol registry file. The developer who created the transfer site protocol can provide you with a list of required and optional parameters with descriptions and valid values.

The following is the *Add Transfer Site* page for a transfer site definition that uses the file services interface transfer.



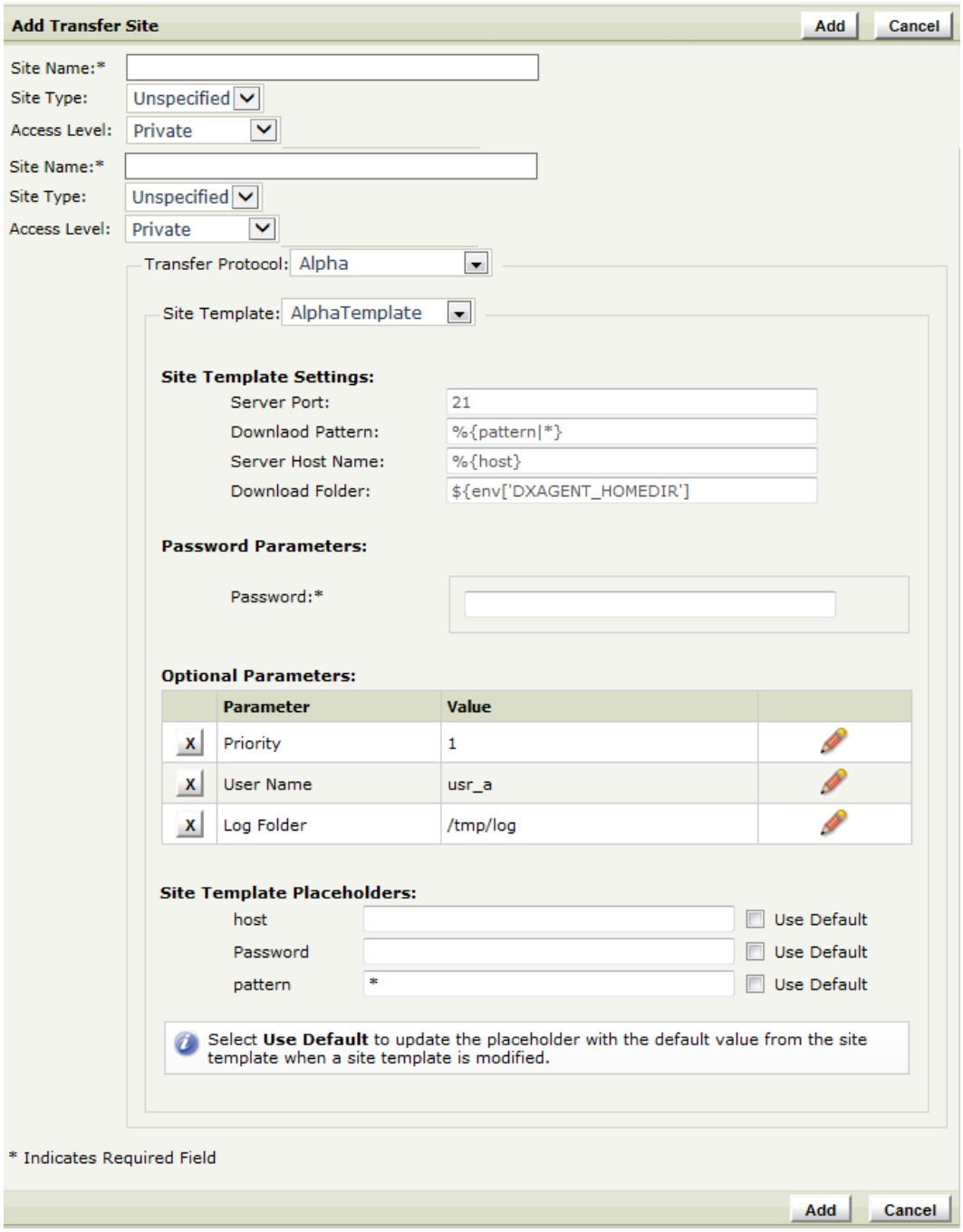
The fields under **Site Settings** are required. If they have default values in the file services interface registry file, the Administration Tool displays those values when the transfer protocol is selected.

To specify a password, select **Use Password** and type the password.

The fields in the **Optional Parameters** table are not required.

- To add an optional parameter, select its name from the **Parameter** list, type a value in the **Value** column, click Add.
- To edit a parameter value, click the Edit icon ( ) type a new value in the **Value** column, and click the Save icon ( ).
- To delete a parameter, click the Delete icon ( ).

You can create a file services interface protocol transfer site using a site template. If you select a template from the **Site Template** list, the transfer site page displays the values from the site template and the list of placeholders you can specify.



You cannot edit the values in the **Site Template Setting** or the **Optional Parameters**. When you select a site template, the **Site Template Placeholders** fields get the default values from the template. You can type values for these placeholders referenced in the other fields. Select **Use Default** to always use the current default value from the template if it is updated.

You can use a site template to define a file services interface protocol transfer site. For more information, see *Site templates* on page 737.

For more information about file services interface protocols for receiving files from other systems, see *File services interface transfers* on page 1039.

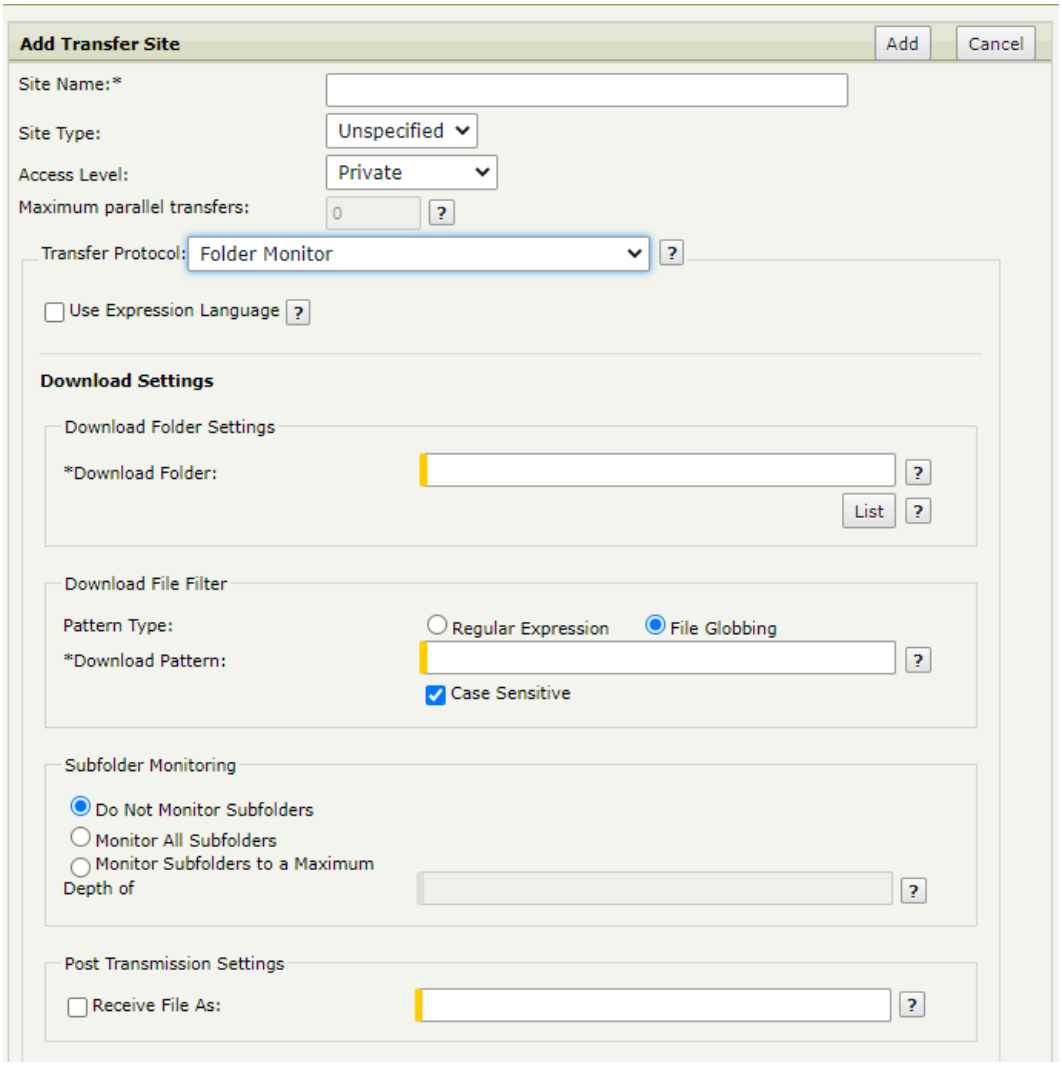
Folder Monitor transfer sites

Create a Folder Monitor transfer site to monitor a designated folder (and optionally its subfolders) for specific files and move them to another location. Once the transfer site is used in a subscription, SecureTransport starts monitoring the folder at fixed intervals as defined in the `FolderMonitor.heartbeatInterval` server configuration option. The default value is `5` seconds.

The Folder Monitor settings are divided into *Download settings* and *Upload settings*.

Download settings

Specify the monitoring area by selecting a download folder (including or excluding its subfolders), and define the name pattern based on which files will be moved to another location.



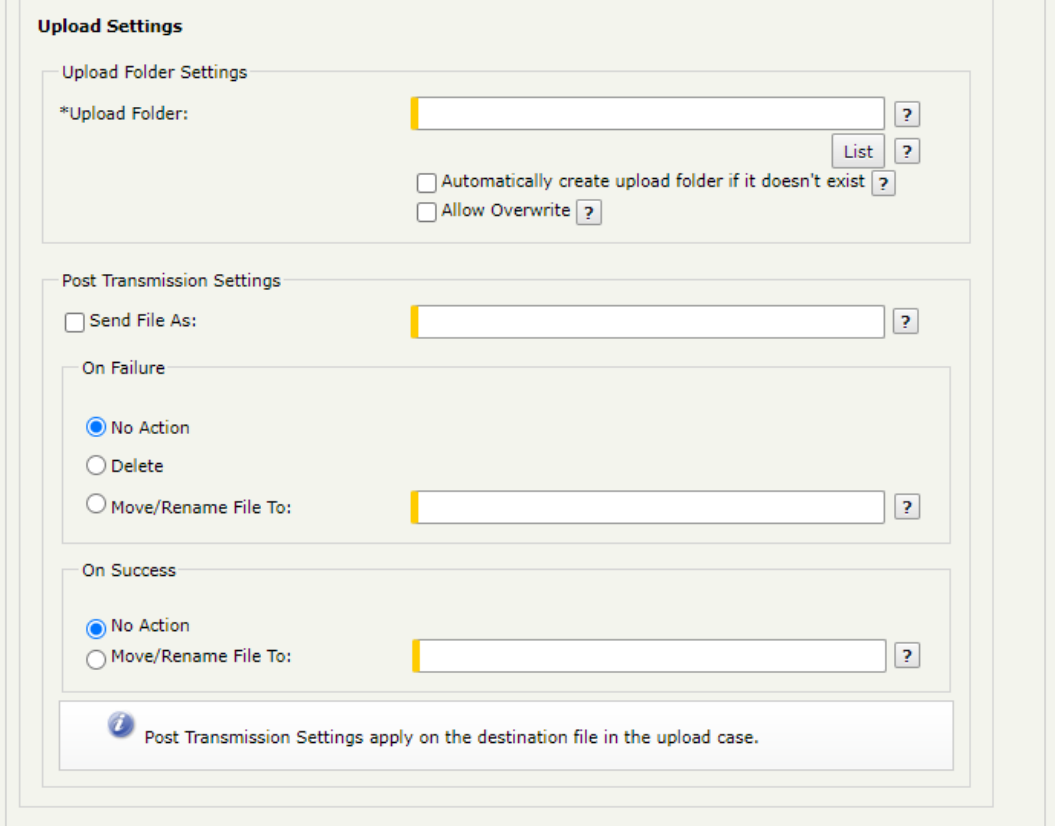
The following table describes the download settings for a Folder Monitor transfer site.

Field	Description
Download Folder Settings	
Download Folder	The full path to the folder containing files that will be moved to the upload folder. To see the list of the download folder's files and subfolders, click **List**. For more details, see *List the contents of the Upload or Download folder on page 630*. EL expressions are supported. For example, the following expression appends the current date to the folder name: `folder_${date("yyyyMMdd")}` On the 30.10.2022, for example, the result would be `folder_20221030`. Specifics: • The download folder should not be set to the root (/) folder of the operating system, because that can corrupt the whole operating system. • `${DXAGENT_TRANSFERSAPI_*}` expressions are not supported. • You cannot enter spaces-only values in this field.
Download File Filter	
Pattern Type	Specify whether you want to use glob pattern matching or regular expression syntax to match the strings specified for the download pattern field.
Download Pattern	Specify a file name pattern to identify the files to be downloaded. For regular expression syntax, see *Regular expressions on page 1117*. For globbing syntax, see *Globbing in SecureTransport on page 1122*. The download pattern is evaluated when the transfer site is being executed. Examples: • The glob `*_${date("yyyyMMdd")}.txt` will be evaluated using the current date of the transfer site execution, e.g., on the 30.10.2022, it will match all files that end with `_20221030.txt`. • The regex `*[a-z]_${date("yyyyMMdd")}.txt` will also be evaluated using the current date, e.g., on the 30.10.2022, it will match all files that start with any combination of letters from `a` to `z` and end with `_20221030.txt`.
Case Sensitive	Select this checkbox to enable case-sensitive file name matching. **Note** This setting applies only when the Pattern Type is set to **File Globbing**.
Subfolder Monitoring	
Do Not Monitor Subfolders	Apply the download pattern to the download folder only.
Monitor All Subfolders	Apply the download pattern to the download folder and all subfolders.

Field	Description
Monitor Subfolders to a Maximum Depth of ___	Apply the download pattern to the download folder and subfolders to the specified depth. For example, if the maximum depth is `2`, the download pattern is applied to the download folder and its immediate subfolders.
Subfolder Name Pattern Type	Select the pattern matching syntax: **Regular Expression** or **File Globbing**.
Subfolder Name Pattern	(Displayed only when monitoring subfolders) Specify a folder name pattern to identify subfolders for monitoring. You cannot enter spaces-only values in this field.
Case Sensitive	(Displayed only when monitoring subfolders) Select this checkbox to enable case-sensitive folder name matching. **Note** This setting applies only when the Subfolder Name Pattern Type is set to **File Globbing**.
Post Transformation Settings	
Receive File As	Select the checkbox to specify a file name. You can use the expression language to specify the criteria you want to match. The expression uses the criteria provided to create a new file name from the original file name when the transfer is received. You can use the SecureTransport-specific variable `${stenv.site_target}` which takes the value from the remote file path. See *Expression Language on page 1104* for more information on SecureTransport-specific variables.

Upload settings

Specify the location where files from the download folder are moved to.



The following table describes the upload settings for a Folder Monitor transfer site.

Field	Description
Upload Folder	The full path to the folder where incoming files from the download folder are moved to. You cannot enter spaces-only values in this field. To see the list of the upload folder's files and subfolders, click **List**. For more details, see *List the contents of the Upload or Download folder on page 630*. **Note** Making the upload folder the same as the download folder may lead to an infinite loop condition when the transfer site is used.
Automatically create upload folder if it doesn't exist	The upload folder will be automatically created if it doesn't exist. The automatically created folder will be owned by the user running the SecureTransport TM Server process.
Allow Overwrite	Taken into account when the site is used by the Send To Partner step. If checked, the value of the upload folder will be overwritten by the value of the *Overwrite Upload Folder* field. For more details, see *Advanced Routing* on page 864

Post-transmission actions

Post-transmission actions are file operations - such as rename, move and delete- that are performed after a file transfer ends and can be different based on the transfer status. For more information, see *Set post-transmission actions in transfer sites* on page 634.

Folder Monitor specifics

The Folder Monitor protocol differs from the other supported protocols in the following aspects:

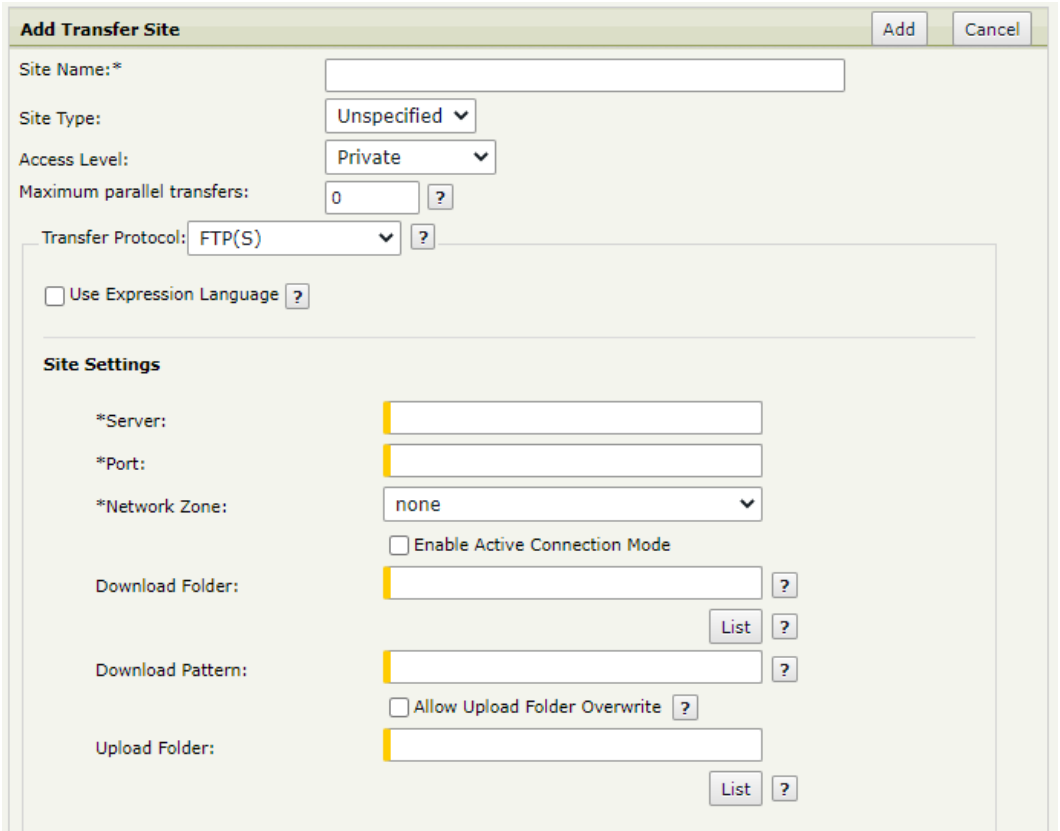
- The folders listed in `<FILEDRIVEHOME>/conf/unsafe.paths.conf` file (and their sub-folders) cannot be specified in the **Upload Folder** and **Download Folder** fields.
- The paths, specified in the **Upload Folder** and **Download Folder** fields, must be full system paths (not relative paths), such as: `/tmp/folder_name/opt/TMWD/folder_name/`
- The files uploaded/moved into the specified upload folder are owned by the root user of the system.
- If SecureTransport is installed as a non-root deployment, the files in upload folder are owned by the user running the SecureTransport Server.
- If you configure the account to impersonate a user, the impersonated user must have full rights to the directories specified in the Folder Monitor.
- If you upload or move a file in the specified upload folder and such a file already exists there, it is overwritten without a prompt.
- If SecureTransport is installed as a non-root deployment, the user running the SecureTransport Server must have the necessary permissions to overwrite the file.
- When the Folder Monitor starts to monitor the download folder, the download pattern is applied to all the files in the folder, even if they existed in the download folder before the Folder Monitor began monitoring.
- If SecureTransport is installed as a non-root deployment, the user running the SecureTransport Server must have the necessary permissions to write to the upload folder.
- SecureTransport does not support placing the download folder and the upload folder under a repository encryption user home folder.
- If you create a service account and a Standard Router application that uses a Folder Monitor transfer site, and you specify the same directory for sending and receiving messages, file transfers fail.
- A Folder Monitor transfer site can be used to receive messages for only one subscription.
- The names of the download folder and the upload folder of a Folder Monitor transfer site cannot contain two or more of the following characters in a sequence: `( ) _ - + = { } ~ ! @ # $ % ^ & ; " ” ' ’`. For example, `folder_name` is supported, but `folder_+name` is not supported.

- The name of a file processed by a Folder Monitor transfer site cannot contain two more of the following characters in sequence: < > | : ? " * / \ % [ ] ~ (at the beginning of the file only).
- Two Folder Monitor transfer sites cannot have the same download folder and download pattern.
- The Folder Monitor service runs on the primary server in a Standard Cluster (SC). If that server fails, the Folder Monitor service automatically fails over to the new primary server.
- The Folder Monitor service runs on one server in an Enterprise Cluster (EC). If that server fails, the Folder Monitor service automatically fails over to the server in the cluster with the Transaction Manager that has been running the longest.

FTP(S) transfer sites

The FTP(S) transfer site uses the FTP protocol to connect and transfer files across servers. SecureTransport accounts can be subscribed to any of the built-in applications and the FTP(S) site can be configured to push to or pull from a destination, or both.

The following image presents a snippet of the *Site Settings* pane for an FTP(S) transfer site.



General settings

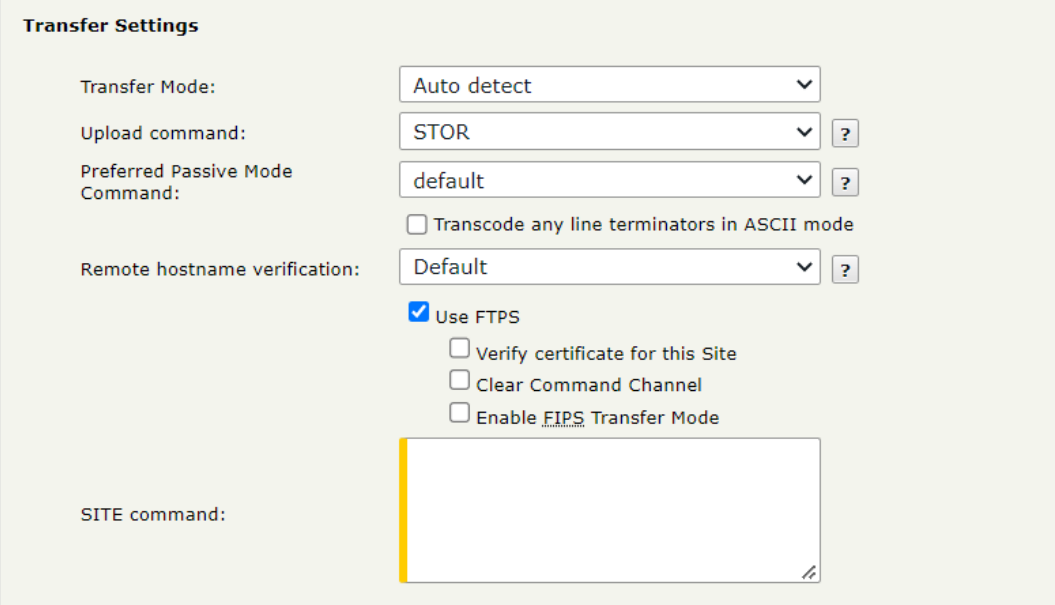
The following table describes the general options for an FTP(S) transfer site.

Field	Description
Site Settings	
Server	The host name or IP address of the remote server to connect to for file transfers. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Port	The port on the remote server to be used for file transfers. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Alternative addresses	The visibility of this option is controlled with the value set for the `TransferSite.AlternativeAddresses.retryPolicy` configuration option. It allows you to set a list of endpoints that act as backup alternatives to the configured Server-Port site settings and are particularly useful in cases of transfer failures. For mode details, see *Set Alternative addresses* on page 633.
Network Zone	The network zone that defines the proxies to use for transfers through this site. • Select **none** to connect directly to the remote FTP server. • Select **any** to allow SecureTransport to select the proxy connection using a network zone that enables an SOCKS5 proxy. • Select **Default** to use the default network zone proxy configuration. If no default network zone is defined, transfers from this transfer site fail. • Select a specific network zone to use the proxy configuration defined for that zone. For more information, see *Specify TM Server communication ports and IP address for protocol servers on SecureTransport Edge* on page 236.
Enable Active Connection Mode	Determines whether passive or active connection mode is used by SecureTransport for server-initiated transfers over FTP. When selected, Active FTP is used.
Download Folder	The folder on the remote server from which the file is transferred. You can use an EL expression to append dates. For example, `folder_${date("yyyyMMdd")}`.The download folder will then be evaluated using the date of the transfer site execution, for example, `folder_20210130`. To see the list of the folder's files and subfolders, click **List**. For more details, see *List the contents of the Upload or Download folder* on page 630.

Field	Description
Download Pattern	Specify a file name pattern, using wildcard characters or a regular expression, to identify files to be downloaded. For regular expression syntax, see *Regular expressions* on page 1117. For globbing syntax, see *Globbing in SecureTransport* on page 1122. The download pattern is evaluated during the transfer site execution.
Allow Overwrite	Taken into account when the site is used by the Send To Partner step. If checked the value of "Upload folder" will be overwritten with the value of "Overwrite upload folder". For more details see *Advanced Routing* on page 864.
Upload Folder	The folder on the remote server to which files are transferred. To see all the files and subdirectories in it, click **List**. For more details, see *List the contents of the Upload or Download folder* on page 630.

Transfer Settings

The *Transfer Settings* options allow you to define various transfer settings with your current transfer site.



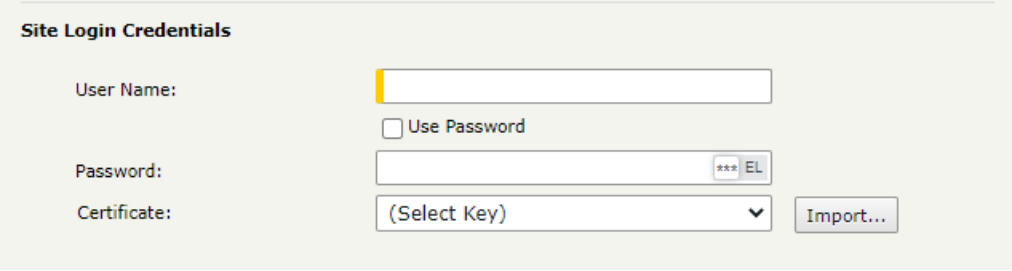
Field	Description
Transfer Settings	

Field	Description
Transfer Mode	Specifies whether data is transferred as ASCII or binary. You can also choose to have SecureTransport automatically determine the correct transfer mode. For more information about automatically determining transfer mode, see *Transfer mode for server-initiated transfers* on page 764.
Upload command	Defines the FTP command to be used in requests when server-initiated transfers are executed: **STOR** (default) - select to use the *STOR* command for server-initiated transfers. Saves Data and Replaces an Existing File **APPE** - select to use the *APPE* command for server-initiated transfers. The upload command is reported to Axway Sentinel and displayed in the Protocol Parameter attribute.
Transcode any line terminators in ASCII mode	When checked, it forces SecureTransport to transcode any sequence of line terminators when ASCII mode is used. In case of a BINARY mode transfer, no action is performed. When unchecked, it forces SecureTransport to add an extra CR to the line endings of the transferred file.
Remote hostname verification	From SecureTransport 5.5-20220825 onwards, administrators can bypass enforcement of matching remote host for the control and data connection by disabling remote hostname verification at the transfer site or global level. The global policy is set via the configuration option `Ftp.HostNameVerificationEnabled`. • On fresh installations of SecureTransport 5.5-20220825 or later, the option is set to `true` by default, meaning SecureTransport checks whether the remote host of a data connection is the same as the host to which the control connection is attached. If the IP addresses of the control and data connection do not match, the data connection for the transfer will be rejected. • On upgraded instances, `Ftp.HostNameVerificationEnabled` is set to `false`. Using the **Remote hostname verification** drop-down in a transfer site configuration, you can override the global policy and enable or disable the check for the particular transfer site only. If you choose **Default**, the transfer site will use the global policy.
Use FTPS	Deselect to use FTP instead of FTPS.
Verify certificate for the Site	Select to verify that the remote system is trusted. This option is displayed when **Use FTPS** is selected.

Field	Description
Clear Command Channel	Select to accept and process a Clear Command Channel subcommand. If the user is authorized to perform the command, send a confirmation message, and change the control connection transmission mode to clear text. This option is displayed when **Use FTPS** is selected.
TLS Shutdown on CCC	Perform a TLS shutdown upon receiving a Clear Command Channel subcommand. This option is displayed when **Clear Command Channel** is selected. **Note** When closing a TLS connection, such as when issuing a CCC command, each party is required to send a `close_notify` before closing the connection. This is mandated by RFC 2246. If both the client and server do not acknowledge that the TLS connection is ending they may be susceptible to a TLS truncation attack. From a security standpoint, Axway recommends that both TLS shutdowns be checked when configuring the transfer site CCC option. When performing FTP transfers to a remote SecureTransport Server, you must configure `Ftp.CCC.TlsShutdownInitiator` for the server. As a result the client sends *Close notify* and the server responds with *Close notify*, the server-initiated transfer is successful, and the partners are not susceptible to a TLS truncation attack.
Enable FIPS Transfer Mode	Restrict FTPS to use only FIPS 140-2 Level 1 certified cryptographic libraries. This option is displayed when **Use FTPS** is selected. When you enable FIPS transfer mode, the panel expands with an additional field where you specify the desired set of cipher suites to be used in FIPS mode for server-initiated transfers through this site. By default, this set is populated with the cipher suites as defined in the `Ftps.FIPS.SIT.Ciphers` configuration option. You can add or remove cipher suites. The supported FIPS cipher suites from which you can select when adding a new one are listed in FIPS-compliant ciphers and cipher suites (login required). **Note** Note that both the sender and the recipient must use supported FIPS ciphers suites. Otherwise, the transfer will fail.
SITE command	Enter a SITE command. You use this command to provide services specific to your system that are not available as FTP commands. EL expressions are supported.

Site Login Credentials

The Site Login Credentials options allow you to define credentials and/or add a certificate for login to the FTP(S) server.



Field	Description
Site Login Credentials	
User Name	The user name to log in to the FTP server. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Use Password	Select to use a password to log into the FTP server.
Password	Password used to log in to the FTP server. Using the toggle provides the ability to switch from literal password to Expression Language input.
Certificate	A private certificate for SecureTransport to use to log in to the FTP server. You can select or import a certificate. This field is displayed when **Use FTPS** is selected. When **Use Expression Language** is enabled, you can set the certificate dynamically by choosing the scope (account or server level) and providing a valid expression that will be evaluated to the name of an available certificate. By default, the usage of expired X509 certificates is allowed for SIT transfers. To forbid it, set the `SIT.allowExpiredCertificates` to `false`.

Post transmission actions

Post-transmission actions are file operations - such as rename, move and delete- that are performed after a file transfer ends and can be different based on the transfer status. For more information, see *Set post-transmission actions in transfer sites* on page 634.

Advanced SSL Settings

Advanced SSL settings allow you to define Cipher suites and SSL protocols with your current FTPS transfer site. Select **Show Advanced SSL Settings** to expand the pane with available options.

Field	Description
Show Advanced SSL Settings	

Field	Description
Cipher suites	The set of cipher suites available with the current FTP(S) transfer site for secure SIT connection. By default, this field is prefilled with the cipher suites as defined in the `Ftps.SIT.Ciphers` configuration option. To set a custom list of cipher strings for the transfer site, click in the **Cipher suites** field and edit as needed. To revert to the default list, click the **Reload** button.
Enabled SSL protocols	The available SSL protocols for secure SIT connection with the current FTP(S) transfer site. By default, this option uses the SSL protocols as defined in the `Ftps.SIT.EnabledProtocols` configuration option. To set a custom list of cipher strings for the transfer site, click in the **Enabled SSL protocols** field and edit as needed. To reset to default values, click the **Reload** button.

For more information about the available security settings, see *Secure your transfer site with SSL/TLS on page 631*.

Supported Active / Passive FTP(S) connections

This table describes the supported Active/Passive FTP(S) connection modes for client/server-initiated transfers over FTP(S).

FTP Exchange type	Active FTP mode supported	Passive FTP mode supported
Client initiated via Edges	Yes	Yes
Server initiated via Edges	No	Yes
Server initiated - no Edges/direct connection	Yes	Yes

Generic HTTP transfer sites

The Generic HTTP transfer site enables file exchange via HTTP protocol with third-party partners over an authenticated connection.

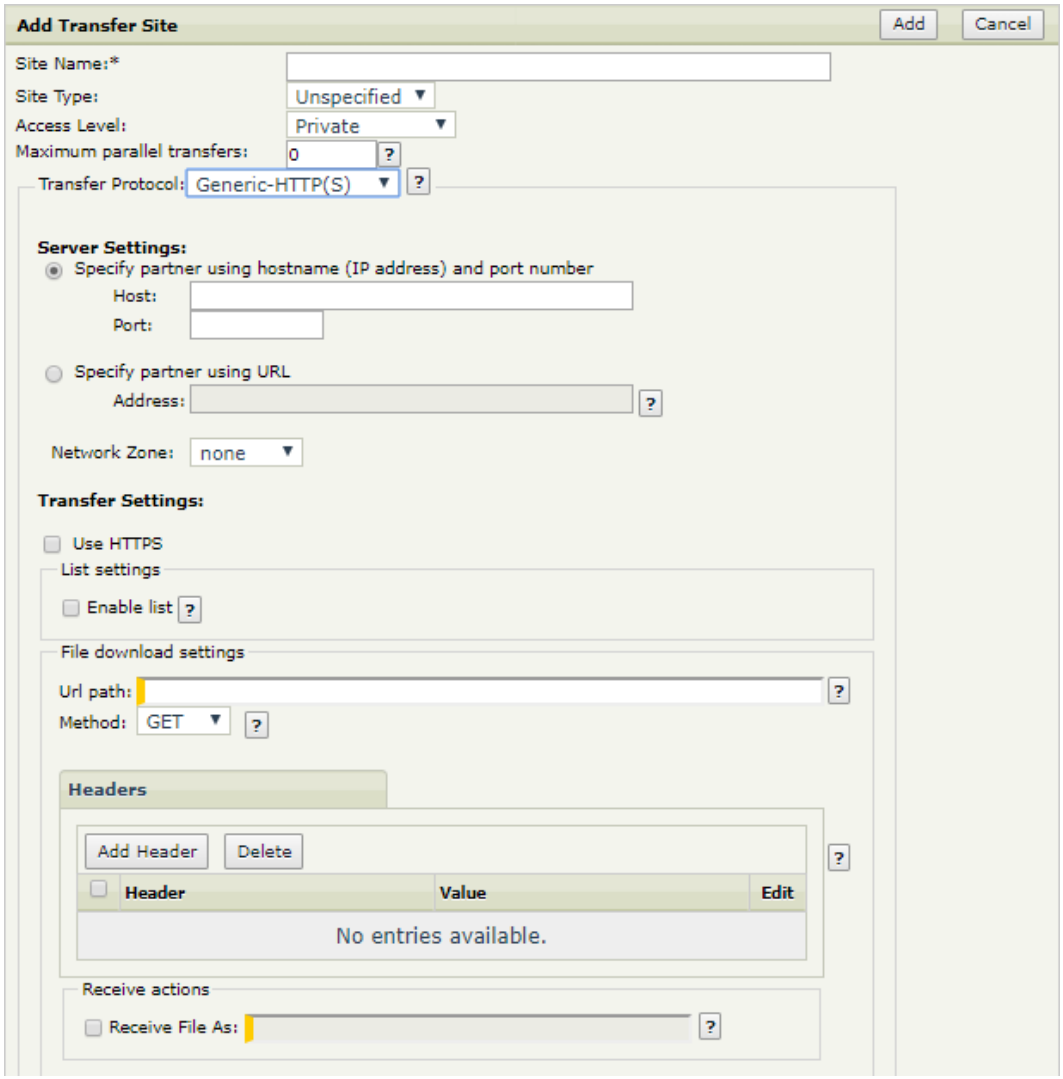
The supported authentication methods are:

- Basic – The client provides a user ID or user ID and password when exchanging files.
- Form-Based – The client will send a request to a remote HTTP server to obtain an authentication cookie.

- Certificate – A HTTPS client certificate is used for mutual authentication. The client certificate can be used in combination with basic or form-based authentications.

The Generic HTTP transfer site can be used with the Basic and Advanced Routing applications for push and pull server-initiated transfers.

The following is the *Add Transfer Site* page for a transfer site definition that uses the Generic HTTP transfer protocol.



The following sections describe the Generic HTTP transfer site configuration options, provide basic sample push and pull flow configurations for Generic HTTP transfer sites, and information on the limited expression language supported by Generic HTTP transfer sites:

- *Settings for Generic HTTP transfer sites* on page 570
- *Sample configuration — List files and download* on page 585
- *Sample configuration — Download file* on page 587
- *Sample configuration — Upload file* on page 589

- *Sample configuration — Push file to SecureTransport user using Form Authentication* on page 592
- *Supported expression language* on page 594

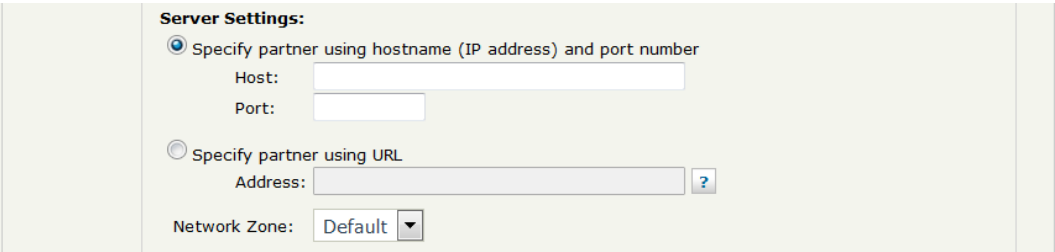
Settings for Generic HTTP transfer sites

The configuration for Generic HTTP transfer site includes the following groups of settings:

- *Server settings for Generic HTTP(S) Transfer sites* on page 570
- *Transfer settings for Generic HTTP(S) Transfer sites* on page 571
- *List settings for Generic HTTP(S) Transfer sites* on page 572
- *File download settings for Generic HTTP(S) Transfer sites* on page 574
- *Receive actions for Generic HTTP(S) Transfer sites* on page 577
- *Upload settings for Generic HTTP(S) Transfer sites* on page 578
- *Send actions for Generic HTTP(S) Transfer sites* on page 580
- *Login settings for Generic HTTP(S) Transfer sites* on page 581
- *Advanced settings for Generic HTTP(S) Transfer sites* on page 583

Server settings for Generic HTTP(S) Transfer sites

The following example shows the server settings for a Generic HTTP transfer site.



The following table describes the server settings for defining a Generic HTTP transfer site.

Field	Description
Server Settings	
Specify partner using hostname (IP address) and port number	When selected, the partner will be specified using the partner hostname and port number.
Host	The host name or IP address of the remote host to connect to for file transfers.

Field	Description
Port	The port on the remote host to be used for file transfers.
Specify partner using URL	When selected, the partner will be specified using an URL.
Address	A URL that specifies the partner host. It can also include the port and a directory. Examples: • `http://example.com` • `https://example.com:443` • `https://example.com:443/websealjun/`
Zone	The network zone that defines the proxies to use for transfers through this site. • Select **none** to connect directly to the remote HTTP server. • Select **any** to allow SecureTransport to select the proxy connection using a network zone that enables an HTTP proxy. • Select **Default** to use the default network zone proxy configuration. If no default network zone is defined, transfers from this transfer site fail. • Select a specific network zone to use the proxy configuration defined for that zone. For more information, see *Specify TM Server communication ports and IP address for protocol servers on SecureTransport Edge* on page 236.

Transfer settings for Generic HTTP(S) Transfer sites

The following example shows the transfer settings for a Generic HTTP transfer site.

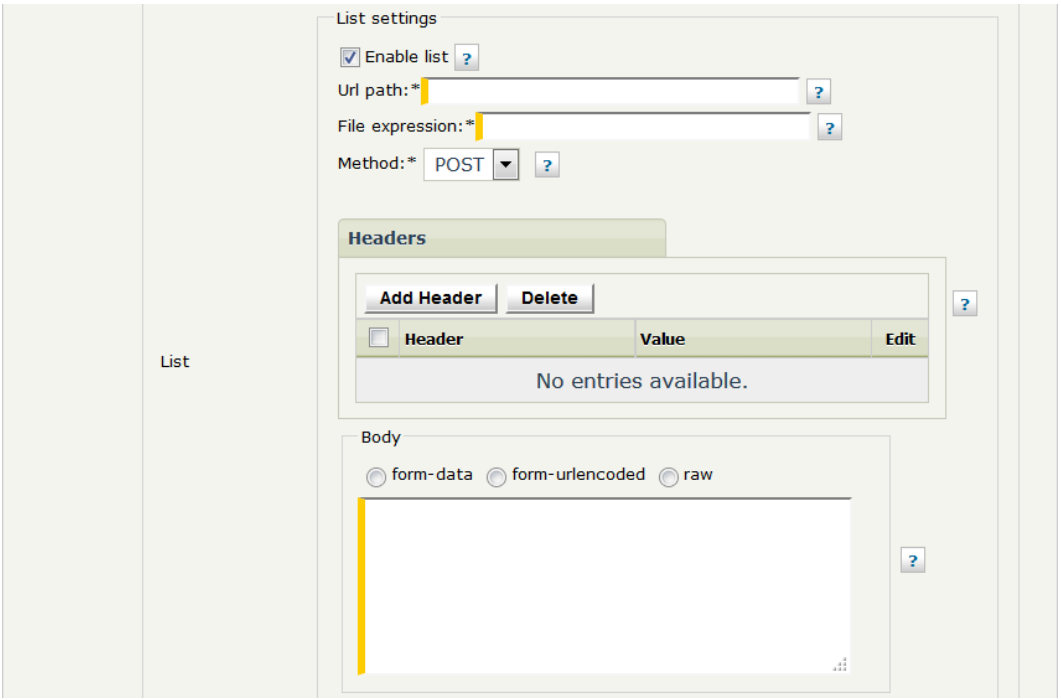
The following table describes the transfer settings for defining a Generic HTTP transfer site.

Field	Description
Transfer Settings	

Field	Description
Use HTTPS	A checkbox indicating if the connection should be secured or not. Deselect this checkbox to use HTTP instead of HTTPS.
Verify certificate for this Site	Select to verify that the remote system is trusted. This field is displayed when **Use HTTPS** is selected.
Enable FIPS Transfer Mode	Restrict HTTPS to use only FIPS 140-2 Level 1 certified cryptographic libraries. This field is displayed when **Use HTTPS** is selected. The sender and the recipient must use the ciphers and ciphers suites listed in FIPS-compliant ciphers and cipher suites (login required). If the sender and the recipient do not provide the required ciphers and ciphers suites SecureTransport does not complete the transfer.

List settings for Generic HTTP(S) Transfer sites

The following example shows the list settings for a Generic HTTP transfer site.



The following table describes the list settings for defining a Generic HTTP transfer site.

Field	Description
List settings	

Field	Description
Enable list	A checkbox indicating if the Generic HTTP transfer site will operate in the single file download mode or will list files and then perform the download operation(s). If checked, the Generic HTTP transfer site will take as an input the result of the configured request and use it as a source for retrieving a list of files to be downloaded. It will then will use the *File download settings* to download each file from the list.

URL path	The HTTP server path that will be used to list files on the remote server:
	• A server absolute path when the partner is specified using a hostname (IP address) and port number.
	Example:
	`/list.php`
	• A server relative path when the partner is specified using an URL.
	Example:
	`list.php`
	The limited expression language can be used to specify the URL.
	Example:
	If the value in **Specify partner using URL** is `https://<host>:<port>/downloads/` and the value for **URL path** is `/list.php`, the final URI that will be resolved is `https://<host>:<port>/list.php`. This occurs because `/list.php` is an absolute path and not a relative path. If the value for **URL path** is `list.php`, the final URI that will be resolved is `https://<host>:<port>/downloads/list.php`.
	This field is displayed when **Enable list** is checked.

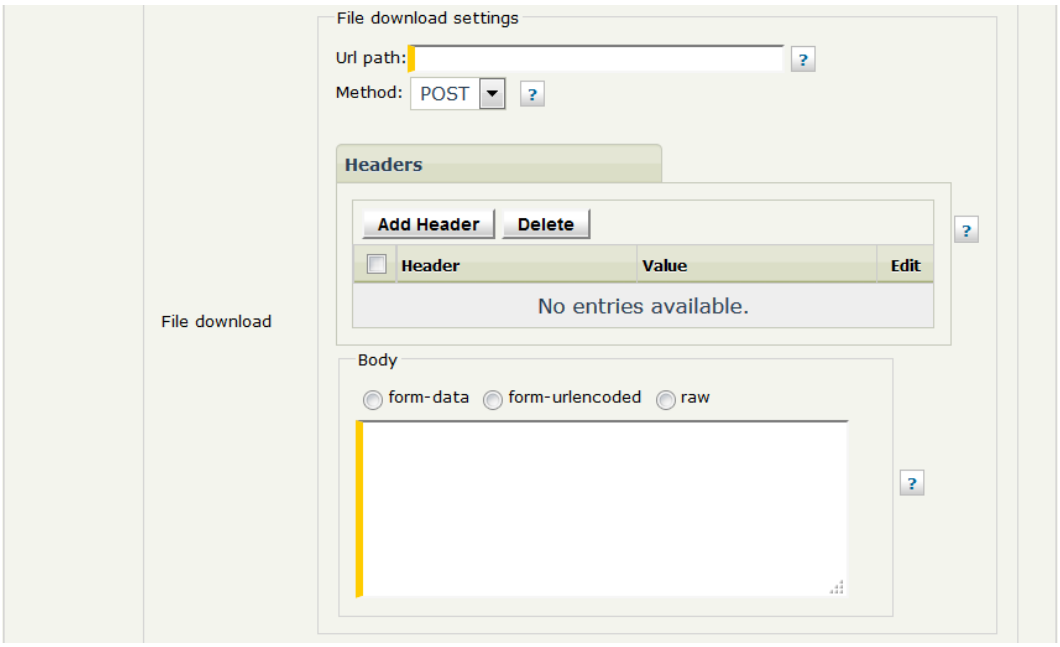
File expression	The expression for the file names that will be applied on the response for the list request to extract the files list. A Java regular expression and the SecureTransport expression language can be used to specify a pattern to match the files that need to be downloaded. It is possible to use () parenthesis in the file expression and everything within the parenthesis will be considered the file name. Example: Filename= `(/something/folder/file\\d.txt)` If they are not used, the matched expression will be the file name. The limited expression language can be used to specify the file name. This field is displayed when **Enable list** is checked.
Method	The HTTP method to be used for listing the files on the remote server. Either **GET** or **POST** can be selected. This field is displayed when **Enable list** is checked.

Field	Description
Headers	The HTTP headers that will be added in the HTTP request for listing files on the remote server. To add a header, click **Add Header** and complete the **Header** and **Value** fields. To edit a header, click the **Edit** ( ) icon and change the **Header** and **Value** fields. To delete a header, select the header to delete and click **Delete**. This Header option is displayed when **Enable list** is checked.

Body	The body of the request for listing files on the remote server. This field is displayed when the selected list Method is **POST**.
	The limited expression language can be used to specify the file name.
	The body can be any of the following types:
	• **form-data** - When selected the body will be transmitted as `multipart/form-data`.
	• **form-urlencoded** - When selected the body will be transmitted as `application/x-www-form-urlencoded`.
	• **raw** – When selected the body can be any text.
	Note When the selected body content type is **form-data** or **form-urlencoded** the body should be formed of key-value pairs on separate lines. Example:
	`param1=value1`
	`param2=value2`
	This field is displayed when **Enable list** is checked and **POST** is the selected Method.

File download settings for Generic HTTP(S) Transfer sites

The following example shows the file download settings for a Generic HTTP transfer site.



The following table describes the file download settings for defining a Generic HTTP transfer site.

Field	Description
File download	
File download settings	

Field	Description

URL path	The HTTP server path used to download file(s):
	• A server absolute path when the partner is specified using a hostname (IP address) and port number.
	Example:
	`/download.php`
	• A server relative path when the partner is specified using an URL.
	Example:
	`download.php`
	If **Enable list** is checked, this URL will be used to download file(s) extracted from the list operation.
	The limited expression language can be used to specify the URL.
	The available environment variables are:
	• `${env['ts_relative_path']}`- The relative path of the file that will be downloaded.
	• `${env['ts_target']}` - The file name of the file that will be downloaded.
	Example:
	If the value in **Specify partner using URL** is `https://<host>:<port>/downloads/` and the value for **URL path** is `/download.php`, the final URI that will be resolved is `https://<host>:<port>/download.php`. This occurs because `/download.php` is an absolute path and not a relative path. If the value for **URL path** is `download.php`, the final URI that will be resolved is `https://<host>:<port>/downloads/download.php`.

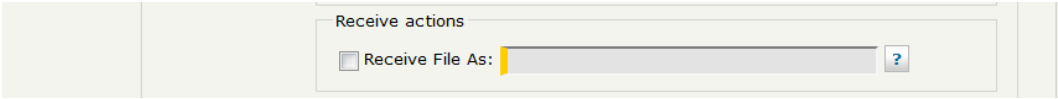
Method	The HTTP method to be used for the file download operation. Either **GET** or **POST** can be selected. If **Enable list** is checked, this method will be used when performing all download operations for the files listed on the remote server.
Headers	The HTTP headers that will be added in the HTTP download request. If **Enable list** is checked, these headers will be added in all the HTTP download requests for the files listed on the remote server. To add a header, click **Add Header** and complete the **Header** and **Value** fields. To edit a header, click the **Edit** () icon and change the **Header** and **Value** fields. To delete a header, select the header to delete and click **Delete**.

Field	Description

Body	The body of the download request. This field is displayed when the selected download Method is **POST**. If **Enable list** is checked, the body will be added in all the HTTP download requests.
	The limited expression language can be used to specify the body.
	The body can be any of the following types:
	• **form-data** - When selected the body will be transmitted as `multipart/form-data`.
	• **form-urlencoded** - When selected the body will be transmitted as `application/x-www-form-urlencoded`.
	• **raw** – When selected the body can be any text.
	The available environment variables are:
	• `${env['ts_content-disposition']}` - The value of the content disposition header presented from the remote HTTP server (if available).
	• `${env['ts_relative_path']}` - The relative path of the file that will be downloaded.
	• `${env['ts_target']}` - The file name of the file that will be downloaded.
	Note When the selected body content type is **form-data** or **form-urlencoded** the body should be formed of key-value pairs on separate lines. Example:
	`param1=value1`
	`param2=value2`

Receive actions for Generic HTTP(S) Transfer sites

The following example shows the receive actions for a Generic HTTP transfer site.



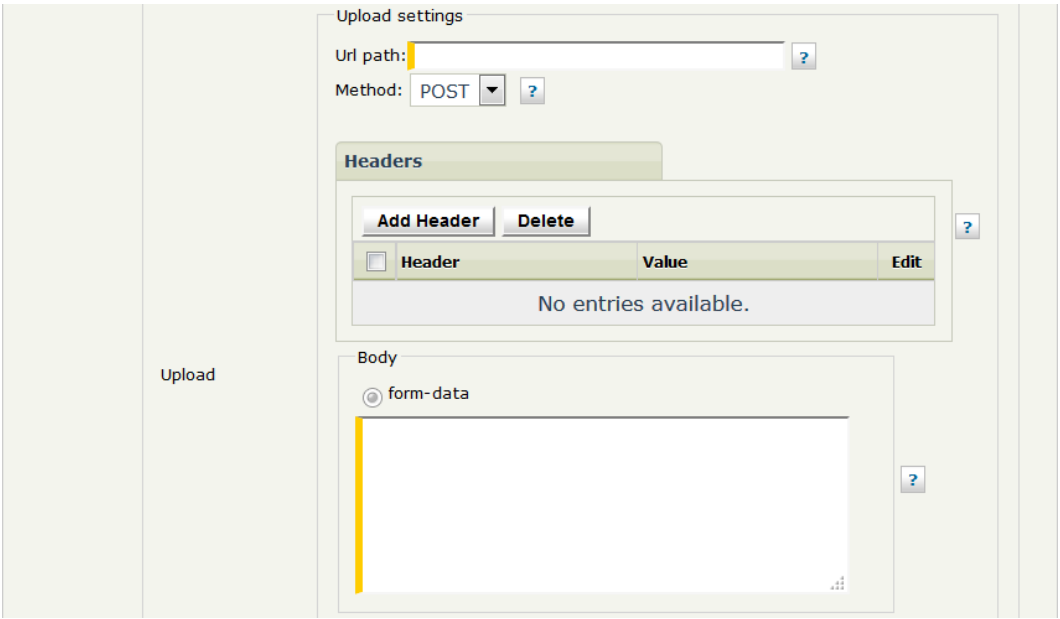
The following table describes the receive actions for defining a Generic HTTP transfer site.

Field	Description
Receive actions	

Field	Description
Receive File As	Select the checkbox to enable renaming the received file and to specify a value to be used to rename the file. The limited expression language can be used to specify the file name. Examples: • `${env['ts_content-disposition']}` • `${env['ts_relative_path']}` • `${env['ts_target']}` • `${env['ts_target']}_${random()}` • `${date('yyyyddMMHHmmss')}` The available environment variables are: • `${env['ts_content-disposition']}` - The value of the content disposition header presented from the remote HTTP server (if available). • `${env['ts_relative_path']}` - The relative path of the file that will be downloaded. • `${env['ts_target']}` - The file name of the file that will be downloaded

Upload settings for Generic HTTP(S) Transfer sites

The following example shows the upload settings for a Generic HTTP transfer site.



The following table describes the upload settings for defining a Generic HTTP transfer site.

Field	Description
Upload	
Upload settings	

URL path	The HTTP server path to be used for uploading files to the remote server:
	• A server absolute path when the partner is specified using a hostname (IP address) and port number.
	Example:
	`/upload.php`
	• A server relative path when the partner is specified using an URL.
	Example:
	`upload.php`
	The limited expression language can be used to specify the URL.
	The available environment variable is:
	• `${env['ts_target']}` - The file name of the file that will be uploaded.
	Example:
	If the value in **Specify partner using URL** is `https://<host>:<port>/uploads/` and the value for **URL path** is `/upload.php`, the final URI that will be resolved is `https://<host>:<port>/upload.php`. This occurs because `/upload.phpt` is an absolute path and not a relative path. If the value for **URL path** is `upload.php`, the final URI that will be resolved is `https://<host>:<port>/uploads/upload.php`.

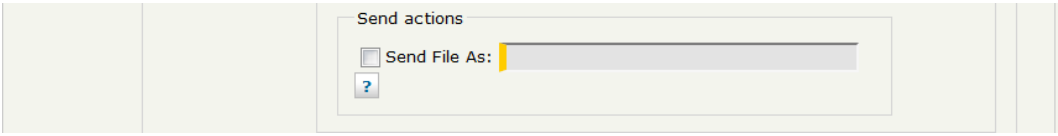
Method	The HTTP method used when uploading files to a remote server. Either **PUT** or **POST** can be selected. This field is displayed when **Enable list** is checked.
Headers	The HTTP headers that will be added in the HTTP request for upload. To add a header, click **Add Header** and complete the **Header** and **Value** fields. To edit a header, click the **Edit** () icon and change the **Header** and **Value** fields. To delete a header, select the header to delete and click **Delete**.

Field	Description

Body	The body of the request for uploading files on the remote server. This option is displayed when the selected upload Method is **POST**.
	The limited expression language can be used to specify the body.
	The available environment variables are:
	• `${env['ts_target']}` - The file name of the file that will be uploaded.
	• `${env['ts_file_form_parameter_name']}` - The name of the form input element with file type on the remote server.
	The body can be of the following type:
	• **form-data** – When selected the body will be transmitted as a `multipart/form-data`.
	Note The body should be formed as key-value pairs on separate lines. Example:
	`param1=value1`
	`param2=value2`
	Example of how to use the environment variables in the body:
	`filename=${env['ts_target']}`
	`${env['ts_file_form_parameter_name']}=myDoc`

Send actions for Generic HTTP(S) Transfer sites

The following example shows the send actions for a Generic HTTP transfer site.

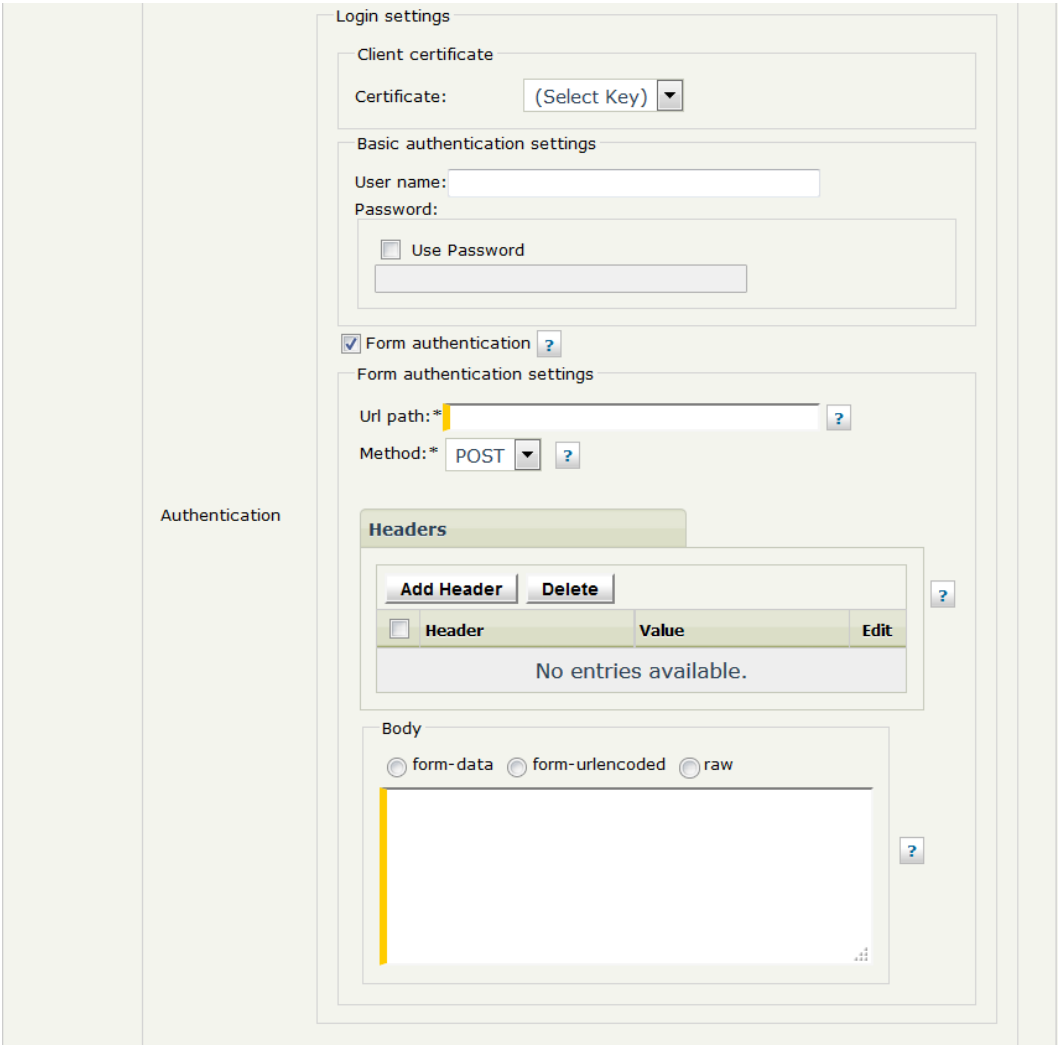


The following table describes the send actions for defining a Generic HTTP transfer site.

Field	Description
Send actions	
Send File As	Select the checkbox to send the file with a different name and specify the file name. The limited expression language can be used to specify the file name. The available environment variable is: • `${env['ts_target']}` - The file name of the file that will be sent. Examples: • `${env['ts_target']}` • `${env['ts_target']}_${random()}` • `${date('yyyyddMMHHmmss')}`

Login settings for Generic HTTP(S) Transfer sites

The following example shows the login settings for a Generic HTTP transfer site.



The following table describes the login settings for defining a Generic HTTP transfer site.

Field	Description
Authentication	
Login Settings	
Client Certificate	
Certificate	The client certificate to be used for mutual authentication.
Basic authentication settings	

Field	Description
User name	The user name to be used to log into a remote HTTP server.
Use Password	Select to use a password to log into the remote HTTP server. Disabled when **Form authentication** is selected.
Password	Password used to log into the remote HTTP server. Disabled when **Form authentication** is selected.
Form authentication	Select to use form authentication. If **Form authentication** is enabled, the transfer site will use form authentication to connect to the remote HTTP server. If username and password in the *Basic authentication settings* pane are set, they will be mapped to the environment variables `${env['ts_form_auth_username']}` and `${env['ts_form_auth_password']}` and can be used in the body of the form authentication request.
Form authentication settings	

URL path	The HTTP server path used for sending the form authentication request:
	• A server absolute path when the partner is specified using a hostname (IP address) and port number.
	Example:
	`/form.php`
	• A server relative path when the partner is specified using an URL.
	Example:
	`form.php`
	The limited expression language can be used to specify the URL.
	Example:
	If the value in **Specify partner using URL** is `https://<host>:<port>/auth/` and the value for **URL path** is `/form.php`, the final URI that will be resolved is `https://<host>:<port>/form.php`. This occurs because `/form.php` is an absolute path and not a relative path. If the value for **URL path** is `form.php`, the final URI that will be resolved is `https://<host>:<port>/auth/form.php`.
	This field is displayed when **Form authentication** is checked.

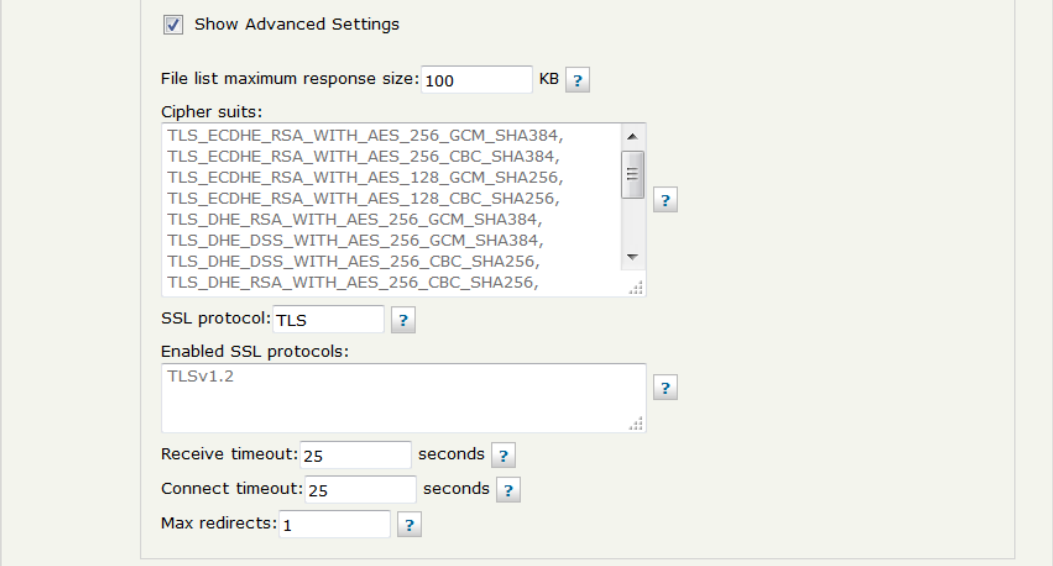
Method	The HTTP method to be used for the form authentication to the remote server. Either **GET** or **POST** can be selected. This field is displayed when **Form authentication** is checked.

Field	Description
Headers	The HTTP headers that will be added in the HTTP request for form authentication. To add a header, click **Add Header** and complete the **Header** and **Value** fields. To edit a header, click the **Edit** ( ) icon and change the **Header** and **Value** fields. To delete a header, select the header to delete and click **Delete**. Displayed when **Form authentication** is checked.

Body	The body of the form authentication request. This field is displayed when the selected authentication Method is **POST**.
	The limited expression language can be used to specify the body.
	The available environment variables are:
	`${env['ts_form_auth_username']}` - Represents the user name specified in the *Basic authentication settings* pane.
	`${env['ts_form_auth_password']}` - Represents the password specified in the *Basic authentication settings* pane.
	The body can be any of the following types:
	• **form-data** – When selected the body will be transmitted as `multipart/form-data`.
	• **form-urlencoded** – When selected the body will be transmitted as `application/x-www-form-urlencoded`.
	• **raw** – When selected the body can be any text.
	Note When the selected body content type is **form-data** or **form-urlencoded** the body should be formed as key-value pairs on separate lines.
	Example:
	`param1=value1`
	`param2=value2`
	Example of how to use the environment variables in the body:
	`user=${env['ts_form_auth_username']}`
	`password=${env['ts_form_auth_password']}`
	This field is displayed when **Form authentication** is checked.

Advanced settings for Generic HTTP(S) Transfer sites

The following example shows the advanced settings for a Generic HTTP transfer site.



The following table describes the advanced settings for defining a Generic HTTP transfer site. To access them, select **Show Advanced Settings**.

Field	Description
Advanced settings	
File list maximum response size	The maximum size in KB of the file list response to handle. If the response exceeds this value, only the specified number of bytes will be processed. Example: If the response is 120 KB and the maximum size is set up to 100 KB, only the first 100 KB from the response will be processed. The default value is `100` KB.
Cipher suites	The cipher suites to be used for the SSL connection. By default, this field is prefilled with the cipher suites as defined in the `Https.SIT.Ciphers` configuration option. To set a custom list of cipher strings for the transfer site, click in the **Cipher suites** field and edit as needed. To revert to the default list, click the **Reload** button.
SSL protocol	The SSL protocol to be used for the SSL connection. The default value is `TLS`.
Enabled SSL protocols	The enabled SSL protocols. By default, the field is prefilled with the SSL protocols as defined in the `Https.SIT.EnabledProtocols` configuration option. To set a custom list of cipher strings for the transfer site, click in the **Enabled SSL protocols**field and edit as needed. To reset to default values, click the **Reload** button.

Field	Description
Receive timeout	The socket timeout in seconds. Any non-zero time out will block the input stream associated with the socket for this amount of time. A timeout of zero is interpreted as an infinite timeout. The default value is `25` seconds.
Connect timeout	The connection timeout in seconds. A timeout of zero is interpreted as an infinite timeout. The connection will then block until established or an error occurs. The default value is `25` seconds.
Max redirects	The maximum number of redirects to be followed. The limit on the number of redirects is intended to prevent infinite loops. The default value is `1` redirect.

For more information about the available security settings, see *Secure your transfer site with SSL/TLS on page 631*.

Sample configuration – List files and download

Sample configuration for a Generic HTTP transfer site to list specific files on an Apache HTTP server and download them:

Field	Description
General Settings	
Site Name	`GHTTP_list`
Site Type	Unspecified
Access Level	Private
Transfer Protocol	Generic-HTTP(S)
Server Settings	
Specify partner using hostname (IP address) and port number	Selected
Host	`10.232.15.114`
Port	`443`
Specify partner using URL	Not selected
Address	—

Field	Description
Network Zone	none
Transfer Settings	
Use HTTPS	Selected
Verify certificate for this Site	Not selected
Enable FIPS Transfer Mode	Not selected
List	
List settings	
Enable list	Selected
URL path	`/uploads`
File expression	`<a href=\"(([^?].*?)\">`
Method	GET
Headers	No headers
File download	
File download settings	
URL path	`${env['ts_relative_path']}/${env['ts_target']}`
Method	GET
Headers	No headers
Receive actions	
Receive File As	Not selected
Upload	
Upload settings	
URL path	—
Method	PUT

Field	Description
Headers	No headers
Send actions	
Send File As	Not selected
Authentication	
Login Settings	
Client Certificate	
Certificate	(Select Key) - No key selected
Basic authentication settings	
User name	`acc`
Use Password	
Password	
Form authentication	Not selected
Advanced settings	
Show Advanced Settings	Not selected

Sample configuration – Download file

Configuration for a Generic HTTP transfer site to download a file from an Apache HTTP server:

Field	Selection or entry
General Settings	
Site Name	`GHTTP_download`
Site Type	Unspecified
Access Level	Private
Transfer Protocol	Generic-HTTP(S)

Field	Selection or entry
Server Settings	
Specify partner using hostname (IP address) and port number	Selected
Host	`10.232.14.182`
Port	`443`
Specify partner using URL	Not selected
Address	—
Network Zone	none
Transfer Settings	
Use HTTPS	Selected
Verify certificate for this Site	Not selected
Enable FIPS Transfer Mode	Not selected
List	
List settings	
Enable list	Not selected
File download	
File download settings	
URL path	`/download.txt`
Method	GET
Headers	No headers
Receive actions	
Receive File As	Not selected
Upload	
Upload settings	

Field	Selection or entry
URL path	—
Method	PUT
Headers	No headers
Send actions	
Send File As	Not selected
Authentication	
Login Settings	
Client Certificate	
Certificate	(Select Key) - No key selected
Basic authentication settings	
User name	`acc`
Use Password	Selected
Password	
Form authentication	Not selected
Advanced settings	
Show Advanced Settings	Not selected

Sample configuration – Upload file

Configuration for a Generic HTTP transfer site to upload a file to an Apache HTTP server:

Field	Selection or entry
General Settings	
Site Name	`GHTTP_upload`
Site Type	Unspecified

Field	Selection or entry
Access Level	Private
Transfer Protocol	Generic-HTTP(S)
Server Settings	
Specify partner using hostname (IP address) and port number	Selected
Host	`10.232.14.182`
Port	`443`
Specify partner using URL	Not selected
Address	—
Network Zone	None
Transfer Settings	
Use HTTPS	Selected
Verify certificate for this Site	Not selected
Enable FIPS Transfer Mode	Not selected
List	
List settings	
Enable list	Not selected
File download	
File download settings	
URL path	—
Method	GET
Headers	No headers
Receive actions	
Receive File As	Not selected

Field	Selection or entry
Upload	
Upload settings	
URL path	`/upload.php`
Method	POST
Headers	Header = `header1` Value = `value1`

Body	Select **form-data**.
	`${env['ts_file_form_parameter_name']}=filename`
	`filename=${env['ts_target']}`

Send actions	
Send File As	Not selected
Authentication	
Login Settings	
Client Certificate	
Certificate	(Select Key) - No key selected
Basic authentication settings	
User name	`acc`
Use Password	Selected
Password	
Form authentication	Not selected
Advanced settings	
Show Advanced Settings	Not selected

Sample configuration – Push file to SecureTransport user using Form Authentication

Configuration for a Generic HTTP transfer site to push file to a SecureTransport user using form authentication:

Field	Description
General Settings	
Site Name	`GHTTP_form`
Site Type	Unspecified
Access Level	Private
Transfer Protocol	Generic-HTTP(S)
Server Settings	
Specify partner using hostname (IP address) and port number	Selected
Host	`10.232.15.114`
Port	`443`
Specify partner using URL	Not selected
Address	—
Network Zone	none
Transfer Settings	
Use HTTPS	Selected
Verify certificate for this Site	Not selected
Enable FIPS Transfer Mode	Not selected
List	
List settings	
Enable list	Not selected

Field	Description
File download	
File download settings	
URL path	—
Method	GET
Headers	No headers
Receive actions	
Receive File As	Not selected
Upload	
Upload settings	
URL path	`/api/v1.4/files?transferMode=BINARY`
Method	POST
Headers	Header=`Referer` Value=`123`

Body	Select **form-data**.
	`${env['ts_file_form_parameter_name']}=filename`
	`filename=${env['ts_target']}`

Send actions	
Send File As	Not selected
Authentication	
Login Settings	
Client Certificate	
Certificate	(Select Key) - No key selected
Basic authentication settings	
User name	`user1`

Field	Description
Use Password	Selected
Password	
Form authentication	Selected
Form authentication settings	
URL path	`/template/login`
Method	POST
Headers	Header=`User-Agent` Value=`Mozilla/5.0`

Body	Select **form-urlencoded**.
	`switch=Log In`
	`user=${env['ts_form_auth_username']}`
	`password=${env['ts_form_auth_password']}`

Advanced settings	
Show Advanced Settings	Not selected

Supported expression language

This topic outlines the limited expression language supported by Generic HTTP transfer sites.

Predefined variables

The predefined variable that is supported:

- `${timestamp}`

Predefined functions

The predefined functions that are supported:

- Functions related to a date. For example: `${date("yyyyMMdd")}`
- Functions related to a Random ID. For example: `${random()}`
- Functions related to a String representation. For example: `${concat('str', 'ing')}`

Note Expression variables and functions related to file name and the SecureTransport environment are not supported.

Added expression variables

On download and list, the following environment variables are added:

- `${env['ts_content-disposition']}` – If a remote HTTP server presents a content disposition header, its filename value will be preserved into this variable.
- `${env['ts_relative_path']}` – The relative path of the file that will be downloaded.
- `${env['ts_target']}` – The file name of the file that will be downloaded or uploaded.

On upload, the following environment variables are added:

- `${env['ts_target']}` – The file name of the file that will be downloaded or uploaded.
- `${env['ts_file_form_parameter_name']}` – Represents the name of the form input element with file type on the remote server.

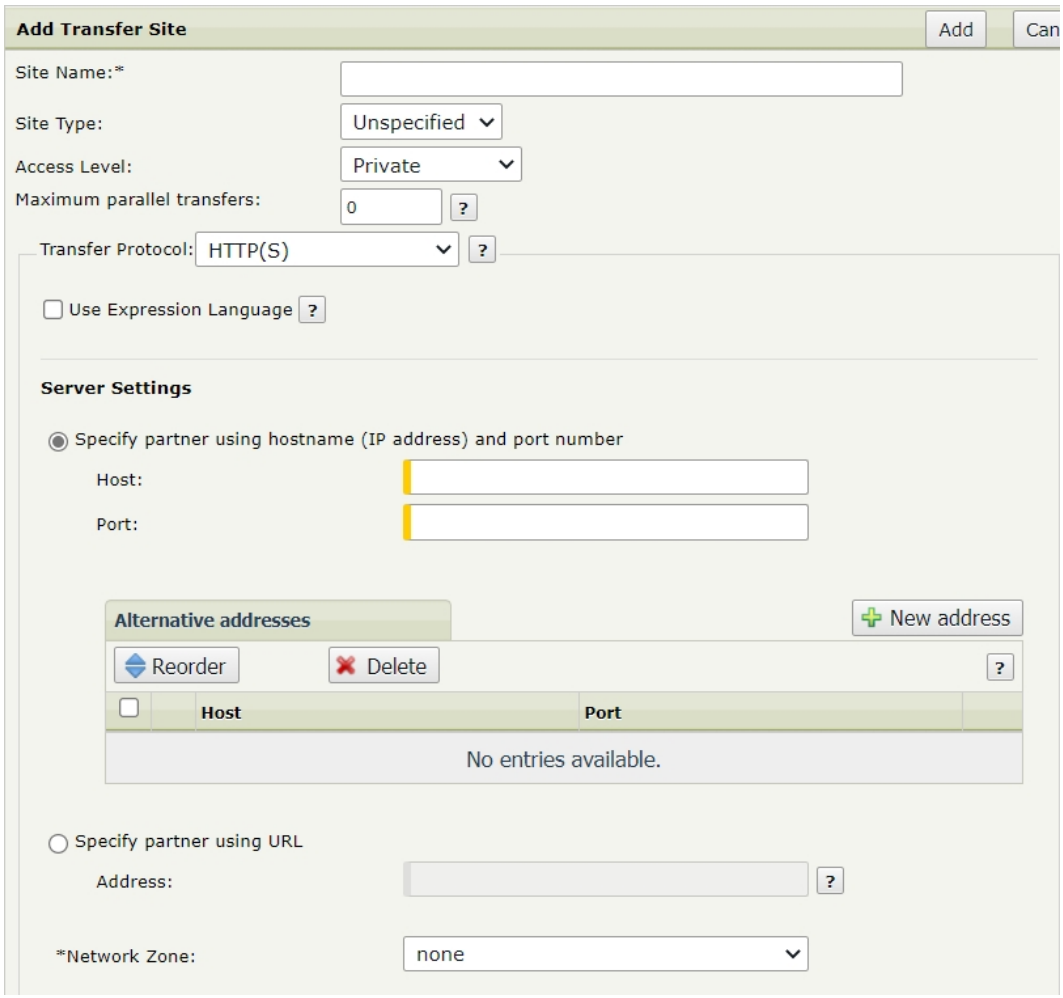
On form authentication, the following environment variables are added:

- `${env['ts_form_auth_username']}` – Represents the user name specified in the *Basic authentication settings* pane.
- `S{env['ts_form_auth_password']}` – Represents the password specified in the *Basic authentication settings* pane.

HTTP(S) transfer sites

SecureTransport Server provides support for guaranteed delivery and restart for transfers using the HTTP protocol when the remote server is a SecureTransport Server.

The *Add Transfer Site* page for HTTP(S) sites presents several sets of options.



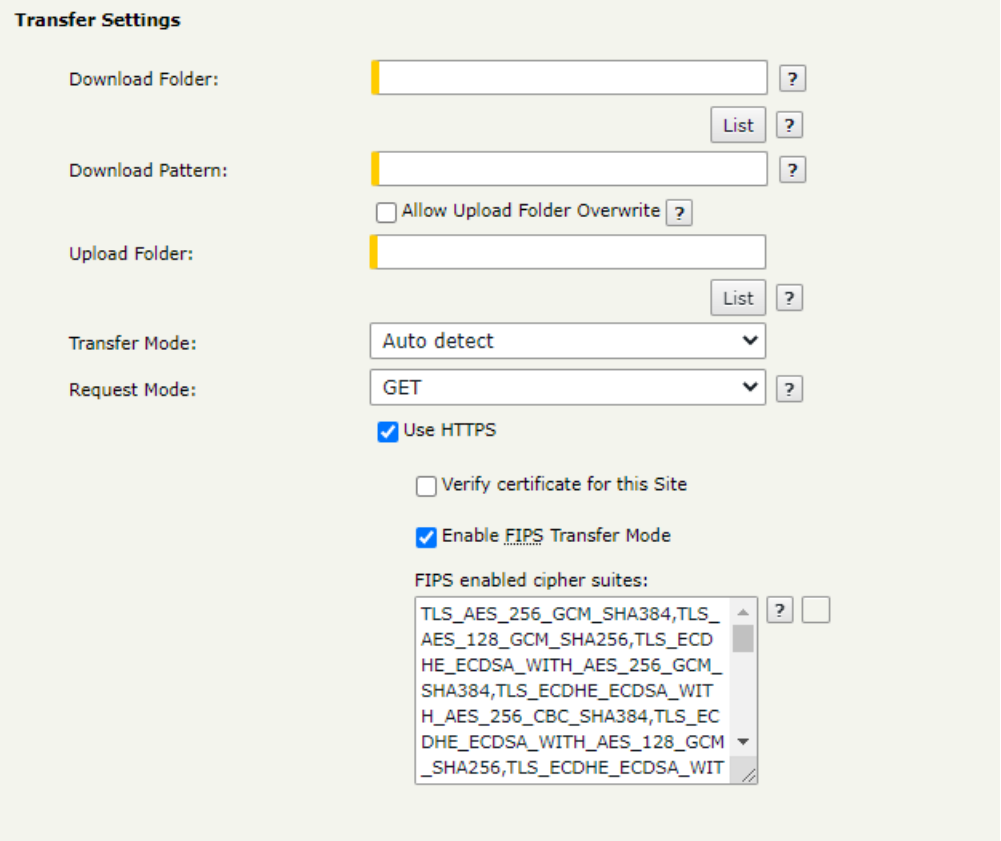
Server settings

The following table describes the general options for a HTTP(S) transfer site.

Field	Description
Server Settings	
Host	Select **Specify partner using hostname (IP address) and port number** to enable this field. Enter either the *host name* or *IP address* of the remote host to connect to for file transfers. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.

Field	Description
Port	Select **Specify partner using hostname (IP address) and port number** to enable this field. Enter the port number on the remote host to be used for file transfers. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Alternative addresses	The visibility of this option is controlled with the value set for the `TransferSite.AlternativeAddresses.retryPolicy` configuration option. It allows you to set a list of endpoints that act as backup alternatives to the configured Server-Port site settings and are particularly useful in cases of transfer failures. For mode details, see *Set Alternative addresses* on page 633.
Address	Select **Specify partner using URL** to enable this field. Note that with this selection, the *Alternative addresses* grid moves under this option on the screen. Enter a URL that specifies the partner host. It can also include the port and a path (directory).
Network Zone	The network zone that defines the proxies to use for transfers through this site. • Select **none** to connect directly to the remote HTTP server. • Select **any** to allow SecureTransport to select the proxy connection using a network zone that enables an HTTP proxy. • Select **Default** to use the default network zone proxy configuration. If no default network zone is defined, transfers from this transfer site fail. • Select a specific network zone to use the proxy configuration defined for that zone. For more information, see *Specify TM Server communication ports and IP address for protocol servers on SecureTransport Edge* on page 236.

Transfer settings



Field	Description
Transfer Settings	
Download Folder	The folder on the remote server from which the file is transferred. You can use the expression language to append dates. For example, if you use the expression `folder_${date("yyyyMMdd")}`, the download folder will be evaluated using the date of the transfer execution. For example `folder_20210130`. To see the list of the folder's files and subfolders, click **List**. For more details, see *List the contents of the Upload or Download folder* on page 630.
Download Pattern	Specify a file name pattern, using wildcard characters or a regular expression, to identify files to be downloaded. For regular expression syntax, see *Regular expressions* on page 1117. For globbing syntax, see *Globbing in SecureTransport* on page 1122. The download pattern is evaluated during the transfer site execution.

Field	Description
Allow Overwrite	Taken into account when the site is used by the Send To Partner step. If checked the value of "Upload folder" will be overwritten with the value of "Overwrite upload folder". For more details see *Advanced Routing* on page 864.
Upload Folder	The folder on the remote server to which files are transferred. To see the list of the folder's files and subfolders, click **List**. For more details, see *List the contents of the Upload or Download folder* on page 630.
Transfer Mode	Specify whether data is transferred as ASCII or binary. You can also choose to have SecureTransport automatically determine the correct transfer mode. For more information about automatically determining transfer mode, see *Transfer mode for server-initiated transfers* on page 764.
Use HTTPS	Deselect this checkbox to use HTTP instead of HTTPS.
Verify certificate for the Site	Select to verify that the remote system is trusted. This field is displayed when **Use HTTPS** is selected.
Enable FIPS Transfer Mode	Restrict HTTPS to use only FIPS 140-2 Level 1 certified cryptographic libraries. This field is displayed when **Use HTTPS** is selected. When you enable FIPS transfer mode, the panel expands with an additional field that lets you specify the desired set of cipher suites to be used in FIPS mode for server-initiated transfers through this site. By default, this set is populated with the cipher suites as defined in the `Https.FIPS.SIT.Ciphers` configuration option. You can add or remove cipher suites. The supported FIPS cipher suites from which you can select when adding a new one are listed in FIPS-compliant ciphers and cipher suites (login required). **Note** Both the sender and the recipient must use supported FIPS ciphers suites. Otherwise, the transfer will fail.

Site login credentials

Field	Description
Site Login Credentials	

Field	Description
User Name	Username used to log in to the HTTP server. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Use Password	Select to use a password to log in to the HTTP server.
Password	Password used to log in to the HTTP server. Switching the toggle provides the ability to use Expression Language for evaluating the password.
Certificate	A private certificate for SecureTransport used to log in to the FTP server. You can select a certificate or import a certificate. This field is displayed when Use HTTPS is selected. When **Use Expression Language** is enabled, you can set the certificate dynamically by choosing the scope (account or server level) and providing a valid expression that will be evaluated to the name of an available certificate. By default, the usage of expired X509 certificates is allowed for SIT transfers. To forbid it, set the `SIT.allowExpiredCertificates` to `false`

Post-transmission actions

Post-transmission actions are file operations, such as rename, move and delete, that are performed after a file transfer ends and can be different based on the transfer status. For more information, see *Set post-transmission actions in transfer sites* on page 634.

Advanced SSL settings

Advanced SSL settings allow you to define the SSL protocol/cipher suites combinations used with the current HTTPS transfer site. Select **Show Advanced SSL Settings** to expand the pane with the available options:

Field	Description
Show Advanced SSL Settings	
Cipher suites	The set of cipher suites available with the current HTTPS transfer site for secure SIT connections. By default, this field is prefilled with the cipher suites as defined in the `Https.SIT.Ciphers` configuration option. To set a custom list of cipher strings for the transfer site, click in the **Cipher suites** field and edit as needed. To revert to the default list, click the **Reload** button.

Field	Description
Enabled SSL protocols	The allowed SSL protocols for secure SIT connection with the current HTTPS transfer site. By default, this option uses the SSL protocols as defined in the `Https.SIT.EnabledProtocols` configuration option. To set a custom list of cipher strings for the transfer site, click in the **Enabled SSL protocols** field and edit as needed. To reset to default values, click the **Reload** button.

For more information about the available security settings, see *Secure your transfer site with SSL/TLS on page 631*.

Note When Single Sign-On (SSO) for end-users is enabled, you can not transfer files over HTTP (S).

Note SecureTransport will not be able perform server initiated file pushes or pulls over HTTP to and from another SecureTransport instance if the second requires SSO authentication for the users as the HTTP transfer site cannot handle the SSO authentication.

PeSIT configuration overview

This section provides an overview of the PeSIT protocol and discusses how it is used in SecureTransport.

What is PeSIT?

PeSIT is a file transfer protocol that allows a machine-to-machine exchange of files. File-writing, also known as file transmission, enables a PeSIT user to transfer the contents of a file to another user of the PeSIT protocol. The party that requests the service is known as the “Caller”, and its correspondent is called the "Server". In this case, the file transfer occurs between the Caller/Sender and the Server/Receiver.

File-reading, or file reception, allows a PeSIT user to request another PeSIT user to transfer data. In this case, the file transfer is between the Caller/Receiver and the Server/Sender.

During a session between two PeSIT users, the dialogue is always asymmetrical: the two parties play different complementary roles. The requester of the session, the Caller, also takes the initiative regarding transfers, while the Server executes the requests received from the Caller.

A given party can have the following roles:

- Caller/Sender
- Caller/Receiver
- Server/Sender
- Server/Receiver

A partner is a SecureTransport object that represents a sender or receiver in a PeSIT transfer. To exchange data over the PeSIT protocol, the two partners must first establish a logical connection, the so-called PeSIT partnership.

Establish a PeSIT partnership

SecureTransport represents an entity of a PeSIT partner by the combination of an account and a transfer site. It can identify a PeSIT partner in one of two ways:

- through the combination of the name of the account user and its PeSIT transfer site. This is the default method, and it requires the aforementioned names to be unique on both sides.
- through the combination of the PeSIT ID parameters of an account and its PeSIT transfer site. The advantage of this method is that it eliminates the need for both parties to use unique names in their configurations.

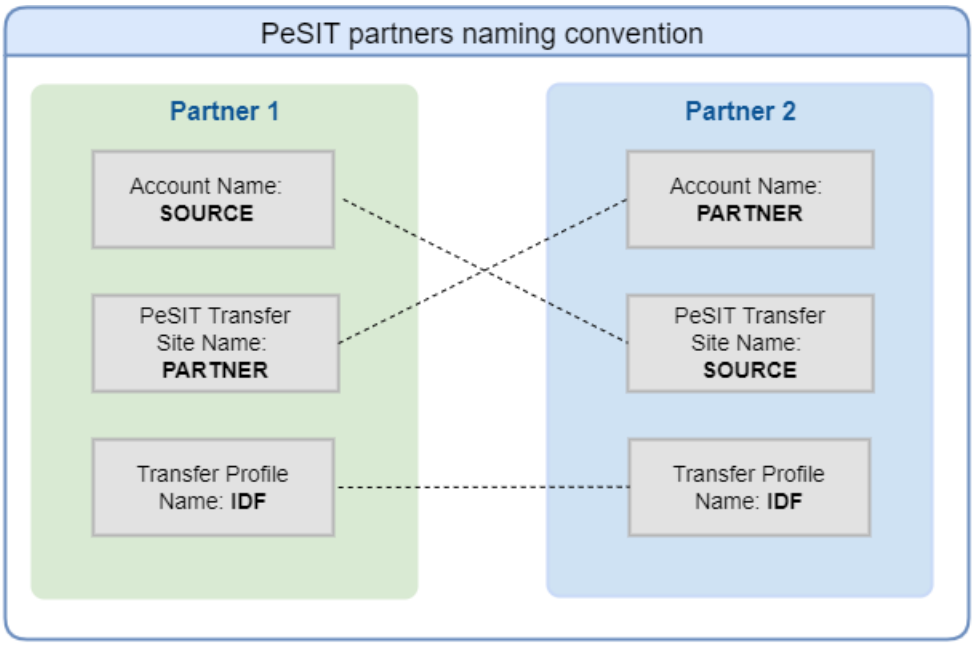
  This functionality becomes available after installing the December 2021 update, 5.5-20211216. To enable it, set the configuration option `Pesit.UsePesitIds` to true. When done, the PeSIT partnership is formed based on the PeSIT ID properties specified in the account and the transfer site settings. PeSIT ID is not a mandatory field, and if it is left empty, SecureTransport defaults to using the name property.

Both methods require the configuration of accounts and transfer sites to be agreed upon between the PeSIT partners because there are some rules for naming identifiers (either the names or PeSIT IDs) and the partnership is defined by using the identifiers of the other party. Note that the selected method also affects the Pre-Connection functionality, routing the acknowledgments, and Store and Forward mode.

In SecureTransport, the configuration of PeSIT partnership involves the following steps:

1. Create an account. See *Create a user account* on page 503.
   - If using name identification, the account name must match the name of the remote partner's transfer site.
   - If using PeSIT IDs, the account name is customizable, and the account's PeSIT ID must match the PeSIT ID of the remote partner's transfer site.
2. Create a PeSIT transfer site for that account. See *PeSIT transfer sites* on page 603.
   - If using name identification, the transfer site's name must match the name of the remote partner's account.
   - If using PeSIT IDs, the transfer site name is customizable, but its PeSIT ID parameter must match the PeSIT ID of the remote partner's account.
3. Configure a transfer profile for the aforementioned transfer site to use when receiving or sending the files. The name of the transfer profile must be the same on both ends. See *Transfer profiles* on page 640.

The following example illustrates the logical connection between two PeSIT partners through name identification:

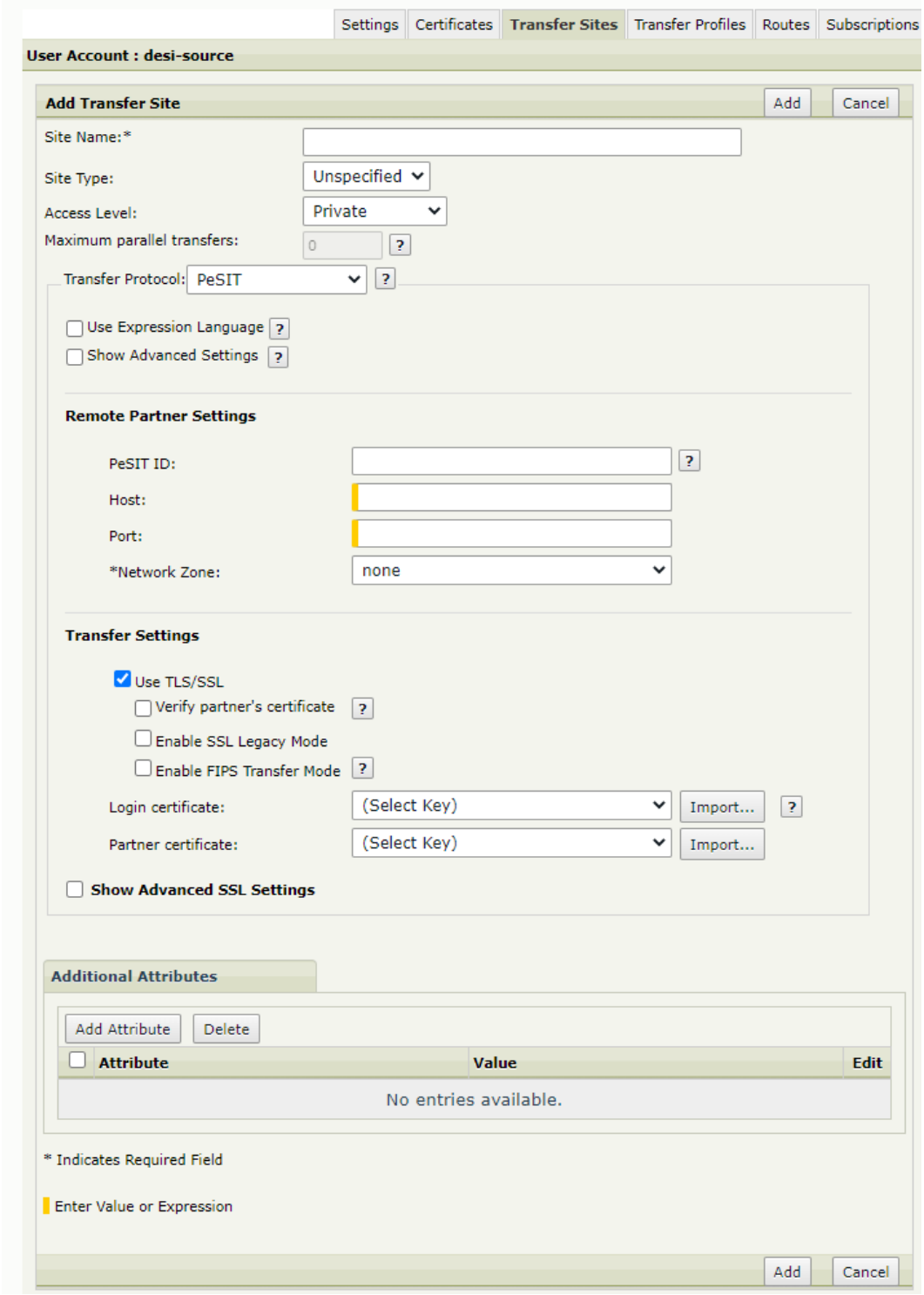


PeSIT transfer sites

Unlike transfer sites for other transfer protocols, a PeSIT transfer site is also used for transfers initiated by the external PeSIT partner (considered client-initiated by SecureTransport). **Site Name** is always required, but you can use also use the **PeSIT ID** to define a PeSIT partnership, see *PeSIT configuration overview* on page 601.

For a PeSIT transfer site, the **Site Name** or site's **PeSIT ID** (depending on what is specified in the `Pesit.UsePesitIds`) designates the destination for an incoming routed transfer. For more information, see *PeSIT configuration overview* on page 601.

The *Add Transfer Site* page for PeSIT sites presents several sets of options.



General site settings

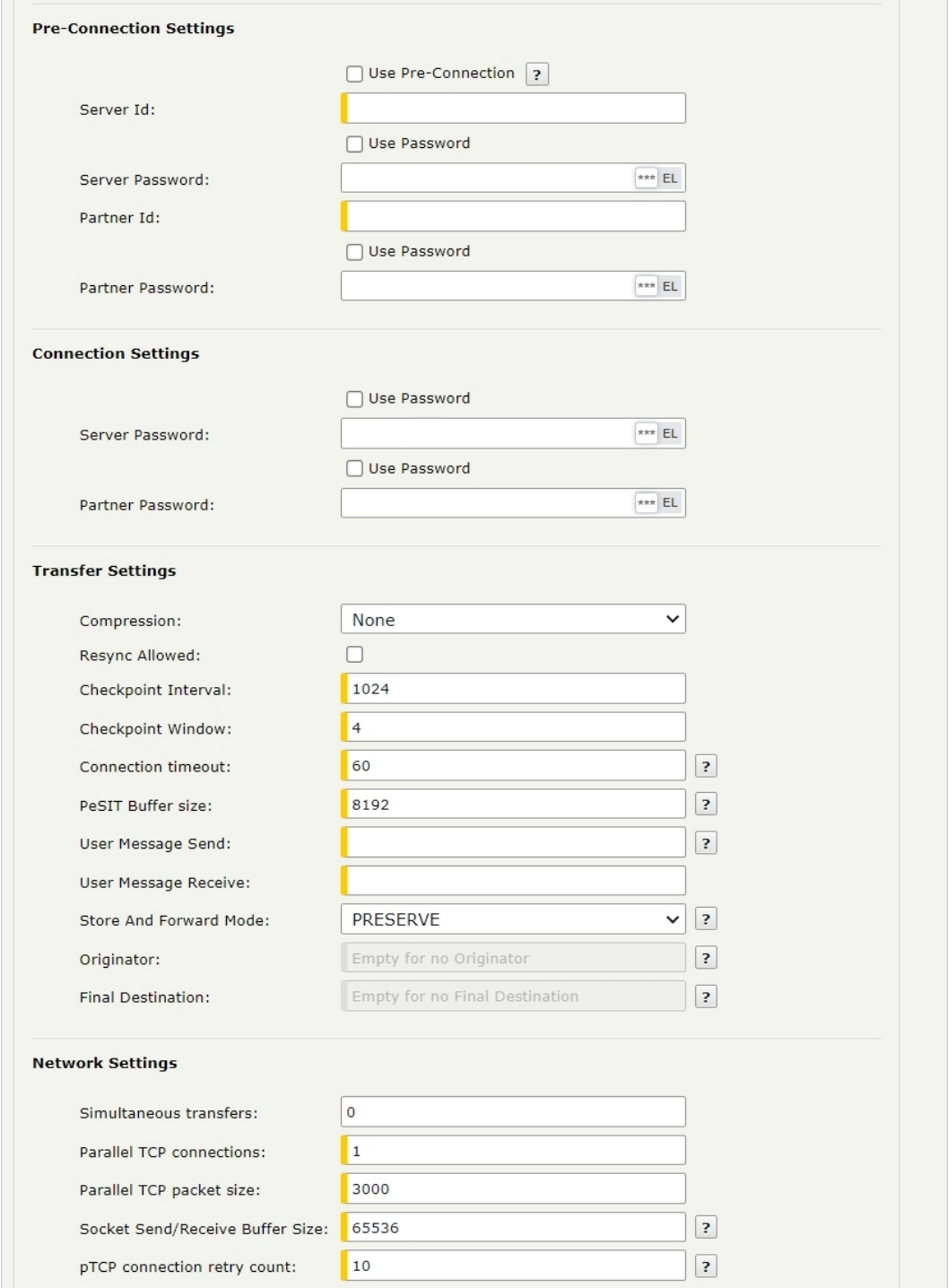
The following table describes the general options for a PeSIT transfer site.

Field	Description
Remote Partner Settings This group of options is displayed right below your selection of **Transfer Protocol**: PeSIT.	
PeSIT ID	When the `PeSIT.UsePesitIds` is set to `true`, this value will be used for PeSIT Server or Partner ID (based on the type of transfers) during transfers and acknowledgments. Otherwise, **Site Name** will be used. See *PeSIT configuration overview* on page 601.
Host	The host name or IP address of the remote server to connect to for file transfers. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Port	The port on the remote server to be used for file transfers. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Alternative addresses	The visibility of this option is controlled with the value set for the `TransferSite.AlternativeAddresses.retryPolicy` configuration option. It allows you to set a list of endpoints that act as backup alternatives to the configured Server-Port site settings and are particularly useful in cases of transfer failures. For mode details, see *Set Alternative addresses* on page 633.
Network Zone	The network zone that defines the proxies to use for transfers through this site. • Select **none** to connect directly to the remote partner server. • Select **any** to allow SecureTransport to select the proxy connection using a network zone that enables an SOCKS5 proxy.SecureTransport • Select **Default** to use the default network zone proxy configuration. If no default network zone is defined, transfers from this transfer site fail. • Select a specific network zone to use the proxy configuration defined for that zone. For more information, see *Specify TM Server communication ports and IP address for protocol servers on SecureTransport Edge* on page 236.
Transfer Settings Please note that the options described appear on display when the **Show Advanced Settings** option is **not** selected. When you select the **Show Advanced Settings** checkbox, these options are moved under **Network Settings**.	
Use TLS/SSL	Requires the use of TLS or SSL for communication with the partner server. When selected, additional fields appear on display.

Field	Description
Verify partner's certificate	Verify the TLS/SSL certificate of the partner site. This field is displayed when the **Use TLS/SSL** option is selected. When selected, SecureTransport verifies whether the server certificate of the partner is chained to a trusted root using the algorithm specified in `AgentServers.Ssl.trustAlgorithm` server configuration parameter and the certificates imported in the Trusted CAs store.
Enable SSL Legacy Mode	Requires the use of SSL Legacy mode for communication with the partner server. This field is displayed when the **Use TLS/SSL** option is selected.
Enable FIPS Transfer	Restrict PeSIT to use only FIPS 140-2 Level 1 certified cryptographic libraries. This field is displayed when the **Use TLS/SSL** option is selected. When you enable FIPS transfer mode, the panel expands with an additional field that lets you specify the desired set of cipher suites to be used in FIPS mode for server-initiated transfers through this site. By default, this set is populated with the cipher suites as defined in the `Pesit.FIPS.SIT.Ciphers` configuration option. You can add or remove cipher suites. The supported FIPS cipher suites from which you can select when adding a new one are listed in FIPS-compliant ciphers and cipher suites (login required). **Note** Both the sender and the recipient must use supported FIPS ciphers suites. Otherwise, the transfer will fail.
Login certificate	The local certificate to use when connecting to the partner site. By default, the usage of expired X509 certificates is allowed for SIT transfers. To forbid it, set the `SIT.allowExpiredCertificates` to `false`.
Partner certificate	The login certificate to use when authenticating the remote site.

Advanced Settings

Scroll down to the bottom of the screen and click the **Show Advanced Settings** to expand the screen with additional options.



Note The Pre-Connection Settings fields have length validation. It is **not** applied by the User Interface (frontend validation) when the value is **Expression Language**. In such situations, the administrator has the responsibility of providing a valid expression and value.

Field	Description
Pre-connection settings	Select **Show Advanced Settings** for the following group of options to appear right below. PeSIT pre-connection settings allow you to map a *server ID* and *password* (Server Settings) to the corresponding client-side *partner ID* and *password* (Partner Settings). PeSIT pre-connection acts as a mechanism for additional verification prior to establishing a PeSIT connection. By default, the **Use Pre-Connection** checkbox is not selected. If you leave it this way, the following rules apply, depending on the SecureTransport role in Pre-Connection phase: • SecureTransport as *Server* – SecureTransport does not validate the received Partner ID and Partner Password. • SecureTransport as *Client* – SecureTransport sends to the target PeSIT Server the Account name as a Partner ID and the Connection Partner Password (if specified) as Partner Password. When you select the **Use Pre-Connection** checkbox you must add either Server or Partner Settings, or both. In all cases, the Id field is required and the Password field is optional. With the input of Server or Partner Settings, the following rules apply, depending on the SecureTransport role in the Pre-Connection phase: • SecureTransport as *Server* – SecureTransport validates the received Partner ID and Partner Password against the configured Server ID and Server Password. • SecureTransport as *Client* – SecureTransport sends the configured Partner ID and Partner Password to the target PeSIT Server.
Server Id	The ID against which the Server validates the received Partner ID during the Pre-Connection phase. It can contain any symbols (up to 8 characters). Leading or trailing spaces will be trimmed.
Use Password	Specifies whether a password should be used during the Pre-Connection Phase. When editing a Transfer Site, the passwords are not visible and you can use this checkbox to either keep using the password in the Pre-Connection phase, or remove the password from the Database.
Password *(optional)*	The password against which the Server validates the received Partner password during the Pre-Connection phase. It can contain any symbols (up to 8 characters). White spaces are not allowed. Leading or trailing spaces will be trimmed.
Partner Settings	These options appear when you select **Pre-Connection settings**.
Partner Id	The ID that the PeSIT client sends during the Pre-Connection phase. It can contain any symbols (up to 8 characters). Leading or trailing spaces will be trimmed.

Field	Description
Password *(optional)*	The password that the PeSIT client sends during the Pre-Connection phase. It can contain any symbols (up to 8 characters). White spaces are not allowed. Leading or trailing spaces will be trimmed.
Use Password	Specifies whether a password should be used during the Pre-Connection Phase. When editing a Transfer Site, the passwords are not visible and you can use this checkbox to either keep using the password in the Pre-Connection phase, or remove the password from the Database.
Connection Settings	
Server Password	Select **Use Password** and type in a password that will be required when this Server connects to a remote partner. Valid passwords are string values consisting of one to eight characters. Expression Language is supported.
Partner Password	Select **Use Password** and type in a password that will be required when a remote partner connects to this Server and password authentication is used. Valid passwords are string values consisting of one to eight characters. Expression Language is supported.
Transfer Settings	These options appear when you select **Show Advanced Settings**. Please note that when you have not selected the **Show Advanced Settings** checkbox, a different group of options is presented, as described here.
Compression	Enables horizontal online compression, vertical online compression, or both for transfers initiated by the SecureTransport Server. If the partner PeSIT server does not support the selected compression, no compression is used for these transfers. SecureTransport support all types of compression for transfers initiated by the partner PeSIT server.
Resync Allowed	Enables dynamics resynchronization of exchanges during transfer, without interrupting the data exchange phase.
Checkpoint Interval	The maximum number of bytes in KB (equals 1024 bytes) that the sender may transmit between two consecutive checkpoints. Checkpoints are used to restart the transfer when required. A value of zero indicates no checkpoints. A value of 65535 indicates an undefined interval.
Checkpoint Window	The greatest difference allowed between the number of the last checkpoint transmitted and the number of the last checkpoint acknowledged. When this number of checkpoints are not acknowledged, the sender suspends data transmission until it receives a checkpoint acknowledgment. A value of zero indicates that no acknowledgments are required.

Field	Description
Connection Timeout	When SecureTransport acts as a client, the value of this field specifies the amount of time (in seconds) that SecureTransport will wait for an acknowledgment for a transfer. Default value: the value specified in the `Pesit.Client.Inactivity.Timeout` configuration option. Accepted values: positive integers. If specified, the Connection Timeout value overwrites the `Pesit.Client.Inactivity.Timeout` value.
PeSIT Buffer size	The size of the internal buffer for this transfer site in bytes. Valid values are 512 to 65535. A larger buffer improves performance. Specifies the maximum size of a PeSIT data element (PI 25). Should be greater than 800 bytes and less than 65535.
User Message Send	A string sent as PI 99 when the SecureTransport Server initiates a file transfer **to** the partner PeSIT server. • The field may contain expressions. The tooltip lists valid expressions. If SecureTransport receives the file over PeSIT, it retains the values of all the PeSIT PI codes as metadata and the PeSIT expression language variables contain those values. See also *Expression Language* on page 1104, especially *PeSIT expressions* on page 1110. The string that results from the evaluation of the expression must be at most 512 characters long. • If SecureTransport is a relay in a PeSIT flow between two Transfer CFTs, leave this field empty to retain and pass PI 99 in the CFT-specific format from the original sender to the final recipient. See *CFT PeSIT extensions* on page 318.
User Message Receive	A string included in messages sent when the SecureTransport Server initiates a file transfer **from** the partner PeSIT server. The field may contain expressions. The string that results from the evaluation of the expression must be at most 512 characters long.
Store and Forward Mode	Select the Store and Forward mode: **START_NEW** or **PRESERVE**. **Note** The Store and Forward mode selected here can be overwritten from the *Send To Partner* on page 948 step settings.

Field	Description
Originator	• In case of SecureTransport initiating a new Store and Forward transfer, this property specifies the originator (PI 61) of the transfer. **Note** The originator specified in the PeSIT transfer site can be overwritten from the Advanced Routing Send To Partner step setting Originator. • In case no value is specified in both this filed and the Advanced Routing Send To Partner step setting **Originator**, PI61 is blank. • When the PRESERVE store and forward mode is selected, this field is disabled as PI preserves the PI61 value.
Final Destination	• In case of SecureTransport initiating a new Store and Forward transfer, this property specifies the final destination (PI 62) of the transfer. **Note** The final destination specified in the PeSIT transfer site can be overwritten from the Advanced Routing Send To Partner step setting Final Destination. • In case no value is specified in both this field and the Advanced Routing Send To Partner step setting **Final Destination**, PI62 is blank. • When the PRESERVE store and forward mode is selected, this field is disabled as PI preserves the PI62 value.
Network Settings	These options appear when you select **Show Advanced Settings**. **Note** The following settings: **Use TLS/SSL** onward are described here.
Simultaneous transfers	The maximum number of simultaneous transfers from this transfer site to remote PeSIT systems. A value of zero means no limit.
Parallel TCP connections	The number of TCP connections to make for parallel TCP (pTCP) to accelerate transfers.
Parallel TCP packet size	The pTCP packet size in bytes.
Socket Send / Receive Buffer Size	The size of the pTCP buffers in bytes. Specifies the TCP Socket maximum send and receive buffer size in bytes. This setting corresponds to SO_SNDBUF and SO_RCVBUF socket parameters.

Field	Description
pTCP connection retry count	The number of attempts SecureTransport makes for each TCP connection for pTCP. When the value of the **Host** field is the address of load balancer for a remote PeSIT cluster, set this field to *connections* * (*nodes* - 1), where: • *connections* is the value of the **Parallel TCP Connections** field • *nodes* is the number of nodes in the remote PeSIT cluster SecureTransport retries the connections until all connections are with the same PeSIT remote server. It specifies the maximum times the SecureTransport will attempt to re-establish a connection with the remote server in case of "Unknown session" error. This is useful in cases where the remote partner is a PeSIT cluster, the address in the transfer site represents the load balancer in front of the PeSIT cluster and the individual nodes behind the Load Balancer are not accessible. In such an environment, all connections have to arrive on the same partner node. Depending on the load balancing configuration different number of retries or no retries (sticky session LB configuration) might be required.

Note The options displayed below **pTCP connection retry count** are described here.

Advanced SSL Settings

Advanced SSL settings allow you to define a custom list of allowed cipher suites and SSL protocols for the current PeSIT transfer site. Select **Show Advanced SSL Settings** to expand the pane with available options.

Field	Description
Show Advanced SSL Settings	
Cipher suites	The set of cipher suites available with the current PeSIT transfer site for secure SIT connections. By default, this field is prefilled with the cipher suites as defined in the `Pesit.SIT.Ciphers` configuration option. To set a custom list of cipher strings for the transfer site, click in the **Cipher suites** field and edit as needed. To revert to the default list, click the **Reload** button.
Enabled SSL protocols	The available SSL protocols for secure SIT connection with the current PeSIT transfer sites. By default, By default, this option uses the SSL protocols as defined in the `Pesit.SIT.EnabledProtocols` configuration option. To set a custom list of cipher strings for the transfer site, click in the **Enabled SSL protocols** field and edit as needed. To reset to default values, click the **Reload** button.

For more information about the available security settings, see *Secure your transfer site with SSL/TLS* on page 631.

Use a default PeSIT transfer site for routing

SecureTransport implements PeSIT routing as an intermediate partner by sending a received file to a PeSIT transfer site specified as the destination of the PeSIT transfer. It matches the specified destination either against the names or the PeSIT IDs of the transfer sites for the account that receives the file.

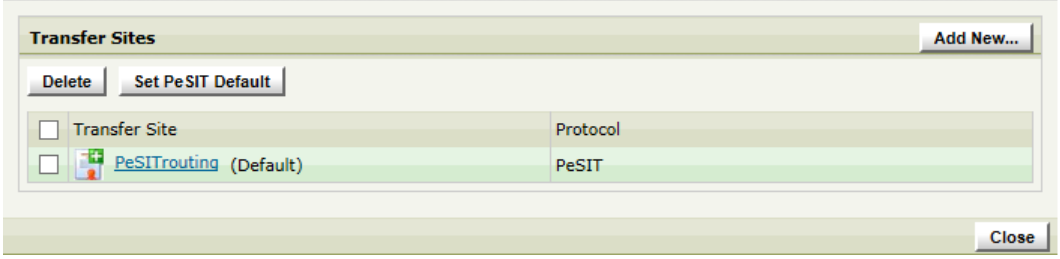
- If there's a match, SecureTransport transfers the file to that site. No subscription is required.
- If there is no match and a default PeSIT transfer site is defined, SecureTransport transfers the file to the default site. If there is no default site, SecureTransport checks the **PeSIT Routing Mode** value for the account. If it is **Reject** (default), the transfer is rejected before it starts. If it is **Accept**, the transfer is performed and the file is retained locally. If it is **Ignore**, SecureTransport ignores the final destination and submit the file to an Advanced Routing application for processing. That is why **Ignore** must be only selected for accounts with Advanced Routing configured.

When SecureTransport routes a transferred file to a final PeSIT destination, it includes PI61 and PI62.

Follow the instructions to set the default PeSIT transfer site for an account:

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click the name of the account for which you want to set the default PeSIT transfer site.
3. Click the **Transfer Site** tab.
4. Select the checkbox next to the name of the PeSIT transfer site you want to make the default.
5. Click **Set PeSIT Default**.

   The default is indicated in the transfer site list.

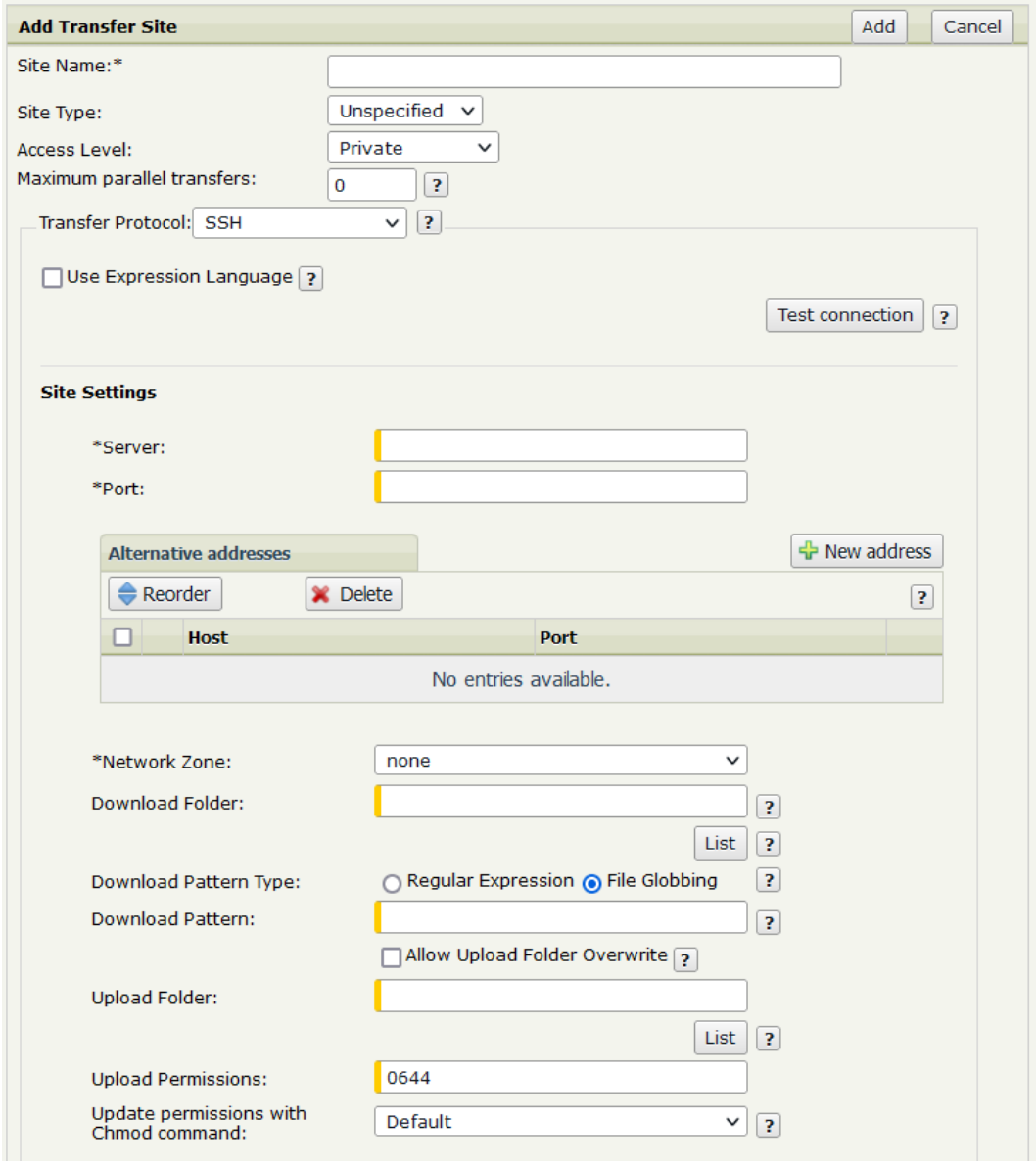


SSH transfer sites

The SSH transfer site uses the SSH protocol to connect and transfer files across remote servers. SecureTransport accounts can be subscribed to any of the built-in applications and the SSH site can be configured to push to or pull from a destination, or both.

Site settings

The following image presents a snippet of the *Site Settings* pane for an SSH protocol transfer site:



The following table describes the site settings options for an SSH protocol transfer site.

Field	Description
Server	The host name or IP address of the remote server to connect to for file transfers. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Port	The port on the remote server to be used for file transfers. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.

Field	Description
Alternative addresses	The visibility of this option is controlled with the value set for the `TransferSite.AlternativeAddresses.retryPolicy` configuration option. It allows you to set a list of endpoints that act as backup alternatives to the configured Server-Port site settings and are particularly useful in cases of transfer failures. For mode details, see *Set Alternative addresses* on page 633.
Network Zone	The network zone that defines the proxies to use for transfers through this site. • Select **none** to connect directly to the remote SSH server. • Select **any** to allow SecureTransport to select the proxy connection using a network zone that enables an SOCKS5 proxy. • Select **Default** to use the default network zone proxy configuration. If no default network zone is defined, transfers from this transfer site fail. • Select a specific network zone to use the proxy configuration defined for that zone. For more information, see *Specify TM Server communication ports and IP address for protocol servers on SecureTransport Edge* on page 236.
Download Folder	The folder on the remote server from which the file is transferred. To list the files and subdirectories in it, click **List**. For more details, see *List the contents of the Upload or Download folder* on page 630. To use the expression language to append dates: The download folder will be evaluated using the current date when the transfer site is being executed. For example `folder_20210130`. Example:`folder_${date("yyyyMMdd")}`
Download Pattern Type	Specify whether you want to use glob pattern matching or regular expression syntax to match the strings specified for the "Download Pattern" field.
Download Pattern	Specify a file name pattern to identify the files to be downloaded. For regular expression syntax, see *Regular expressions* on page 1117. For globbing syntax, see *Globbing in SecureTransport* on page 1122. The download pattern is evaluated when the transfer site is being executed. Examples: • The glob `*_${date("yyyyMMdd")}.txt` will be evaluated using the current date of the transfer site execution, e.g., on the 30.10.2022, it will match all files that end with `_20210130.txt`. • The regex `*[a-z]_${date("yyyyMMdd")}.txt` will also be evaluated using the current date, e.g., on the 30.10.2022, it will match all files that start with any combination of letters from `a` to `z` and end with `_20210130.txt`.

Field	Description
Allow Overwrite	Taken into account when the site is used by the Send To Partner step. If checked, the value of "Upload Folder" will be overwritten with the value of "Overwrite upload folder". For more details, see *Advanced Routing* on page 864.
Upload Folder	The folder on the remote server to which files are transferred. To list the files and subdirectories in it, click **List**. For more details, see *List the contents of the Upload or Download folder* on page 630.
Upload Permissions	Sets permissions of the remote file during SFTP push. SecureTransport changes the file permissions as they are set in the SSH Transfer Site during the transfer.
Update permissions with `Chmod` command	Determines how SecureTransport changes the permissions of a file: using the `chmod` or the `umask` command. This setting overrides the global setting specified via the `Ssh.UpdateFilePermissionsWithChmodCommand` option. The possible values are: • `Default` – The global setting is applied. • `True` – The file permissions, specified in the SSH transfer site, are set after the transfer ends with `chmod`. • `False` – The file handler is opened with specified permissions. The file permissions, specified in the SSH transfer site, are set with `umask`. **Note** On Windows environments, if SecureTransport is expected to receive files with names that contain Windows reserved characters, this setting must be set to `false`.

Test Connection

The Test Connection feature is available from the Administration Tool and the REST API 2.0 under *POST /sites/operations*. It lets you check if the connection between SecureTransport Server and the remote partner is configured correctly before you save the transfer site. The test is performed based on what is currently specified in the transfer site definition.

The table below lists all transfer site parameters that are considered in the connection test:

Mandatory fields for Test Connection:	Server, Port, Network Zone, Password, SSH Key, User Name
Optional fields and settings for Test Connection:	• all Transfer settings • all Network settings • all Advanced SSH settings

Mandatory fields must not contain expression language. Using variables in a mandatory field will cause a test connection failure.

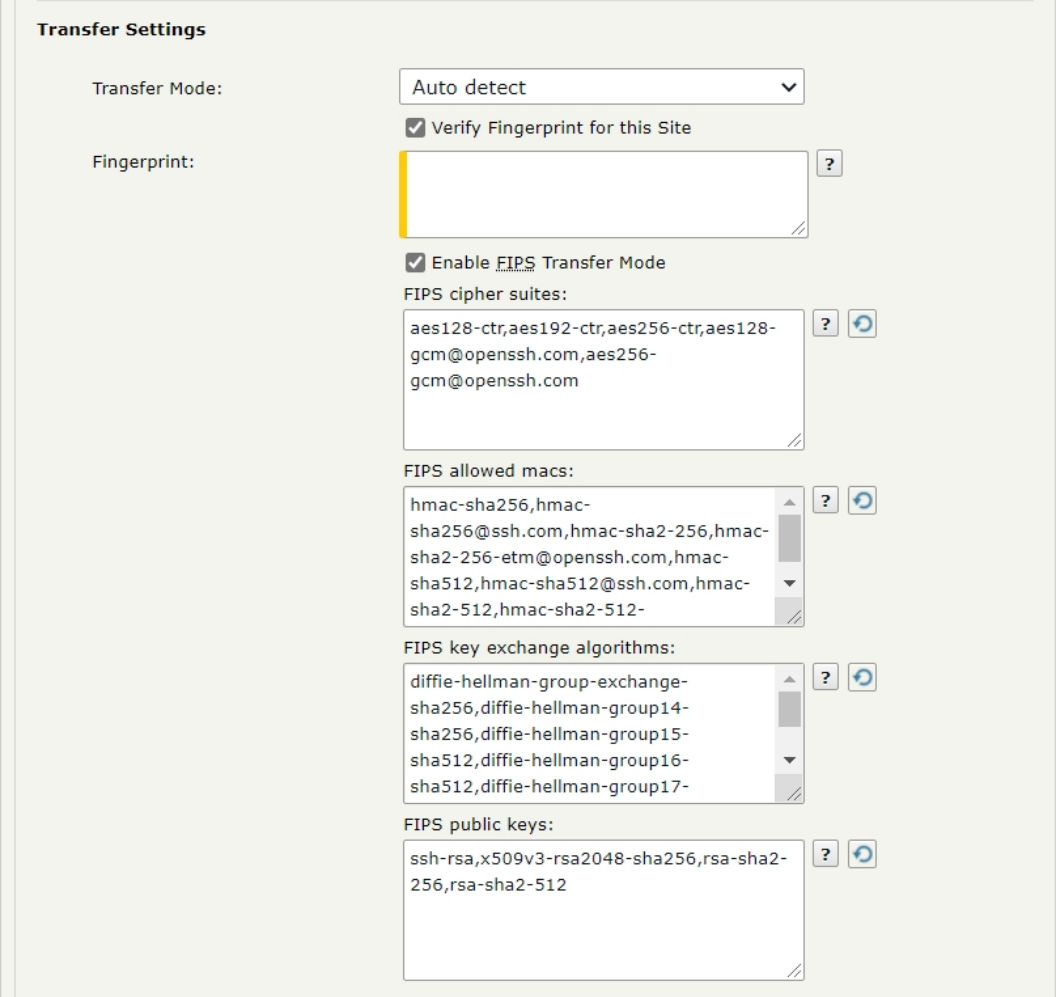
Variables in the optional fields are ignored and do not affect the test. If an optional field or setting is not specified, the test connection is made with its default value.

To initiate a test connection, click the **Test Connection** button. The *Result test connection* box appears, providing status and test results:

- `Connection status` – `success` or `failed`.
- `Fingerprint verification status` – `success`, `failed` or `not verified`.
    - `success` – the fingerprint verification during the test connection is successful.
    - `failed` – the fingerprint verification during the test connection failed.
    - `not verified` – the fingerprint verification is skipped during the test connection.
- `Fingerprint` – the fingerprint used in the test connection.
- `Cipher suite` – the name of the cipher suite used in the test connection.
- `HMAC` – hash-based message authentication codes used in the test connection.
- `Key exchange algorithms` – the KEX used in the test connection.
- `Public key` – the public key used in the test connection.
- `Send Buffer size` – the size of the send buffer in bytes `(SO_SNDBUF)` used in the test connection.
- `Receive Buffer size` – the size of the receive buffer in byte `(SO_RCVBUF)` used in the test connection.
- `Authentication status` – either `success` or `failed`.
- `SSH key alias` – the SSH key alias used in the test connection.
- `Session ID` – the Session ID associated with the test connection, represented as a link to the filtered Server Log entries.
- `Error details` – in the event of an error, displays detailed information on why the test connection failed.

Transfer settings

Transfer Settings pane for an SSH protocol transfer site:



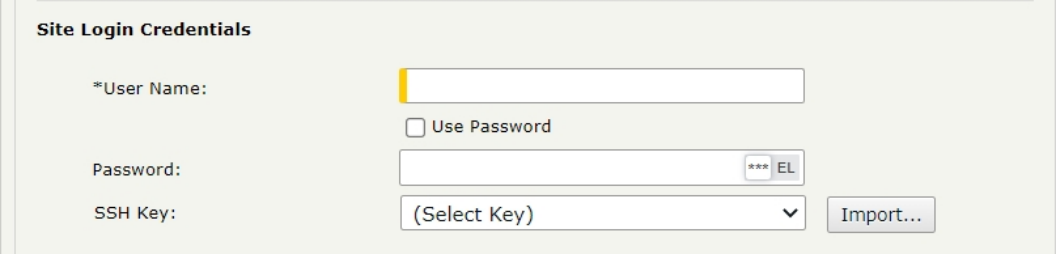
The following table describes the transfer settings options for an SSH protocol transfer site.

Field	Description
Transfer Settings	
Transfer Mode	Specify whether data is transferred as ASCII or binary. You can also choose to have SecureTransport automatically determine the correct transfer mode. For more information about automatically determining transfer mode, see *Client-initiated and server-initiated transfers* on page 761.
Verify Fingerprint for this Site	Select this checkbox to require SecureTransport to verify the fingerprint for the SSH key against the value you specify below. If the values do not match, the connection is refused.

Field	Description
Fingerprint	The value against which you want to verify the fingerprint from the remote server. If the partner SSH server has both DSA and RSA keys configured, the fingerprint that SecureTransport must verify for a server-initiated transfer depends on FIPS transfer mode. With FIPS transfer mode enabled, enter the fingerprint for the DSA key. With FIPS transfer mode disabled, enter the fingerprint for the RSA key. **Note** The fingerprint value must start with a formatted hashing algorithm name in the following format: `<hashing_algorithm>:<certificate_ssh_fingerprint_hash>` Examples: • `MD5:e5:07:3a:1c:08:bd:c5:bd:65:47:a2:4e:3c:b7:5f:27` • `SHA256:0WHO0AyTy1jixmk0zj6zxK36DeBqj2aNhgMFQEIZnvw=`
Enable FIPS Transfer Mode	Restrict SSH to use only FIPS 140-2 Level 1 certified cryptographic libraries. When you enable FIPS transfer mode, the panel expands with the following fields that let you specify the desired set of SSH ciphers and algorithms for server-initiated transfers through this site: • FIPS cipher suites – allowed ciphers for server-initiated transfers through this site in FIPS mode. By default, this set is populated with the cipher suites as defined in the `Ssh.FIPS.SIT.Ciphers` configuration option. • FIPS allowed macs – allowed MAC algorithms for server-initiated transfers through this site in FIPS mode. By default, this set is populated with the MAC algorithms as defined in the `Ssh.FIPS.SIT.AllowedMacs` configuration option. • FIPS key exchange algorithms – allowed KEX algorithms for server-initiated transfers through this site in FIPS mode. By default, this set is populated with the KEX algorithms as defined in the `Ssh.FIPS.SIT.KeyExchangeAlgorithms` configuration option. • FIPS public keys – allowed public keys for server-initiated transfers through this site in FIPS mode. By default, this set is populated with the public keys as defined in the `Ssh.FIPS.SIT.PublicKeys` configuration option. All fields are editable. The supported FIPS ciphers and algorithms from which you can select when adding new ones are listed in FIPS-compliant ciphers and cipher suites (login required). **Note** Both the sender and the recipient must use FIPS-compliant ciphers and algorithms supported by SecureTransport. Otherwise, the transfer will fail.

Site login credentials

Site Login Credentials pane for an SSH protocol transfer site:

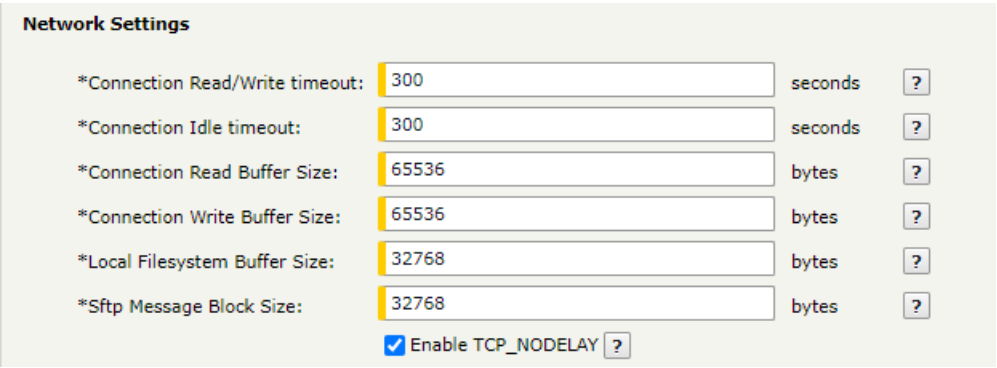


The following table describes the site login credentials options for an SSH protocol transfer site.

Field	Description
Site Login Credentials	
User Name	Username used to log in to the SSH server. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
Use Password	Select to use a password to log in to the SSH server.
Password	Password used to log in to the SSH server. Switching the toggle provides the ability to use Expression Language for evaluating the password.
SSH Key	The certificate used to identify the user logging in. You can select a certificate or import a certificate. By default, the usage of SSH keys contained in expired X509 certificates is allowed for SIT transfers. To forbid it, set the `SSH.SIT.allowExpiredCertificates` to `false`.

Network settings

Network Settings pane for an SSH protocol transfer site:



The following table describes the network settings options for an SSH protocol transfer site.

Field	Description
Network Settings	
Connection Read/Write timeout	The maximum number of seconds the server waits to read a block of data from the partner server, or write a block of data to the partner server. If not specified, its value is 300 seconds. This option corresponds to the `SO_RVCTIMEO` and `SO_SNDTIMO` Socket options.
Connection Idle timeout	The maximum length of time, in seconds, that a connection can stay active when no traffic is sent through it. The default is 300 seconds.
Connection Read Buffer Size	The size of the receive buffer in bytes used by the socket open for the transfer. It is used by the platform's networking code as a hint for the size to set the underlying network I/O buffers. Increasing the receive buffer size can increase the performance of network I/O for high-volume connections, while decreasing it can help reduce the backlog of incoming data. This value is also used to set the TCP receive window that is advertised to the remote peer. This option corresponds to the `SO_RCVBUF`. The value should be a positive integer.
Connection Write Buffer Size	The size of the send buffer in bytes used by the socket open for the transfer. It is used by the platform's networking code as a hint for the size to set the underlying network I/O buffers. This option corresponds to the `SO_SNDBUF`. The value should be a positive integer.
Local Filesystem Buffer Size	The size of the buffer in bytes used for reading from the local file system when performing the transfer.
SFTP Message Block Size	The SFTP block size value used for the transfer.
Enable TCP_NODELAY	Enable or disable Nagle algorithm for the transfer.

SSH connection reuse

SecureTransport Update 5.5-20230126 introduces connection pooling for server-initiated transfers, which allows SecureTransport to reuse recently established connections to the remote SSH server. A separate connection pool is created for each SSH transfer site. If set, the **Maximum parallel transfers** property defines the maximum number of connections that can be opened by the connection pool per node.

This feature can be enabled and modified via the following global server configuration options:

Server configuration option	Description	Default value
`Ssh.SIT.ConnectionPool.Enabled`	Enable or disable connection pools for SSH transfer sites.	`false`
`Ssh.SIT.ConnectionPool.MinEvictableIdleDuration`	The minimum amount of time, in seconds, that SecureTransport waits before closing an idle connection.	`30`
`Ssh.SIT.ConnectionPool.TimeBetweenEvictionRuns`	The minimum amount of time, in seconds, that SecureTransport waits in between checking for idle connections.	`15`

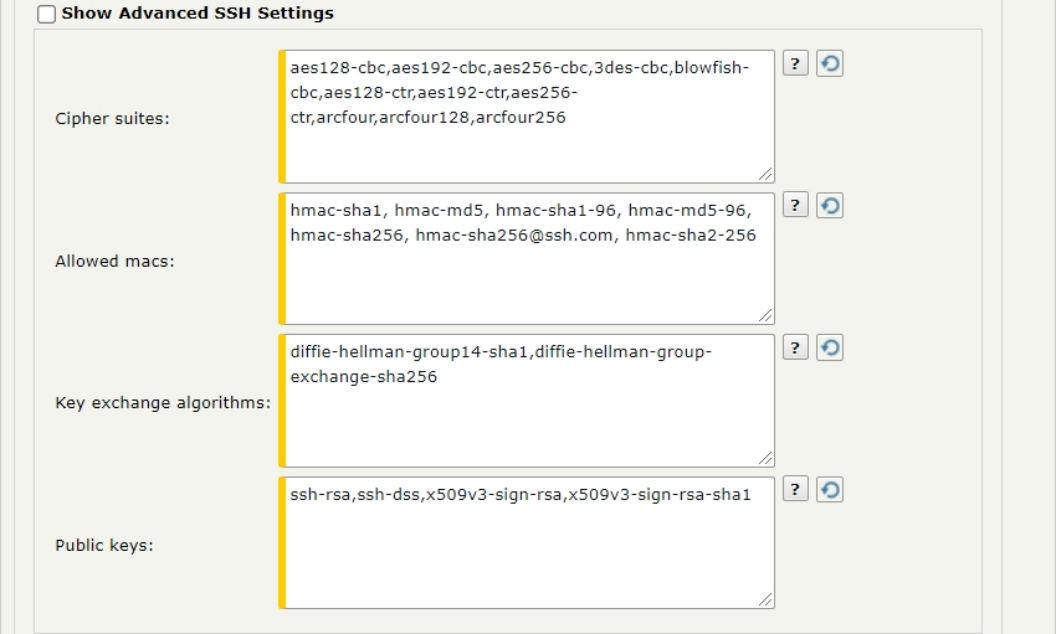
Note Changing any of the values requires restarting the TM service.

Post-transmission actions

Post-transmission actions are file operations - such as rename, move and delete- that are performed after a file transfer ends and can be different based on the transfer status. For more information, see *Set post-transmission actions in transfer sites* on page 634.

Advanced SSH Settings

Scroll down to the bottom of the screen and select **Show Advanced SSH Settings** to expand the pane with additional options.



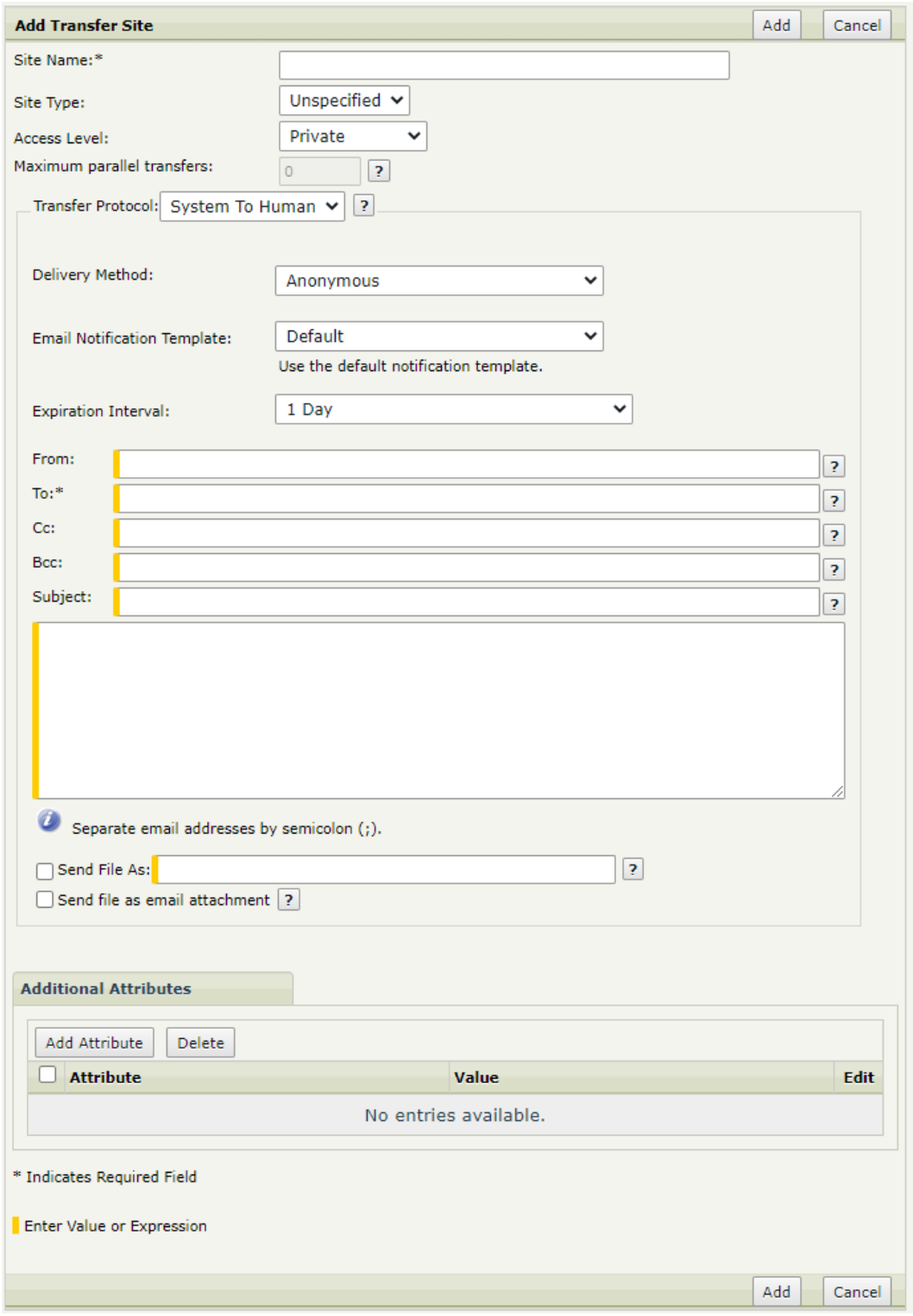
The following table describes the Advanced SSH Settings for an SSH protocol transfer site.

Field	Description
Show Advanced SSH Settings - select this check-box to expand the pane with available options.	
Cipher suites	The set of cipher suites for secure SIT connection with the current SSH transfer site. By default this set is populated with the cipher suites as defined in the `Ssh.SIT.Ciphers` configuration option. To reset to default values, click the button next to the tooltip.
Allowed macs	The set of allowed HMAC algorithms with the current SSH transfer site for secure SIT connection, presented in a comma-separated list. By default this list is populated with the supported MAC algorithms as defined in the `Ssh.SIT.AllowedMacs` configuration option.
Key exchange algorithms	The set of allowed key exchange algorithms with the current SSH transfer site for secure SIT connection, presented in a comma-separated list. By default this list is populated with the supported key exchange algorithms as defined in the `Ssh.SIT.KeyExchangeAlgorithms` configuration option.
Public keys	The set of allowed public key algorithms with the current SSH transfer site for secure SIT connection, presented in a comma-separated list. By default this list is populated with the supported public exchange algorithms as defined in the `Ssh.SIT.PublicKeys` configuration option.

System to Human transfer sites

You can use a System to Human (S2H) transfer site to send files to email recipients. The S2H transfer site sends a notification email to each of the email addresses that you entered containing one or more download file links and optionally the actual file, a custom subject line and message. The receiver retrieves the file based on the chosen delivery method and access level. In some cases, the receiver may need to log in to ST Web Client and answer a question to retrieve the file.

The following is the *Add Transfer Site* page for a transfer site definition that uses the System to Human transfer protocol.



You can use hardcoded values, expressions in the supported expression language, or a combination of both to complete the fields indicated by a vertical yellow bar. For more information about expressions, see *Expression Language* on page 1104.

The following table describes the System to Human options for a transfer site.

Field	Description
Delivery Method	Controls user enrollment: • **Anonymous** – The recipient clicks a link in the email and can retrieve the files. The prefix for the URLs displayed in the notification emails are set in the network zone's *Public URL Prefix* field. • **Challenge** – When the recipient clicks the link in the email to retrieve the files, the recipient must answer a secret question correctly. • **Existing Account** – The recipient must have a SecureTransport account. • **Enroll Unlicensed** – SecureTransport enrolls the recipient as an unlicensed user, if necessary. • **Enroll Licensed** – SecureTransport enrolls the recipients as licensed users, if necessary.
Secret Question	This field is displayed when the **Delivery Method** is `Challenge`. Type the question that the email recipient must answer to retrieve the files.
Answer Re-enter Answer	These fields are displayed when the **Delivery Method** is `Challenge`. Type the answer to the question.
Email Notification Template	Select `Default` or an email template that SecureTransport uses to compose the file transfer notification and status emails. You specify the email templates on the *Mail Template Repository* page. You specify the default email notification template on the *AdHoc Settings* page.
Expiration Interval	The number of days before the message expires. The choices are: 1 Day, 7 Days, 30 Days, 60 Days, and Never.

Field	Description
From	Sets the sender information that appears in the "From" field of the outgoing emails, also known as display name. It could be either a single email address or an RFC 822 phrase-style address. Phrase-style addresses are in the form *phrase <email-address>*, where • The phrase portion is optional and can contain ASCII characters, spaces and predefined variables. For example `ST Administrator ${stenv.loginname}` is a valid syntax. • If there is a phrase portion, the email address must be enclosed in angle brackets; for example, `Sender ${stenv.loginname} <john.doe@mail.com>`. The way a display name appears depends on the recipient's email program. Some email clients show the full name with the email address, others might only display the email address part or the first 20 characters of the text portion. Also, the SMTP servers can place restrictions based on the email address entered in the "From" field. If you do not configure the **From** field, SecureTransport will use the email address specified in the **Email Contact** field of the user account's settings. If it is also blank, SecureTransport will not be able to send the email and will log an error.
To	Defines the main recipients. You can list multiple email addresses separated by a semicolon (`;`) or use a regular expression.
Cc	Defines the cc recipients. You can list multiple email addresses separated by a semicolon (`;`) or use a regular expression.
Bcc	Defines the bcc recepients. You can list multiple email addresses separated by a semicolon (`;`) or use a regular expression.
Subject	Sets the subject of the file transfer emails. Expression language is supported; for example, you can use *${date('yyyyddMM')}* to have the date automatically added to the subject line.
Text	In the unlabeled email body field, enter the text of the notification emails. The field supports expression language.
Send File As	Select the checkbox to specify a file name. You can enter a regular expression for SecureTransport to use to construct a new file name based on the original file name. For example, when a file arrives in a S2H file transfer, SecureTransport renames the file by prepending a unique ID to the file name. If the file it routed to an S2H transfer site to forward it, the expression, `${stenv.target.replace('^.{66}', '')}`, removes the ID.

Field	Description
Send File as Email Attachment	Select the checkbox to send the file as an attachment. There is a limit on the attachment size (25 MB) that can be adjusted via the `System.To.Human.EmailAttachmentMaxFileSize` configuration option. If a file exceeds the maximum attachment size, SecureTransport will not send the email and will record an error. This option is not available when the delivery method is set to `Challenge`.

Manage transfer sites

Use the *Transfer Sites* page to create, edit, and delete transfer sites.

The following topics provide how-to instructions for managing transfer sites:

- *Create a transfer site* on page 628
- *Edit a transfer site* on page 629
- *Delete a transfer site* on page 629

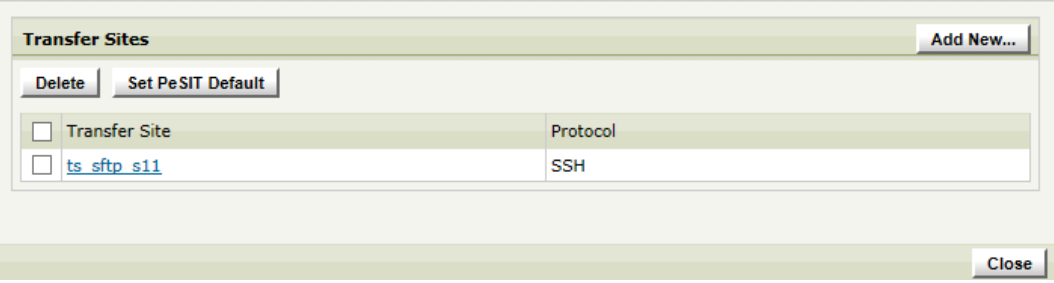
Create a transfer site

This topic provides a general procedure for creating a transfer site for an account. All supported protocols require custom settings. For details, see the reference topic for each protocol.

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click the name of the account to which you want to add a transfer site.

   The *Account Settings* page for that account is displayed.
3. Click **Transfer Sites**.

   The *Transfer Sites* page is displayed.



4. Click **Add New.**

   The *Add Transfer Site* page is displayed.
5. In the **Site Name** box, enter a unique name for the transfer site.
6. The Site Name is unique per account. Two sites could have the same names if they are associated with different accounts.

7. Select an **Site Type**. Use this parameter to differentiate between sites that transfer files internally and those that transfer files between partners. Choose from the following:
    - **Unspecified** – Default value. All transfer sites created using previous versions of SecureTransport have this value.
    - **Internal** – Transfers for this site occur within a single organization.
    - **Partner** – Transfers for this site occur between organizations.
8. Select the protocol that the transfer site uses for file transfers. The supported protocols are AS2, FTP(S), HTTP(S), SSH (SFTP and SCP), PeSIT, Connect:Direct, Folder Monitor, System to Human, Generic-HTTP(S), and SharePoint.
9. By default, the AS2 protocol settings are displayed first. This example displays the settings for creating an FTP(S) transfer site.
10. Edit the custom options depending on the selected transfer protocol.
11. Click **Add**.

    The transfer site is added to the list of transfer sites available to the current account.

Edit a transfer site

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click the name of the account that owns the transfer site to edit.
3. Click the **Transfer Sites** tab.
4. Click the name of the transfer site to edit.
5. The *Edit Transfer Site* page is displayed.

Note Editing a transfer site does not affect transfers in progress, including transfers that are being retried. However, if you update the transfer site settings while SecureTransport is trying to get the list of files on the remote server during a server-initiated download, any consecutive *get file list* retries will use the updated settings.

6. Edit the desired settings for the transfer site.

Note The *Edit Transfer Site* page is identical to the *Add Transfer Site* page for the corresponding protocol.

7. Click **Save**.

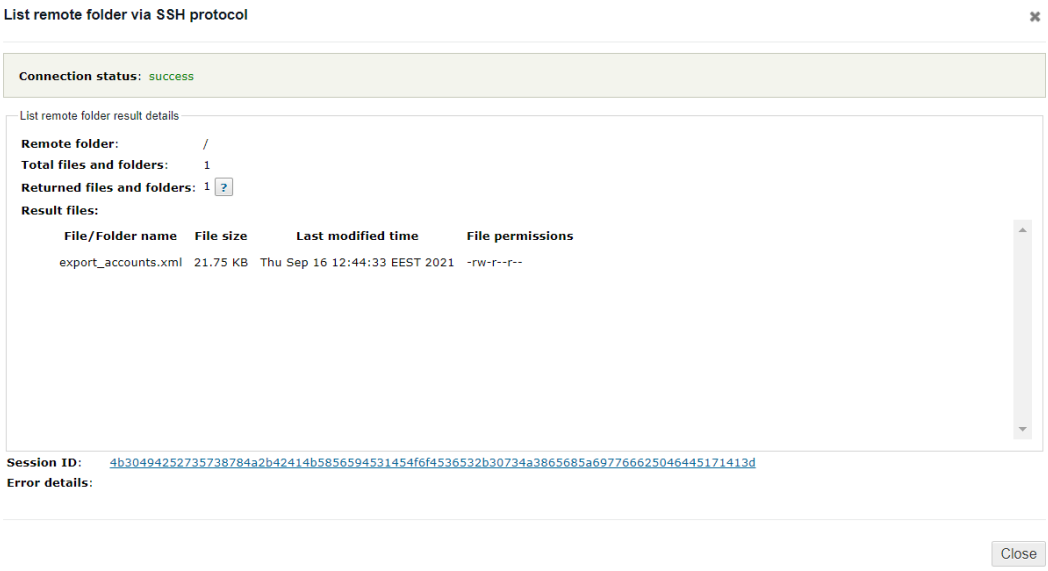
Delete a transfer site

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click the name of the account that owns the transfer site to delete.
3. Click the **Transfer Sites** tab.
4. Select the checkboxes next to the names of the transfer sites to delete.
5. Click **Delete**.

List the contents of the Upload or Download folder

The List Folder functionality is available for the following transfer sites: SSH, HTTP(S), FTP(S) and Folder Monitor. It allows you can check whether the Upload or the Download folder of a transfer site is accessible and list its content by clicking the **List** button below the folder path. Using the transfer site settings currently specified on the page, SecureTransport connects to the remote partner and presents a list of the remote folder's files and subfolders, along with the following information:

- Connection status - success or failed.
- The total number of files and folders in the remote folder. When a download pattern is specified in a HTTP(S) transfer site, the total number of files and folders is the same as the number of the filtered files.
- The number of listed files and folders. The quantity of listed entries is limited to 50 by default. You can adjust the limit to suit your needs by specifying a value for the `ListRemoteFolder.Result.Files.Limit` server configuration option.
- File permissions held by the user. Note that for Folder Monitor transfer sites, the file permissions field is empty for Windows folders.



Note This functionality is also available as a REST API resource: `/sites/operations`.

Note During the remote folder listing only a limited set of Expression Language functions can be used. For example, environment variables will not be evaluated. Custom properties passed in a REST API list request through the `/sites/operations` endpoint are supported.

Secure your transfer site with SSL/TLS

SecureTransport performs server-initiated transfers through its transfer sites. It supports a variety of protocols - FTP(S), Generic-HTTP(S), HTTP(S), PeSIT, AS2, and SFTP (SSH). While some are highly secure, others require an additional SSL/TLS security layer. Transfer sites that use FTP, HTTP, Generic-HTTP, and PeSIT have SSL encryption enabled by default in their configurations, advanced SSL options, and FIPS mode support. This topic describes how to configure them to secure your transfer sites.

Configure Advanced SSL Settings

The level of security that TLS provides is most affected by the protocol version and the allowed cipher suites. SecureTransport enables configuring which TLS version and cipher suites to permit for server-initiated transfers at two levels: per transfer protocol and transfer site.

The security configuration, defined at the transfer protocol level, represents default values that apply to all transfer sites that use that protocol. It can be modified at any time or overridden at the transfer site level.

Set the defaults

The transfer protocol determines the default enabled SSL/TLS versions and cipher suites for the transfer site. These TLS defaults are set per protocol via the following server configuration options:

Server Configuration Option	Description

`As2.SIT.EnabledProtocols`
`Pesit.SIT.EnabledProtocols`
`Ftps.SIT.EnabledProtocols`
`Https.SIT.EnabledProtocols`
	Determines which TLS protocols are allowed for server-initiated transfers over the respective protocol. The value is a list of one or more comma-separated protocol versions. Starting 5.5-20210930, TLSv1.2 and TLSv1.3 are enabled by default on new installations. On upgraded instances, you need to enable TLSv1.3 manually by following the instructions in the Security Guide (login required).

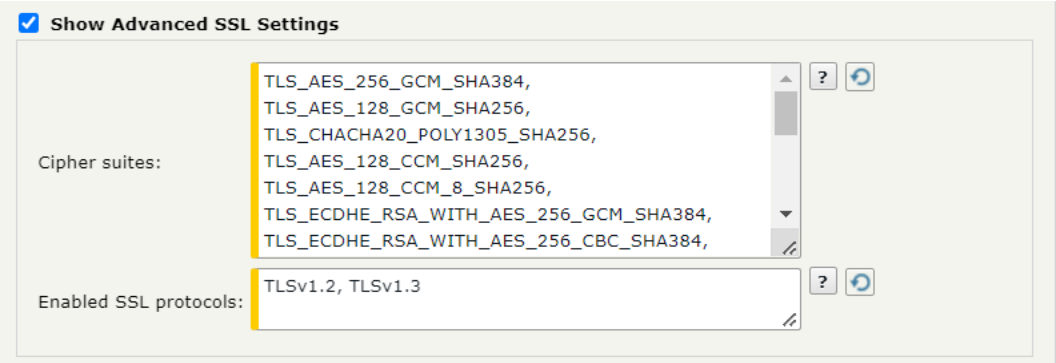
`As2.SIT.Ciphers`
`Pesit.SIT.Ciphers`
`Ftps.SIT.Ciphers`
`Https.SIT.Ciphers`
	Determines which cipher suites are allowed for server-initiated transfers over the respective protocol. The value is a list of one or more comma-separated protocol cipher strings. You can edit it to add any of the supported cipher suites listed here.

By default, the configuration options are set to contain the most secure and recommended TLS versions according to the Axway Secure By Default policy but you can override it if required.

Note Changes made to the configuration option will not affect existing transfer sites.

Set custom SSL configuration for a transfer site

You can use the Advanced SSL Settings section in the transfer site definition to specify a custom TLS version/cipher suite combination for that transfer site. These settings become available after you enable SSL on the transfer site and select the **Show Advanced SSL Settings** checkbox. Be aware that you can only add cipher suites from the list here.



Enable FIPS transfer mode

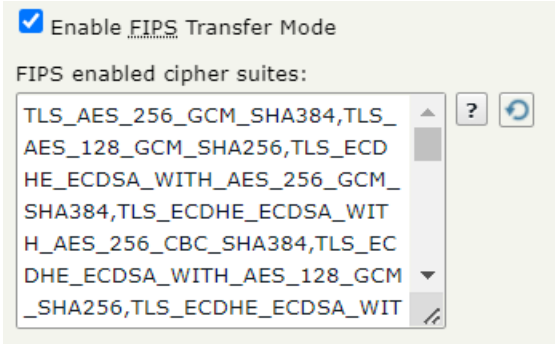
Enabling FIPS mode limits SecureTransport to encryption algorithms certified to be FIPS 140-2 compliant. A complete list of FIPS-compliant cipher suites that SecureTransport supports is available here (login required).

By default, transfer sites use non-FIPS mode. Consider the following before you enable FIPS mode:

- TLS 1.0 or 1.1 are disabled by default. Although we do not recommend using these protocol versions, you can enable them using the configuration option `FIPS.DisabledProtocols`.
- Both the sender and the recipient must use supported FIPS cipher suites. Otherwise, the transfer will fail.
- On update, SecureTransport ensures that the latest supported FIPS ciphers will be applied. This can lead to a configuration change to preserve the FIPS mode.

To enable FIPS mode for a transfer site:

1. Enable SSL on the transfer site.
2. Select the **Enable FIPS Transfer Mode** checkbox.



3. Optional: Click in the **FIPS enabled cipher suites** field to customize the list of allowed cipher suites in FIPS mode. Otherwise, SecureTransport will use the cipher suites listed in the configuration option `<protocol>.FIPS.SIT.Ciphers` for the corresponding selected transfer protocol. Note that you can only add supported cipher strings, see the full list.

Set Alternative addresses

This feature is available for the following transfer sites: AS2, HTTP(S), FTP (S), SSH, and PeSIT. It allows you to add, delete and set a priority order of alternative endpoints. These endpoints act as backup alternatives to the configured Server-Port Site Settings and are particularly useful in cases of transfer failures. Specifying alternative endpoints as backup servers provides a way to temporarily reroute pending transfers and minimize the risk of transfer failure. For AS2 transfer sites, the connection to each alternative endpoint is defined by its URL. For the rest supported transfer sites, the connection to each alternative endpoint is defined by its host name (or IP address) and port number.

- To add an alternative server endpoint, click **New Address**. The Alternative Addresses table expands with a new row, that allows you to enter a hostname (or IP address), a port number and save these changes.
- To delete an alternative server endpoint, select the corresponding check-box on the same row and click **Delete**.
- To reorder the list of alternative endpoints, click **Reorder**. A new option (upward and downward arrow) appears next to each entry. You must hover with the mouse pointer over this newly appeared option and the mouse pointer will assume the "move" shape: a four-directional arrow pointer. This indicates which alternative endpoint is in focus. You can now drag & drop it up and down to the order number you want it at. Perform this action with other alternative endpoints until the list is ordered according to your needs. When you are done, click **Save Order** to keep the newly changed order.

The Alternative Addresses feature is disabled by default. To enable it, go to **Operations > Server Configuration** and set the policy type using either of the following values:

- `AllHostsOnEachRetry` – with this policy SecureTransport iterates through each endpoint, one by one, starting with the first in the list. If connection not successful, SecureTransport will continue trying each endpoint one after another until the maximum number of retries is reached. You can set the maximum retry value by editing the `EventQueue.maxRetryCount` configuration option.
- `OneHostOnEachRetry` – with this policy SecureTransport tries to connect to the first endpoint in the list. If connection not successful, SecureTransport will continue trying that endpoint until the maximum number of retries is reached; and then will move to the next one in the list. Following that same pattern, SecureTransport will try each endpoint until success; or until end of list. You can set the maximum retry value by editing the `EventQueue.maxRetryCount` configuration option.
- `Disabled` (default) – this is the default value that keeps the table with endpoints entirely hidden from view.

Set post-transmission actions in transfer sites

Post-transmission actions are file operations, such as rename, move, and delete, that are performed after a file transfer ends. In SecureTransport, post-transmission actions can be configured per transfer site or as part of a Standard Router or an Advanced Routing subscription. This topic describes how to set a post-transmission action at the transfer site level.

The following built-in transfer sites can be set to trigger post-transmission actions: FTP(S), HTTP(S), SSH, and Folder Monitor. For those used for SIT pulls and pushes, post-transmission actions are configured independently for the files being sent and received based on the status of the transfer.

Understanding statuses

A file transfer can have three statuses:

- *Success* - indicates a successful file transfer.
- *Temporary Failure* - indicates that the transfer has failed and that the server will retry it at least one more time. Note that some of the failures (like authentication failures) are not retried. In this case, the transfer goes directly to the permanent failure state and post-transmission actions, assigned to SIT temporary failure, are skipped.
- (Permanent) *Failure* - indicates that the transfer is incomplete and all retry attempts were unsuccessful. By default, the number of retries is 5 and can be changed via the configuration option `EventQueue.maxRetryCount`.

Available actions

You can choose the appropriate action to be executed based on the status of the transfer. The available actions vary based on the protocol used.

Action	Description
No Action	No post-transmission action is performed, resulting in the same file on both sending and receiving sides. If another file with the same name is transferred to the specified location, the original file is overwritten.

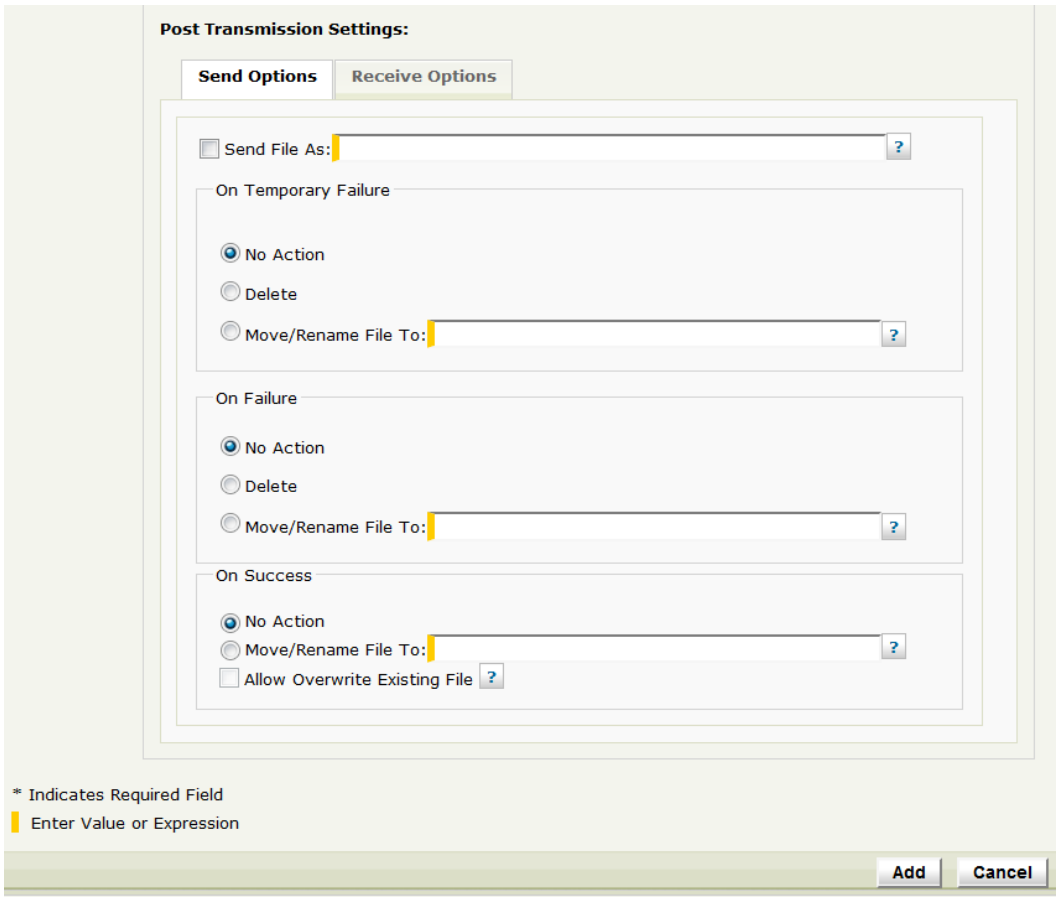
Action	Description
Delete	The action does one of the following: • after successful transfer, it removes the source file from the sending side. When SecureTransport Server receives a file from a remote partner, the remote files gets deleted. • after temporary or permanently failed transfer, it removes the partially received file from the receiving side. When SecureTransport Server sends a file to a remote partner, the partial remote file gets deleted. When SecureTransport Server receives a file from a remote partner, the partial local file files gets deleted.
Move/ Rename File To	After a transfer ends, SecureTransport moves the remote file to a new directory on the same machine, providing an option to specify a new file name. This action is available for files transferred over SSH and FTP(S). The field supports Expression Language. If you want to preserve the filename, use the `${stenv.target}` or `${stenv['target']}` variables. Use the **Allow Overwrite** option to specify the desired behavior in case a file with the same name already exists in the target directory. If selected, the existing file with be replaced with the source file. If not selected, SecureTransport will abort the post-transmission action and show the transfer with a FAILED_SUBTRANSMISSON status.

Configuration in the transfer sites

In transfer sites used for SIT pulls and pushes, post-transmission actions are configured under Post Transmission Settings independently for the files being sent and received.

Send Options

In the *Send Options* tab, you specify the post-transmission actions for outgoing files, i.e., server-initiated pushes to a remote partner. The transfer site serves as a source. All post-transmission actions are performed on the destination file on the remote server, and not on the source file on the local server.



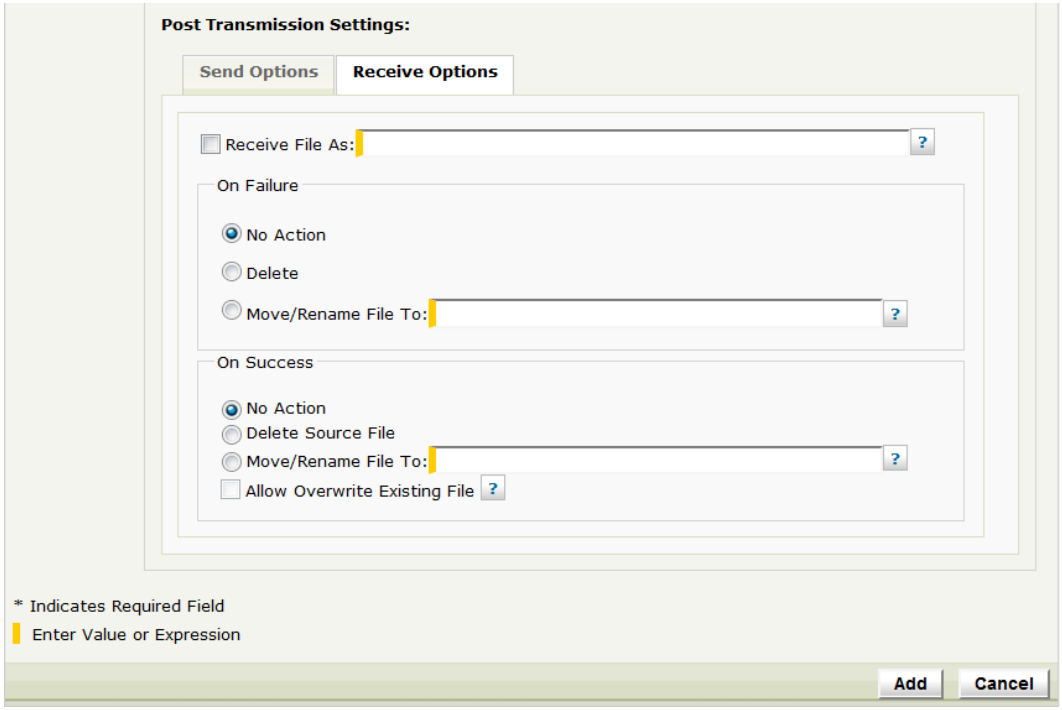
1. Select the **Send File As** checkbox to specify a file name pattern for the files at the destination. SecureTransport directly pushes the source file with the new name to the remote server. The filename pattern may consist of different EL variables. See *Expression Language on page 1104*. The variable `${stenv.target}` contains the name of the file being sent.

   For example, `${stenv.target}_${timestamp}` constructs the name of the destination file by appending a timestamp to the local filename.

2. Specify an action to be performed in case the push fails temporary or permanently:
   - **No Action**
   - **Delete** the partially received file from the remote server (available for SSH, HTTP, and FTP transfer sites)
   - **Move/rename** the partially received file to another directory on the same remote server (available for SSH and FTP transfer sites only)
3. Specify an action to be performed after a successful push:
   - **No Action**
   - **Move/rename** the destination file to another directory on the same remote server (available for SSH and FTP transfer sites only)

Receive Options

In the *Receive Options* tab, you specify the post-transmission actions for incoming files, i.e., server-initiated pulls from a remote partner to the particular transfer site. The transfer site serves as a destination, the source file is located on the remote server.



1. Select the **Receive File As** checkbox to specify a filename pattern for incoming files. Files are received directly with the new name and saved on the local server.
   The pattern may consist of different EL variables. See *Expression Language on page 1104*. To reference the file being pulled, you can use the SecureTransport-specific variable `${stenv.site_target}`, which takes the value from the remote file path.
2. Specify an action to be performed in case the pull fails temporary or permanently:
   - **No Action**
   - **Delete** the partially received file from the local server (available for SSH, HTTP, and FTP transfer sites)
   - **Move/rename** the remote file to another directory on the same server (available for SSH and FTP transfer sites only)
3. Specify an action to be performed after a successful pull:
   - **No Action**
   - **Delete** the source file from the remote server (available for SSH, HTTP, and FTP transfer sites)
   - **Move/rename** the source file to another directory on the same remote server (available for SSH and FTP transfer sites only)

   To avoid downloading the same file again, use **Delete** or **Move/Rename File To**.

Specifics

The available post-transmission actions and transfer statuses vary based on the protocol used. For example, the HTTP protocol does not support remote renaming of files.

File tracking

Detailed information about a file transfer, including a chronologically ordered list of the post-transmission actions and transformations performed with it, is available in the transfer's *Status Details*. See *View file transfer information* on page 309.

If a file has been transferred successfully, but a post-transmission action fails, the transfer will be marked with  (FAILED_SUBTRANSMISSON) on the *File Tracking* page. You can use this transfer status in the search criteria under Advanced Search.

How to use DXAGENT_TRANSFERSAPI variables in transfer sites

The environment variables with the `DXAGENT_TRANSFERSAPI` prefix can be used in the transfer site definition fields to reference properties submitted via the REST API `customProperties` parameter of the `/transfers/operations` request body.

1. On the *Add Transfer Site* page or the *Edit Transfer Site* page, in the desired property field, type: `${DXAGENT_TRANSFERSAPI_*}`, where * is an arbitrary name.
   For example, ${DXAGENT_TRANSFERSAPI_FOO}.
   Concerning the use of Expression language in fields:
   - FTP, HTTP, SSH, PeSIT and Folder Monitor support Expression Language in all fields of type string.
   - FTP, HTTP and SSH support Expression Language also for the certificates (only when provided via the Admin UI).
   - Refer to the schema for each transfer site for complete picture and understanding of the supportability.

   PeSIT Acknowledgements are not supported when **Expression Language** is used in a transfer site field related to the following fields:
   - Pre-Connection phase,
   - Server and Partner passwords,
   - Checkpoint Interval,
   - Checkpoint Window.

Important note: Using Expression Language in these fields while PeSIT Acknowledgments are configured could lead to locking of the account.

2. To trigger a server-initiated transfer using the */transfers* RESTful API resource, you need to set the value of the dynamic property in the request body.

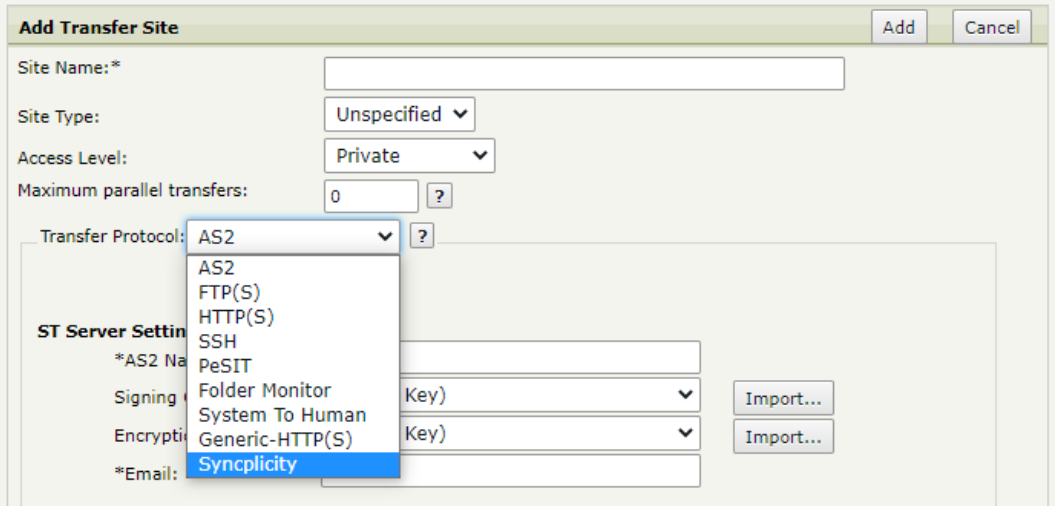
Note Resubmit and Retry are not supported when using `${DXAGENT_TRANSFERSAPI_*}` expressions.

Pluggable Transfer Sites

You can extend SecureTransport to support other transfer protocols to exchange files with specific systems and object storage solutions by using an Axway pre-built or a custom connector. Connectors developed and supported by Axway are available from the Amplify Repository and the Support website under Type: "Product extensions". These connectors are deployed on SecureTransport Server by unzipping the package in the `<FILEDRIVEHOME>/plugins/transferSites` directory. In a clustered environment, the connector must be deployed on all nodes.

To build and deploy your own connector, use the SecureTransport SDK and follow the instructions given in the Developer's Guide. See Custom Connectors.

Once you deploy the connector, the new transfer protocol appears in the **Transfer Protocol** drop-down on the **Add Transfer Site** page. You can select it and configure transfer sites for your user accounts. Transfer sites created using connectors are referred to as pluggable throughout SecureTransport documentation.



Pluggable Transfer sites are used in almost the same way that you use the built-in transfer sites. There are only few specifics:

When using expressions to configure a pluggable transfer site, there is a difference in the variable spaces depending on the account type:

- With standard user accounts, the flow attributes of a file are accessible with an expression in the following format:
  `${ts['userVars.test']}`

The `${ts['userVars.test']}` expression is applicable only for evaluating the Upload folder and a post-transmission action (PTA); it does not work with the Download folder or a pattern.

- With user account templates, the flow attributes of a file are accessible with an expression in the following format:
  `${stenv['DXAGENT_SUBSCRIPTION_ATTR_userVars_folder']}`

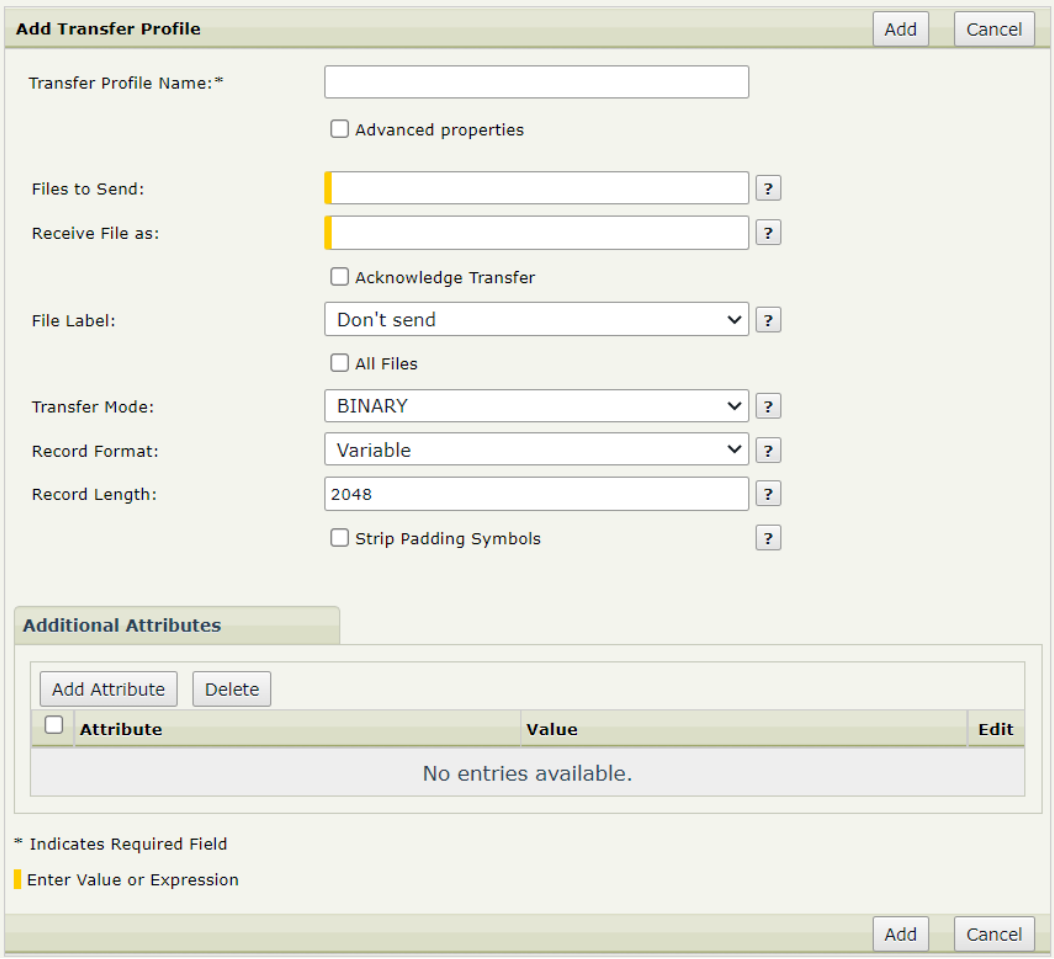
Transfer profiles

A transfer profile is a set of rules that a PeSIT transfer site uses when receiving or sending files, or when receiving a message. Before you are able to configure a transfer profile, you need to have an account with a PeSIT transfer site. Then, you can create a transfer profile following the instructions in this section. The final step is to reference it by its name in a subscription that retrieves files from a PeSIT transfer site or sends files directly to one, or a subscription that triggers an Advanced Routing flow upon the arrival of a message. You can also set a default PeSIT transfer site and transfer profile for each account. If you don't explicitly select a transfer site or a profile in the subscription, SecureTransport will use the default ones to perform the transfer.

Create a transfer profile

Use the following procedure to create a transfer profile.

1. Select **Accounts > User Accounts** to see a list of all user accounts.
2. Click the name of the account to which to add a transfer profile.
3. Click the **Transfer Profiles** tab.
4. Click **Add New**.
   The transfer profile basic configuration tab is displayed.



5. If the files that are to be sent and received via the transfer profile have the same characteristics (encoding, record length and format) on the source and target system, you can do so with the basic configuration without activating the advanced properties. See *Transfer Profile: Basic Configuration on page 642*. If you want to set different file transformations based on the transfer direction (incoming or outgoing), or if you want to receive PeSIT messages, you need to enable and configure the Advanced Properties. See *Transfer profile: Advanced Properties on page 645*.

   Note In a single transfer profile, you can set either the Basic configuration or the Advanced properties but not both.

6. Based on you selection in the above step, configure the required properties.
7. Click **Add**.

   The new transfer profile is added to the list for the user.

Set a default transfer profile for an account

Go the account's **Transfer Profiles** tab where you should see a list of all transfer profiles configured for that account. Select the one you want to set as default and click the **Set As Default** button.

Edit a transfer profile

Use the following procedure to edit a transfer profile.

1. Select **Accounts > User Accounts**.
2. On the *User Accounts* page, click the name of the account that owns the transfer profile you want to edit.
3. Click the **Transfer Profiles** tab.
4. Click the name of the transfer profile to edit.

Note Editing a transfer profile does not affect transfers in progress, including transfers that are being retried.

5. Edit the desired settings.

Note The *Edit Transfer Profile* page is identical to the *Add Transfer Profile* page.

6. Click **Save**.

Delete a transfer profile

Use the following procedure to delete a transfer profile.

1. Select **Accounts > User Accounts**.
2. On the the *User Accounts* page, click the name of the account that owns the transfer profile you want to delete.
3. Click the **Transfer Profiles** tab.
4. Select the checkboxes next to the names of the transfer profiles to delete.
5. Click **Delete**.

Transfer Profile: Basic Configuration

The Basic Configuration allows you to specify one transfer mode and one type of record format transformation to all files transferred (both sent and received) via the profile. If you need to set a different file transformation based on the transfer direction, use the Advanced Properties. See *Transfer profile: Advanced Properties on page 645*.

Advantages:

- Ability to remove padding while retrieving a file
- Dedicated option for setting `EBCDIC_NATIVE` transfer mode
- User input gets automatically populated when enabling Advanced Parameters

Limitations:

- One transfer mode, one type of format record transformation for all transferred files regardless of the transfer direction
- The transfer profile cannot be used to receive PeSIT messages

The Basic Configuration consists of the following fields:

Field	Description
Transfer Profile Name	The name you use to select the transfer profile in the subscription. It must matches the IDF (PeSIT protocol PI 12) on a remote PeSIT partner.
Files To Send	Determines the files to be sent to the partner. You can specify a file name pattern using wildcard characters or a regular expression. For client-initiated transfers only, you can also enter the actual name of the file. The filed is evaluated relative to the user's home folder.
Acknowledge Transfer	When selected, SecureTransport automatically acknowledges any incoming transfer after it ends and has been processed successfully.
Receive Files As	Sets the name of a received file. It is either an exact file name or a regular expression. The file name patterns cannot include the wildcard character * or ?. The filed is evaluated relative to the user's home folder.
File Label	Controls the value of PeSIT PI 37 in sending mode: • PI 37 contains the file name relative to the user's home folder • PI 37 contains the full file path and file name • PI 37 is empty This parameter is ignored when receiving files.

Field	Description
All files	When the checkbox is selected, SecureTransport will retrieve all available files. When the checkbox is not selected, SecureTransport retrieves only the first found file.
Transfer Mode	`BINARY`, `ASCII`, `EBCDIC`, or `EBCDIC_NATIVE`. Use `EBCDIC_NATIVE` to transfer EBCDIC files that use percentage sign "%" (0x25) as the record delimiter unaltered
Record Format	Fixed or Variable.
Record Length	Specifies the length of the records in bytes. The specified **Record Format** and **Record Length** apply only to files that do not have PI 31 and PI 32 defined. Those two PeSIT parameters describe the transmitted file; they are sent by the sending partner to the remote partner in the transfer request and stored as file's transfer attributes. The original PI 31 and PI 32 values do not change: they are used for each subsequent resend of the file, ignoring the transfer profile settings.
Strip padding symbols	When **Strip padding symbols** is selected, the padding characters will be removed. This is the default behavior. When the checkbox is not selected, the padding characters will not be removed. Note that this is only applicable for incoming transfers when Fixed Record format is set by the remote site. Padding symbols are added by the sending partner. SecureTransport recognizes **null** when transfers with BINARY mode, **space** for ASCII mode, and **@** for EBCDIC and EBCDIC NATIVE modes.

Field	Description
Additional attributes	Add custom attributes as *attribute:value* pairs with your transfer profile. See *Additional attributes* on page 759. *Add an attribute* • Click **Add Attribute** and fill in the **Attribute** and **Value** fields. • Click the Save (✓) icon. Repeat the process to add more additional attributes, if necessary. *Delete an attribute* Select the corresponding check-box of one or more attributes and then click **Remove**.

Transfer profile: Advanced Properties

Using the Transfer profile's Advanced Properties you can configure different transformations for files being sent and received over the PeSIT protocol. You can also configure the settings for receiving PeSIT messages.

Advantages:

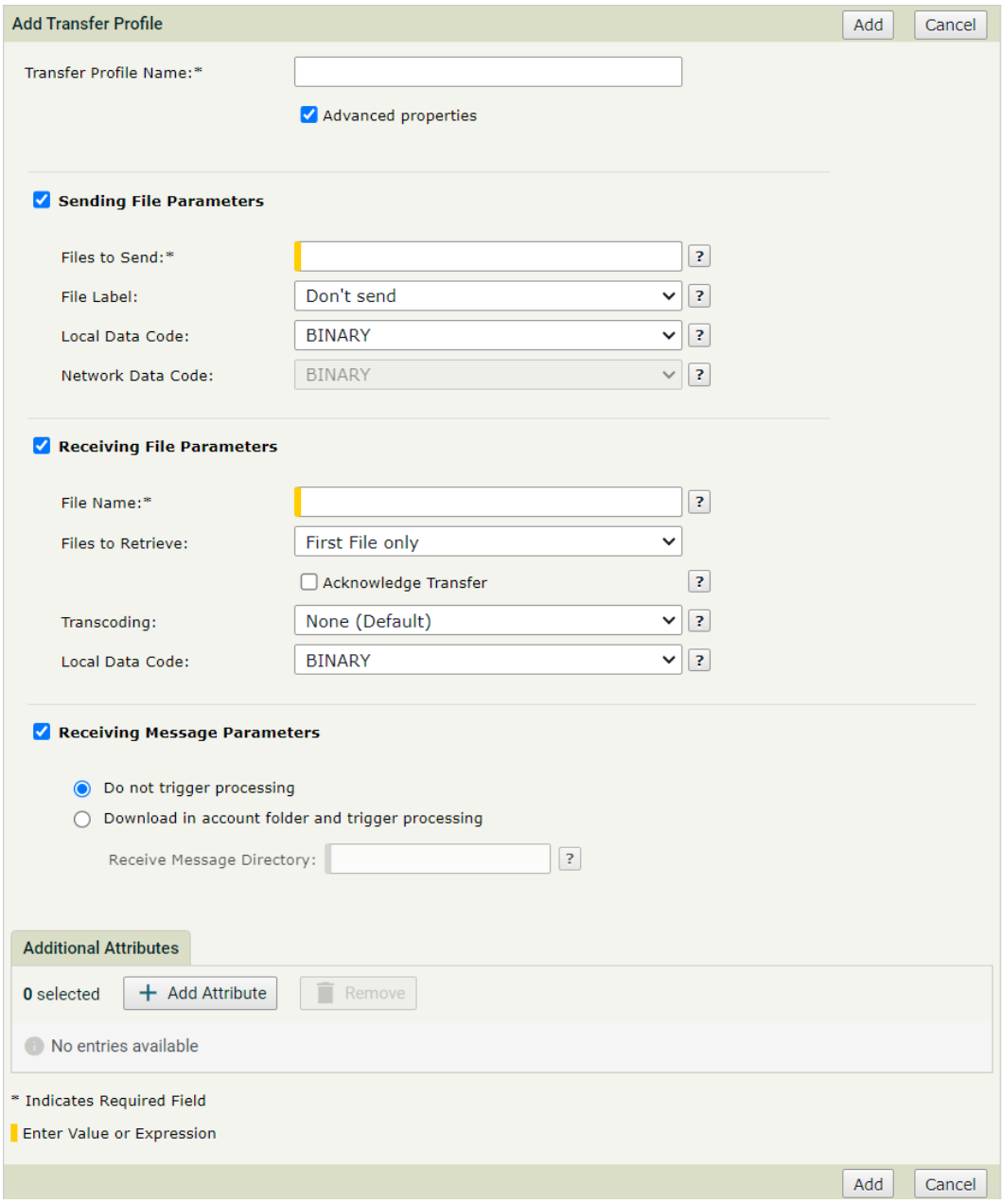
- Clearer UI with sending and receiving parameters arranged into separate groups
- Ability to set different file transformations depending on the transfer direction (send or receive)
- Ability to set the padding character for fixed-length text files
- Ability to set the line ending format of the retrieved files
- Completely covers the functionality of the AR Encoding Conversion step
- Ability to receive PeSIT messages

Limitations:

- No option to strip padding
- User input in Advanced Properties is neither reflected nor saved in the Transfer profile's basic configuration
- Custom translation tables are not supported

Overview of the configuration procedure

1. Enter a name for the transfer profile. This name must match the IDF (PI 12) defined on the remote site.
2. Select the **Advanced Properties** checkbox.
3. Depending on purpose of the transfer profile, enable one or more of the parameter groups:
    - Sending File Parameters are evaluated during outgoing transfers - CIT downloads and SIT pushes. See *Sending File parameters on page 647*.
    - Receiving File Parameters are evaluated during incoming transfers - CIT uploads and SIT pulls. See *Receiving File parameters on page 649*.
    - Receiving Message Parameters are evaluated during incoming PeSIT message transfers. See *Receiving Message parameters*.



4. Optionally, define additional attributes. See *Add an attribute on page 645*.
5. Click **Add** to save the transfer profile.

Sending File parameters

This group of parameters determines the characteristics of the files being sent.

1. In the **Files to send** field, enter a file name pattern using wildcard characters or a regular expression to specify the files to send. For client-initiated transfers only, you can also enter the name of the file. The field is evaluated relative to the account's home folder.

2. Select whether to send or not **File Label** (PeSIT PI 37). The receiving partner may use PI 37 in an expression to construct the names of the retrieved files.

   You can choose from the following options:

   - **Don't Send**- PI 37 is empty.
   - **Send Filename** - PI 37 contains the name of the file at the source.
   - **Send Path And Filename** - PI 37 contains the name of the file at the source including the full path to it.

3. From the **Local Data Code** drop-down, select the code (method of encoding characters) of the file to be sent at the source. You can choose from BINARY, ASCII, and EBCDIC.
   - Binary files will be transferred unaltered when **Local Data Code** is set to **Binary** at both the receiving and sending end.
   - EBCDIC files that use the percentage sign "%" (0x25) as the record delimiter will be transferred unaltered (EBCDIC_NATIVE mode) when **Local Data Code** is set to **Binary** at both the receiving and sending end.
   - Text files, ASCII or EBCDIC, can be optionally modified and transferred over the network in a format required by the target platform. Be cautious when configuring the transfer profile: the configurations on the sending and receiving system must be in accord. Note that if you send a text file (ASCII or EBCDIC) and the receiver's **Local Data Code** is set to **Binary**, all line endings will be removed, resulting in a single line file.
4. The **Transcoding** drop-down determines if and how a file is modified during transfer, e.g., the characteristics of the network file.
   - Select **None** to send the text file as is, without charset convention.
   - Select **Predefined** to modify the file and configure the encoding conversion made to the file during transfer. SecureTransport uses predefined translation tables to convert the transmitted data.
     a. From the **Network Data Code** drop-down, select the desired code (method of encoding characters) for the network file.
     b. From the **Source Encoding** drop-down, select the original character encoding of the file at the source.
     c. From the **Output Encoding** drop-down, select the character encoding for the file when being transferred over the network. It must be compliant with the selected **Network Data Code**.
5. Configure record format transformation if required.

   Specify the desired record format for the network file:

   - **Variable** - records can contain any number of bytes up to 32767.

     For example, if the format is set to *Variable* and the length is set to *2048*:
     - Records that are 2048 bytes or fewer are sent as is.
     - Records that are over 2048 bytes lead to a transfer failure and the file is not sent.
   - **Fixed** - all records in the file contain the same number of bytes up to 32767 that is specified in the **Record Length** parameter.

For example, if the format is set to *Fixed* and the length is set to *2048*:

- Records that are 2048 bytes are sent as is.
- Records that are over 2048 bytes lead to a transfer failure and the file is not sent.
- Records that are fewer than 2048 bytes are padded with the specified padding character.

You can specify the padding character by its hexadecimal value in the **Padding Character** field. If you don't, the default for the Data code is used: null for BINARY, space for ASCII, and @ for EBCDIC.

Note The specified **Record Format** and **Record Length** apply only to files that do not have PI 31 and PI 32 defined. Those two PeSIT parameters describe the transmitted file; they are sent by the sending partner to the remote partner in the transfer request and stored as file's transfer attributes in `<user_home_dir>/.stfs/attr`. The original PI 31 and PI 32 values cannot be overridden; they are used in each subsequent resend of the file.

Receiving File parameters

This group of parameters determines the incoming file transfer attributes.

1. In the **Receive File as** field, specify an exact name or file name pattern for the received files. You can use a regular expression that does not contain the wildcard characters "*" or "?".
2. From the **Files to Retrieve**, select whether to retrieve one or multiple files.
   - Select **All files** to retrieve all available files.
   - Select **First File only** to retrieve the first found file.
3. Select the **Acknowledge transfer** checkbox if you want SecureTransport to automatically acknowledge (ACK) a transfer after it ends and the file is processed successfully. If this option is disabled, SecureTransport will not send an ACK message for a successful transfer but administrators can manually acknowledge inbound PeSIT transfers positively (ACK) or negatively (NACK) from the File Tracking Page or via the REST API 2.0. See *Acknowledge a PeSIT transfer on page 315*
4. The selected **Transcoding** determines if and how the network file is modified during retrieval.
   - When **Transcoding** is set to **None**:
     If the type of the transferred file matches the selected **Local Data Code**, the file is retrieved and saved unaltered. Otherwise, the network file gets converted to the format specified in the **Local Data Code**. For example, if **Local Data Code** is set to **ASCII**, the incoming file is assumed to be ASCII format. Non-ASCII files will be converted to ASCII. EBCDIC files that use the percentage sign "%" (0x25) as the record delimiter will be retrieved unaltered (EBCDIC_NATIVE mode) when **Local Data Code** is set to **Binary** at both the receiving and sending end.
   - Select **Predefined** to configure encoding conversion. SecureTransport uses predefined translation tables to perform the specified conversion while retrieving the file and stores the file in the transformed format. To configure encoding conversion:

a. From the **Source Encoding Scheme** drop-down, select the expected character set encoding of the incoming files.

b. From the **Output Encoding Scheme** drop-down, select the desired character set encoding in which the files to be saved.

5. Configure record format transformation on retrieval.

   Specify the desired record format:

   - **Variable** - records can contain any number of bytes up to 32767.

     For example, if the format is set to *Variable* and the length is set to *2048*:

     - Records that are 2048 bytes or fewer are received as is.
     - Records that are over 2048 bytes are trimmed.

   - **Fixed** - all records in the file contain the same number of bytes up to 32767 that is specified in the **Record length** parameter.

     For example, if the format is set to *Fixed* and the length is set to *2048*:

     - Records that are 2048 bytes are received as is.
     - Records that are over 2048 bytes are trimmed.
     - Records that are fewer than 2048 bytes are padded with the specified padding character.

     You can specify the padding character by its hexadecimal value in the **Padding Character** field. If you don't, SecureTransport will use the default for the Data code: null for BINARY, space for ASCII, and @ for EBCDIC.

   Note The specified **Record Format** and **Record Length** apply only to files that do not have PI 31 and PI 32 defined. Those two PeSIT parameters describe the transmitted file; they are sent by the sending partner to the remote partner in the transfer request and stored as file's transfer attributes in `<user_home_dir>/.stfs/attr`. The original PI 31 and PI 32 values cannot be overridden; they are used in each subsequent resend of the file.

6. From the **Line Ending Format** drop-down, select the one with which the file to be saved.

   The options are:

   - **Default** - your operating system's native line endings
   - **Windows** (CRLF)
   - **Unix** (LF)

Receiving Message parameters

This group of parameters determines how SecureTransport handles incoming PeSIT messages. See *PeSIT message transfers* on page 321 for more information. If not enabled, any attempts to send PeSIT messages to SecureTransport will fail.

Select one of the available options:

- **Do not trigger processing** - SecureTransport will receive PeSIT messages without triggering additional processes.
- **Download in account folder and trigger processing** - SecureTransport will trigger further Advanced Routing actions upon receiving a PeSIT message. In the **Receive Message Directory** field, enter the relative path to the subscription folder where the PeSIT message will be received.
  You can open the tooltip for a list of expressions that can be used. Additionally, you can use the `${extract(pesit.msgData, '@', 1)}` expression which will extract the required data directly from the received PeSIT message. Note that `@` is the separator and `1` indicates the data's position in the message.

  Note The directory must not be set to `../<NOT_USED_DIR>/..` as it is not a valid path.

Transfer profile: Example configurations

This topic provides sample configurations as examples to help you learn the transfer profile's advanced properties and create your own transfer profiles.

When you set the advanced transfer profile, it is crucial that the configurations on the sending and receiving ends are in accord to ensure successful file transfers. The sending and receiving partners could be any PeSIT software, but the example configurations are demonstrated using SecureTransport at both sides.

- SecureTransport behavior: Transferring text files
- Example configurations:
  - 1. Transferring text files unaltered
  - 2. Transferring EBCDIC files unaltered
  - 3. ASCII to BINARY conversion at the receiving side
  - 4. ASCII to EBCDIC conversion at the sending side
  - 5. ASCII to EBCDIC conversion at the receiving side

SecureTransport behavior: Transferring text files

When transferring text files, you need to pay attention to the data code used. ASCII-to-Binary and Binary-to-ASCII transfer setups are not recommended, as they can result in invalid or corrupted files. Below are the SecureTransport behaviors observed under different configurations:

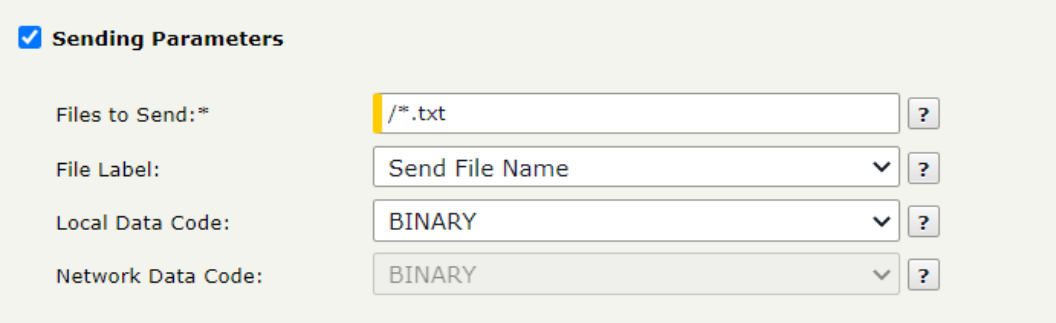
Sender's Local Data Code	Receiver's Local Data Code	Received file
BINARY	BINARY	The received file will be the same as the original.

Sender's Local Data Code	Receiver's Local Data Code	Received file
ASCII	ASCII	• If the original file ends with a line ending, the received file will be the same. • If the original file does not end with a line ending, a line ending will be added at the end of the received file.
BINARY	ASCII, fixed length	Only the last line of the file will be padded with the specified character to the specified length. The rest of the lines will retain their original length. A line ending will be added at the end of the file, regardless of whether the original file ends with a line ending. For example, if the original file has 10 lines, each with 5 characters, and the receiving side has set a fixed record length of 60 characters, the last line of the resulting file will have 10 padded characters.
BINARY	ASCII, variable length	A line ending will be added at the end of the file, regardless of whether the original file ends with a line ending.
ASCII, variable length	BINARY	All line endings from the original file will be removed and the file contents will be collected into a single line.
ASCII, fixed length	BINARY	A single-line file with no line endings, where each original row is padded to the configured length. For example, if the original file has 3 lines and is sent with a fixed record length of 50 characters (RL=50), the resulting file will be a single line with a total length of 150 characters.

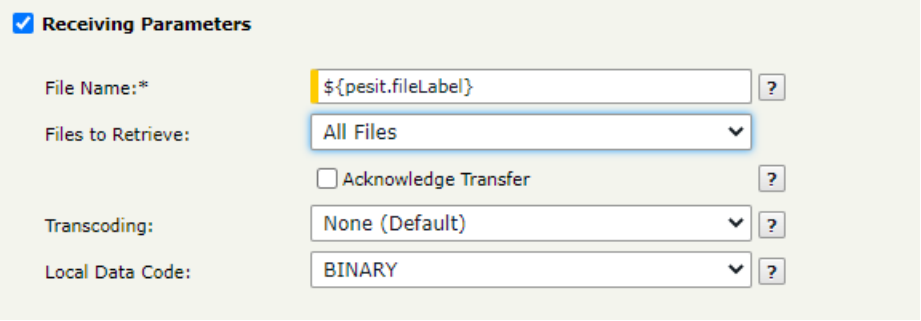
Example configuration 1: Transferring text files unaltered

This example shows how to transfer text files unaltered. The receiver uses the file name at the source to construct the name of the received file.

Sender partner configuration



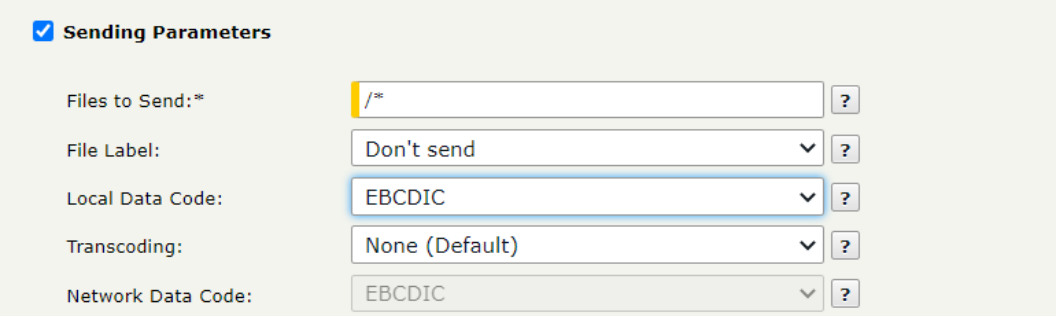
Receiving partner configuration



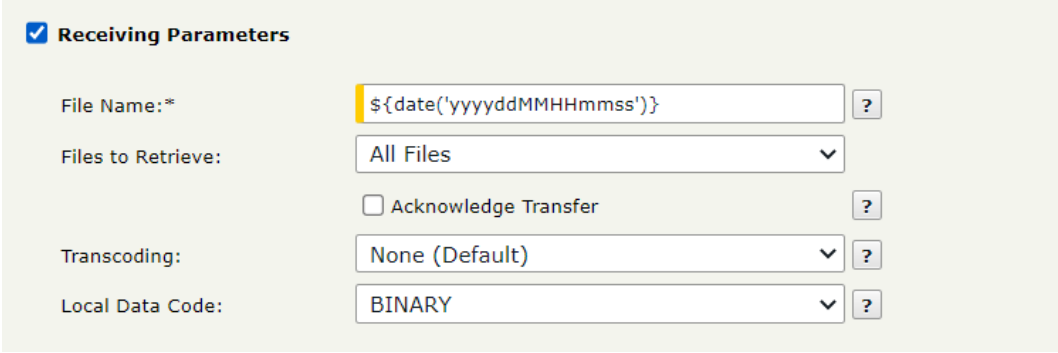
Example configuration 2: Transferring EBCDIC files unaltered

This example shows how to transfer EBCDIC files with "%" or NAK line endings unaltered, resulting in the same file on the source and destination system. The receiver uses *YYYY-MM-DD HHmmss* format for naming the received files.

Sending partner configuration



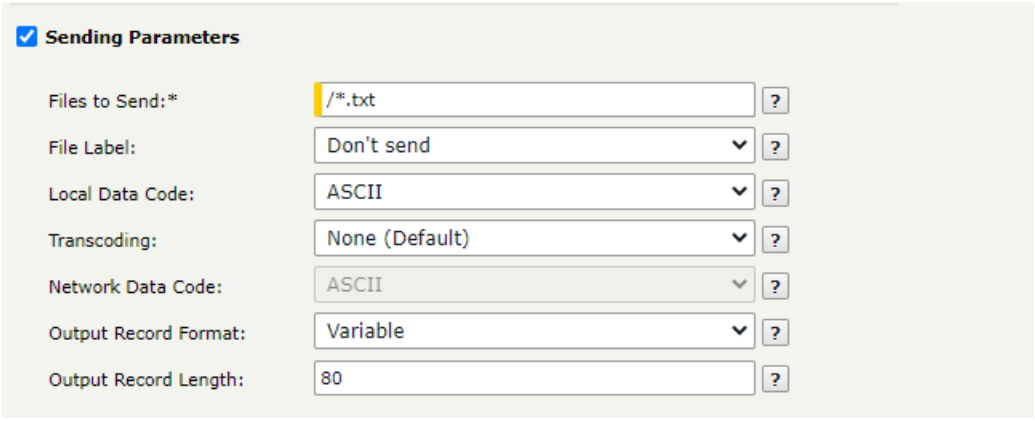
Receiving partner configuration



Example configuration 3: ASCII to BINARY conversion at the receiving side

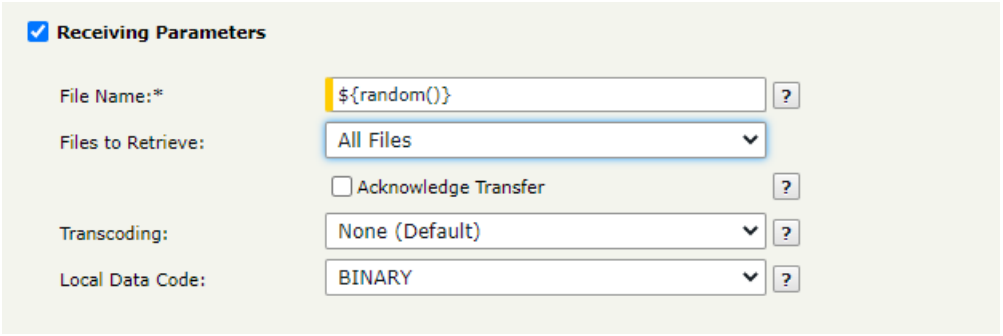
In this example, an ASCII file with variable-length records is sent unaltered. On reception, it is converted to a binary file.

Sending partner configuration



SecureTransport removes all line endings in the file during sending, resulting in a single-line file.

Receiving partner configuration

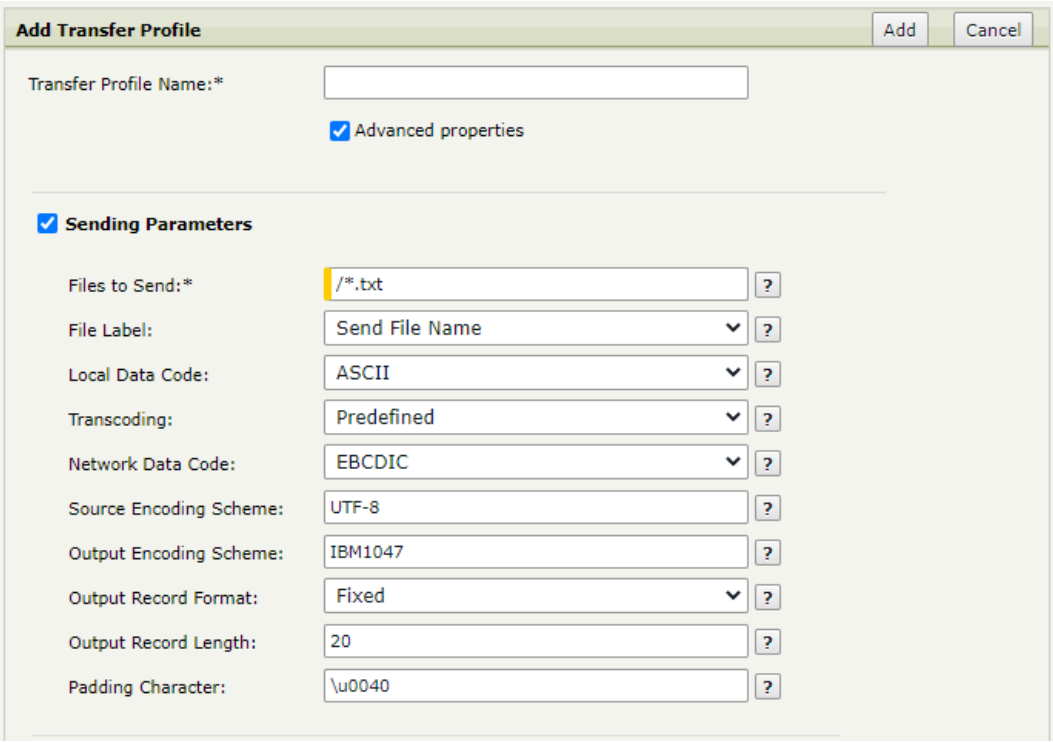


As a result, the single-line text file is converted to a binary file.

Example configuration 4: ASCII to EBCDIC conversion at the sending side

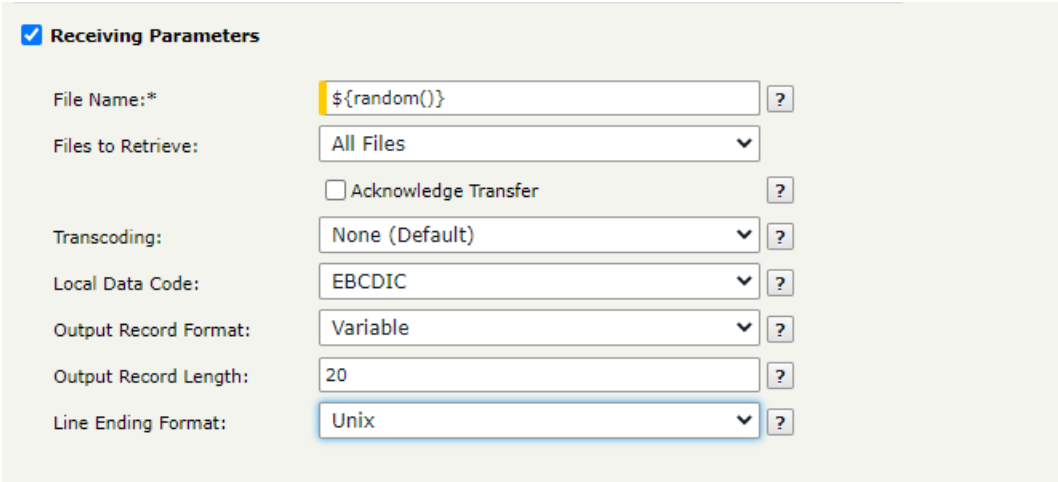
In this example, SecureTransport identifies the .txt files in the user's home folder and converts them from UTF-8 to IBM1047 with fixed-length records while sending them to the PeSIT partner. Upon receiving, the files get converted to variable-length records and saved using a file name pattern.

Sending partner configuration



During the sending process, the file is converted it to EBCDIC using a predefined UTF-8 to IBM1047 translation table. All line feed characters are deleted and record lines are padded to 20 bytes with "@" character. If the record length in the original file is more than 20 bytes, the file will fail to send.

Receiving Partner



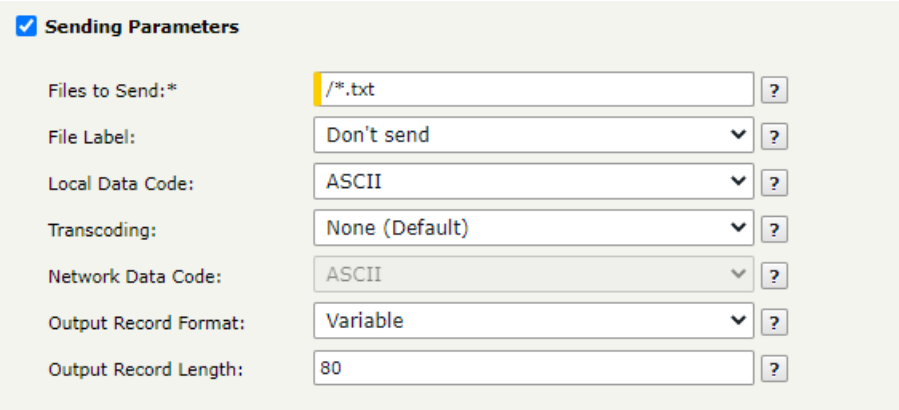
Since the **Local Data Code** matches the data code of the network file, no transcoding is applied. SecureTransport adds an LF at the end of each record.

Note With this configuration, the padding characters added by the sender will not be removed at the destination. If the receiving system expects a file without padding characters, we recommend using variable format in both sending and receiving configurations.

Example configuration 5: ASCII to EBCDIC conversion at the receiving side

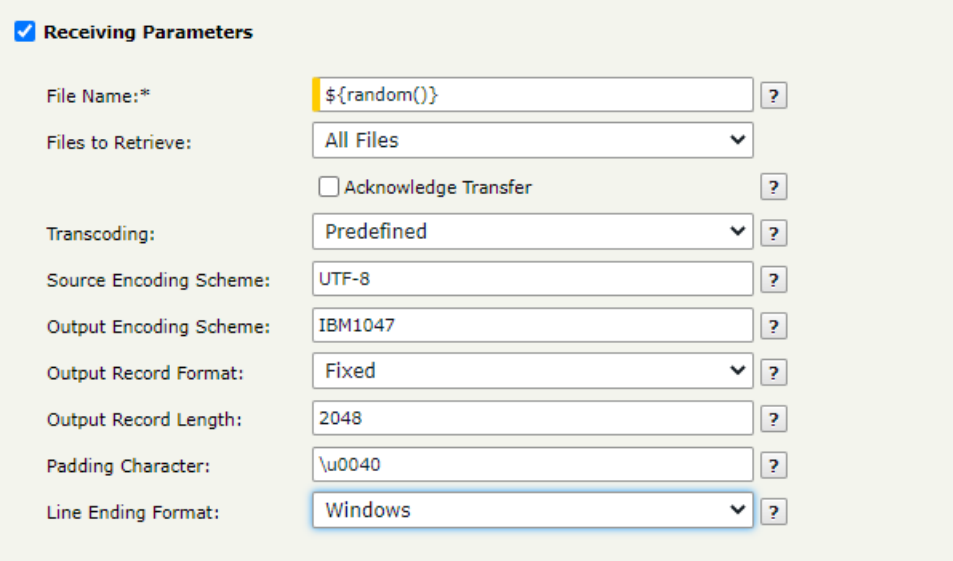
In this example, an ASCII file with variable-length records is sent without modification. On reception, it is converted to an EBCDIC file with code page IBM1047, fixed-length records of 2048, padded with "@".

Sending partner configuration



The text file is sent as is, with no modification.

Receiving partner configuration



The file is converted to EBCDIC. The record lines shorter than 2048 are padded to this length with "@", longer ones are trimmed to this size, and a CRLF is added at the end of each record.

Pulling multiple files via PeSIT: Example configuration

This topic describes the prerequisites and configuration steps for pulling multiple files via the PeSIT protocol. It provides you with the minimum set of parameters required to accomplish this use case.

The topic covers two main scenarios:

Scenario 1: Pull multiple files with SecureTransport via PeSIT from a partner as client

Scenario 2: Pull multiple files using a PeSIT Partner from SecureTransport as server

Configuration prerequisites

- All partner files must be in the same directory (first scenario)
- All SecureTransport files must be in the same directory (second scenario)
- The PeSIT listeners must be enabled
- A file processing application like Basic application, Advanced routing, etc. (only in the first scenario)

Note In the context of your daily operations, you can add functions and adapt the example configuration to your specific needs.

Pulling multiple files from a partner via PeSIT

Steps to configure SecureTransport:

1. Create a user account.
2. Create a new PeSIT transfer site.
    a. **Name**: Use the same name as you use in the Partner definition.
    b. In the **Remote Partner Settings**, enter values in the following fields:
        - **Host**: Enter the IP address of the Partner.
        - **Port**: Enter the TCP port for transfer access.

Note We identify the characteristics of the remote machine (Partner) to SecureTransport by creating a transfer site.

3. Create a new transfer profile (known as IDF in Axway Transfer CFT due to the usage of the file network identifier to generate a local file identifier - PeSIT PI 12).
    a. Set it as default.
    b. In the **Receive File As** field, enter `/${pesit.fileLabel}`.

    You can use any valid expression, including PeSIT expressions. For details, see *Expression Language* on page 1104.

c. Check the **All files** checkbox.

By selecting this option, when SecureTransport initiates a download from the PeSIT Partner, it gets all files that are made available by the server. This is one of the mandatory attributes for pulling multiple files. For details, see *Create a transfer profile* on page 640.

Note If **All files** is not selected, SecureTransport will download only one file.

4. Subscribe the account to the Basic application.
5. In the subscription, configure the following attributes:
   a. Select the **Automatically retrieve files from** checkbox.
   b. Select the transfer site created above from the drop-down.
   c. Locate the **Transfer Profile** attribute and select the transfer profile name from the drop-down.
   d. To submit the pull request, schedule server-initiated downloads from the transfer site by either clicking **Automatically retrieve files from** or click **Retrieve files now**.

Pull request result: SecureTransport pulls all the files available from the partner and shows details of each transfer on the *File Tracking* page.

Pulling multiple files using a PeSIT Partner from SecureTransport

Steps to configure SecureTransport:

1. Create a user account.
2. Create a new PeSIT transfer site.
   a. **Name**: To define the partnership, use the same name as you use in the Partner definition.
   b. In the **Remote Partner Settings**, enter values in the following fields:
      - **Host**: Enter the IP address of the Partner
      - **Port**: Enter the TCP port for transfer access.

Note We identify the characteristics of the remote machine (Partner) to SecureTransport by creating a transfer site.

3. Create a new transfer profile (known as IDF in Axway Transfer CFT due to the usage of the file network identifier to generate a local file identifier - PeSIT PI 12).
   a. Set it as default.
   b. In the **Files to send** field, enter `/*`.

   You can use any valid expression including PeSIT expressions. For details, see *Expression Language* on page 1104.

   Note If an exact filename is specified, then the client/partner will download only this file.

c. Check the **All files** checkbox.

If this option is selected, when a PeSIT Partner initiates a download from SecureTransport, it gets all files whose names match the pattern. This is one of the mandatory attributes for pulling multiple files. For details, see *Create a transfer profile* on page 640.

Note The **All files** checkbox must always be selected on the server side, unless when an exact filename is specified in the **Files to Send** field.

d. In the **File Label** drop-down, select **Send File Name**.

Pull request result: The PeSIT partner pulls all the files available from SecureTransport. Details for each transfer, initiated by the PeSIT Partner, are displayed on the *File Tracking* page.

Subscriptions

A subscription provides a functional connection between a user account and an application.

For each subscription, SecureTransport creates a subscription folder and stores and manages all files that are transferred or transformed as a result of the application activity. A single application can have subscriptions from multiple accounts and a single account can subscribe to multiple applications. An account can subscribe to new instances of the same application as long as each instance has a unique subscription folder name.

Additional transfer configurations are possible for subscriptions. Use subscriptions to trigger the execution of specific actions, defined in the respective application, when a subscription event occurs, such as an incoming file transfer in the dedicated subscription folder.

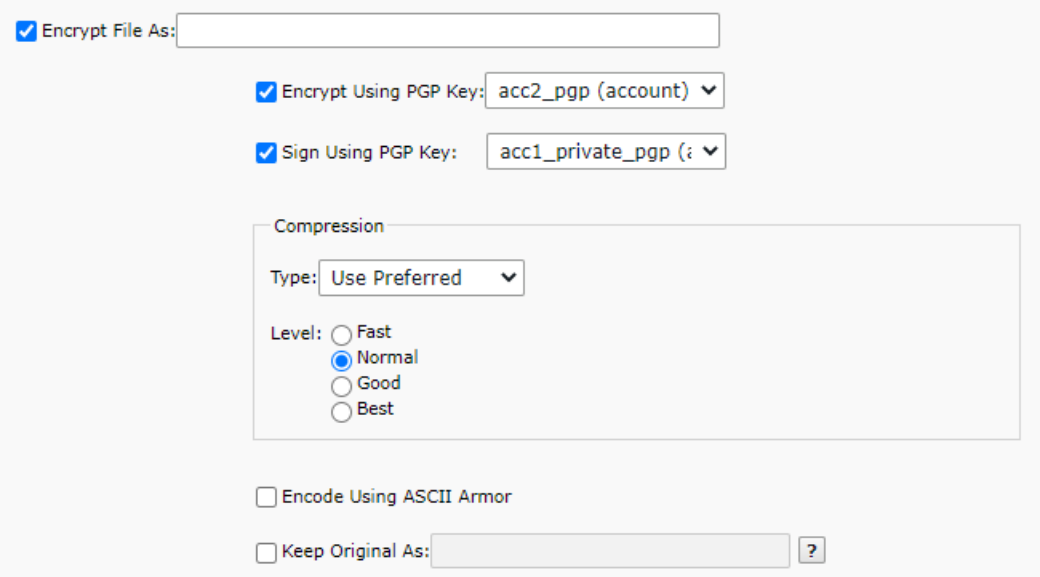
Note The application is not triggered if the file is uploaded into another folder first and is then moved or copied into the subscription folder.

The following topics describe how to manage subscriptions:

- *Manage subscriptions* on page 664 - Provides how-to instructions for managing subscriptions.
- *Encryption options* on page 659 - Lists the subscription encryption options.
- *Post-transmission actions* on page 661 - Lists the subscription post-transmission actions.

Encryption options

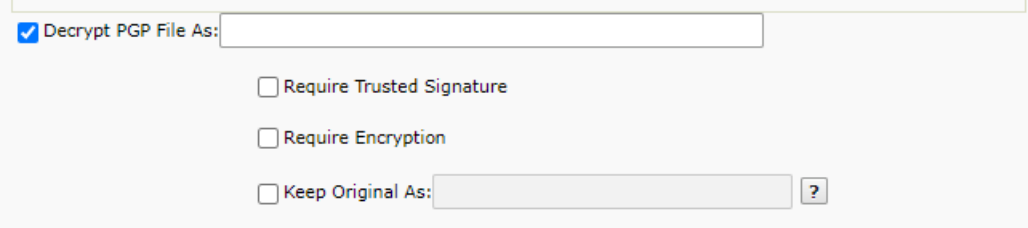
SecureTransport can encrypt a file before transferring it.



You can specify the following options for files you are transferring:

Encryption setting	Description
Encrypt File As	Determines if the file should be encrypted and if the encrypted file is saved to a different name, location, or both. You can use a file name expression.
Encrypt Using PGP Key	Determines if SecureTransport uses PGP to encrypt the file and which PGP key it uses. Select whether to PGP encrypt an outgoing file with a public key.
Sign Using PGP Key	Determines if SecureTransport signs the file using a PGP key and which PGP key it uses.
Compression	Determines what type of compression is used. Choose from ZIP, ZLIB, or BZIP2. You can also choose to use no compression or to use preferred compression. Preferred compression methods and order of preference are determined by examining the recipient's PGP key. If the data compression method you choose is not among the recipient's preferred methods, it is possible that the recipient will not be able to access the data. You must also select the compression level: Fast, Normal, Good, or Best.
Encode Using ASCII Armor	Determines if SecureTransport uses ASCII armor encoding. ASCII armor refers to using binary-to-text encoding for plain text data.
Keep Original As	Determines if the original unencrypted file is saved to a different name, location, or both. You can use a file name expression.

You can also choose to decrypt encrypted files when you download them.



Decryption includes the following options:

Decryption setting	Description
Decrypt PGP File As	Determines if the file should be decrypted and is the decrypted file is saved to a different name, location or both. You can use a file name expression. Note that when using AS2 to receive PGP encrypted files, the MDN verification only works if the original encrypted file is kept.
Require Trusted Signature	Requires that the file contains a trusted signature or the transfer fails.
Require Encryption	Requires that the file is encrypted or the transfer fails.
Keep Original As	Determines if the original encrypted file is saved to a different name, location, or both. You can use a file name expression.

Note Files that were encrypted using ASCII armoring or data compression are automatically decrypted and decompressed when you decrypt the PGP file.

Post-transmission actions

Subscriptions can also be set up to trigger post-transmission actions for either outgoing or incoming files. Post-transmission actions can be used to move, delete, or rename files based on the success or failure of a transfer. Using these options you can prevent files from being overwritten by renaming them, delete failed transfers, or move transferred files to a different directory on the server. You can also delete a file on the remote server after it is transferred. An expression language is provided so you do not need to specify a file name but can use patterns to control the post-transmission actions.

The following post-transmission actions are provided:

Post-transmission setting	Description
Send Options	

Post-transmission setting	Description
On Failure	A failure occurs when the transfer is incomplete and all retry attempts were unsuccessful. Select one of the three choices: **No Action**, **Delete**, or **Move/Rename File To**. • Selecting **No Action** causes the file to stay in the new location with the file name you specified. If another file with the same name is transferred to this location, the original file is overwritten. • Selecting **Delete** removes the file from the new location. • Selecting **Move/Rename File To** requires you to specify a directory in the location where you are transferring the files to and to provide an expression used to rename the file.
On Success	Select one of the three choices: **No Action**, **Delete**, or **Move/Rename File To**. • Selecting **No Action** causes the file to stay in the new location with the file name you specified. If another file with the same name is transferred to this location, the original file is overwritten. • Selecting **Delete** removes the file from the original location. • Selecting **Move/Rename File To** requires you to specify a directory in the location where you are transferring the files to and to provide an expression used to rename the file.
Receive Options	
On Failure	A failure occurs when the transfer is incomplete and all retry attempts were unsuccessful. Select one of the three choices: **No Action**, **Delete**, or **Move/Rename File To**. • Selecting **No Action** causes the file to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten. • Selecting **Delete** removes the file from the original location. • Selecting **Move/Rename File To** requires you to specify a directory in the location where you are transferring the files from and to provide an expression used to rename the file.

Post-transmission setting	Description
On Temporary Failure	A temporary failure can occur when the transfer is incomplete and a retry occurs. Select one of the three choices: **No Action**, **Delete**, or **Move/Rename File To**. • Selecting **No Action** causes the file to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten. • Selecting **Delete** removes the file from the original location. • Selecting **Move/Rename File To** requires you to specify a directory in the location where you are transferring the files from and to provide an expression used to rename the file. On Temporary Failure is only available for server-initiated transfers.
On Success	Select one of these choices: **No Action** or **Move/Rename File To**. • Selecting **No Action** causes the file to stay in the original location. If another file with the same name is transferred to this location, the original file is overwritten. • Selecting **Move/Rename File To** requires you to specify a directory in the location where you are transferring the files from and to provide an expression used to rename the file.

Note To preserve the original file name when using the **Move/Rename File To** option, use the `${stenv.target}` or `${stenv['target']}` expressions.

Specifics:

- On Windows environments, when renaming a file with a post-transmission action, you cannot use the following characters: `* < > ? " / \ | :`. Use URL encoding instead if you need to represent one of these characters for a remote post-transmission action.
- When using SecureTransport on a Windows platform and you configure a post-transmission action in a subscription to move a file from one drive partition to another, no folders are created on the new drive partition and the files are not moved.
- If you are using SecureTransport on a UNIX-based platform, the following characters cannot be used: `/ . \`.
- Paths specified in post-transmission options are treated as either relative to the subscription folder or an absolute path starting from the subscription folder.

◦ Relative paths are resolved against the target location which might not be the subscription folder, but can be any of its subfolders. If you use ".." in the file name expression, the final destination cannot go up the folder tree past the subscription folder.

◦ Absolute paths are calculated as relative to the subscription folder. The final destination here is bound to the subscription folder even when the expression uses ".." one or more times.

Manage subscriptions

Use the *Subscriptions* pane of the *User Account* window to manage subscriptions.

The following sections provide how-to instructions for managing subscriptions:

Subscribe an account to an application

Before creating a subscription for an account, you must create at least one application for the system.

Note PGP decryption and encryption paths, regardless of whether you use a relative or an absolute path type, are relative and restricted to the directory where the file resides.

Prerequisites

- Create an application. For details, see *Manage applications*.
- If the account is to have server-initiated transfers associated with it, you must create at least one transfer site for that account. For details, see *Transfer sites on page 540*.

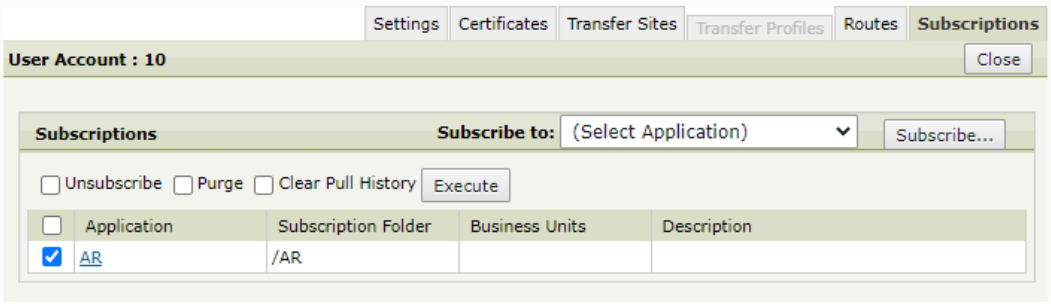
Workflow

Select a user account

1. Select **Accounts > User Accounts**. The *User Accounts* page is displayed.
2. Click the name of the account for which you want to create a subscription.

   The *User Account Settings* page is displayed with details for the selected account.
3. Click the **Subscriptions** tab for the selected account.

4. Select the application you want to subscribe the account to from the **Subscribe to** list.



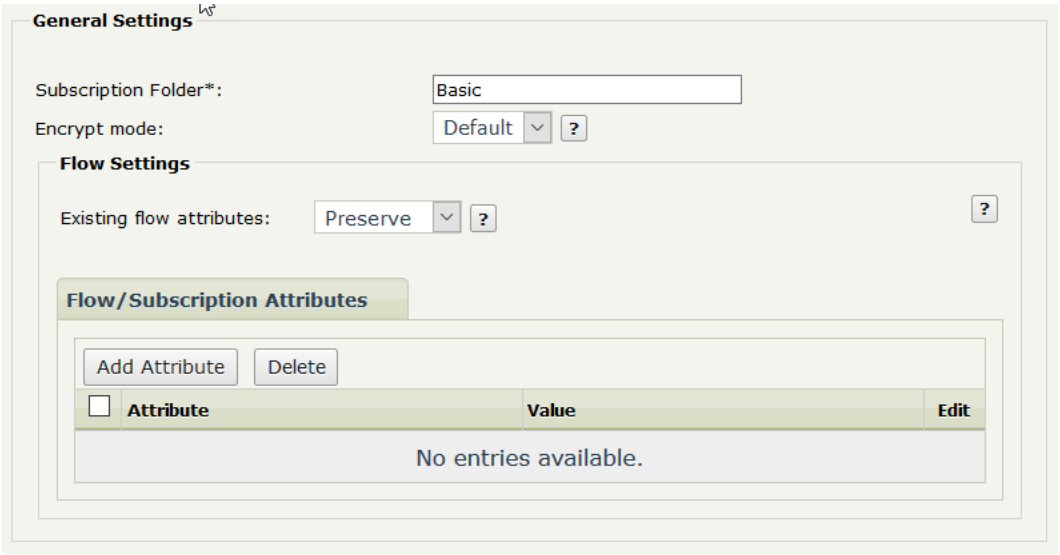
5. Click **Subscribe**.

   The settings page for the subscription is displayed.

   Note Some applications do not include all the fields described here.

Configure general settings

In the *General Settings* pane:



1. In the **Subscription Folder** field, type the full path to the subscription folder or use the default folder name. The subscription folder name can contain 254 characters or less.

   Note Axway does not recommend you to add leading or trailing space intervals to your subscription folder name as you may experience unexpected behavior with different SFTP / FTPS clients (although leading spaces are accepted with certain SFTP clients). You cannot use the following characters in the subscription folder name: `* < > ? " / \ | :`

2. Select the **Encrypt mode**. Selecting the **Encrypt mode** allows you to configure repository encryption for accounts at the per-subscription level. For additional information, refer to *Repository encryption on page 46*.

Select **Default** to inherit the encryption mode for the subscription folder from the account or the global settings.

Select **Enable** to encrypt all files uploaded to the subscription folder.

Select **Disable** to upload unencrypted files to the subscription folder.

You can subscribe multiple accounts to the application. Some accounts may have repository encryption enabled for the subscription folder and others may have it disabled. As a result, the files uploaded from some accounts are encrypted while from other accounts the uploaded files are not encrypted.

For example if:

- **User1** has encryption **enabled** and is subscribed to **Shared Folder Application A**.
- **User2** has encryption **disabled** and is subscribed to **Shared Folder Application B**.
- **User3** is subscribed to both **Shared Folder Application A** and **Shared Folder Application B** and has encryption **enabled** for **Application A** and **disabled** for **Application B**.

SecureTransport applies encryption to files that **User3** uploads and **User1** can download all files because they have repository encryption **enabled** for **Application A**. **User2** can only download unencrypted files because they have repository encryption setting **disabled** for **Application B**. In this example, **User1** and **User3** can download all shared folder files because they have identical settings for repository encryption. **User1** and **User2** files are encrypted or not based on their repository encryption setting.

For example, three users are subscribed to a Shared Folder Application:

- **UserA** - Subscription repository encryption is set to **Enable**
- **UserB** - Subscription repository encryption is set to **Disable**
- **UserC** - Subscription repository encryption is set to **Disable**

Upload actions:

- **UserA** uploads a file to the subscription folder – The file is **encrypted**.
- **UserB** uploads a file to the subscription folder – The file is **unencrypted**.
- **UserC** uploads a file to the subscription folder – The file is **unencrypted**.

Result:

The subscription folder that the three users share contains both **unencrypted** and **encrypted** files.

Download actions:

- **UserA** can download all files from the subscription folder.
- **UserB** and **UserC** can only download unencrypted files from the subscription folder.

3. If you selected the Standard Router application type, Enter an ID in the **Subscriber ID** field.

   When the **Rename submitted files to include Subscriber ID** checkbox is selected during the application definition, the uploaded file is renamed before it is sent to the internal system.

The file is renamed in the format <ID> <FILE_NAME> where <ID> is the Subscriber ID that is specified here for the current Subscription, and <FILE_NAME> is original file name. For details, see *Standard Router application on page 850*.

4. In the *Flow Settings* pane, select the **Existing flow attributes**.

   If **Preserve** is selected, the attributes defined in the *Flow Attributes* pane will be applied only for newly received files which do not have associated flow attributes.

   If **Overwrite** is selected, the attributes defined in the *Flow Attributes* pane will overwrite any existing attributes for incoming files (for example, files published to this folder from another subscription folder).

   When **Append** is selected, only the attributes which are not defined for incoming files will be applied. Existing attributes will be preserved.

5. In the *Flow/Subscription Attributes* pane:

   a. To add an attribute, click **Add Attribute**. For additional information on flow attributes, refer to *Flow and subscription attributes on page 198*.
   Add Attribute enables the administrator to add custom properties (Key=Value). Their values can be set using expression language. Administrators can use flow attributes, session and environment variables that are evaluated at the time a file is transferred through the subscription.
   Examples:

   - `${env.DXAGENT_ACCOUNT_EMAIL}`
   - `${sess.STSESSION_LDAP_DOMAIN_ID}`
   - `${flow.userVars.sampleKey1}` if set in a previous step
   - `${flow.MetaDataKey1}` if set in an external transfer site

   Flow attributes are bound to files; subscription attributes are bound to subscriptions. Note that these attributes are not available during server-initiated incoming transfers, as well as for trigger file transfers.

   Caution With pulls that do not return files, no flow attributes are available as no files are actually transferred. In this case, only subscription attributes are set.

   Caution Expressions used in attribute values cannot evaluate a subscription attribute, i.e., `${subscription....}`. This is because the subscription attribute is bound to the subscription itself and is not transferred with the file to a new subscription.

   Flow attributes can be accessed using the following expression: `${flow.attributes['userVars.ATTRIBUTE_NAME']}`. Note that flow attributes can be used for expression evaluation in Advanced Routing only when the application operates with files.
   Subscription attributes can be accessed using the following expression: `${subscription.attributes['userVars.ATTRIBUTE_NAME']}`.
   Subscription attributes can be used for expression evaluation in all Advanced Routing

fields.
Examples of Attributes:

Attribute	Value
userVars.1	internalEmail@axway.com
userVars.2	ReportsMonitor

To access attributes, see the following examples:

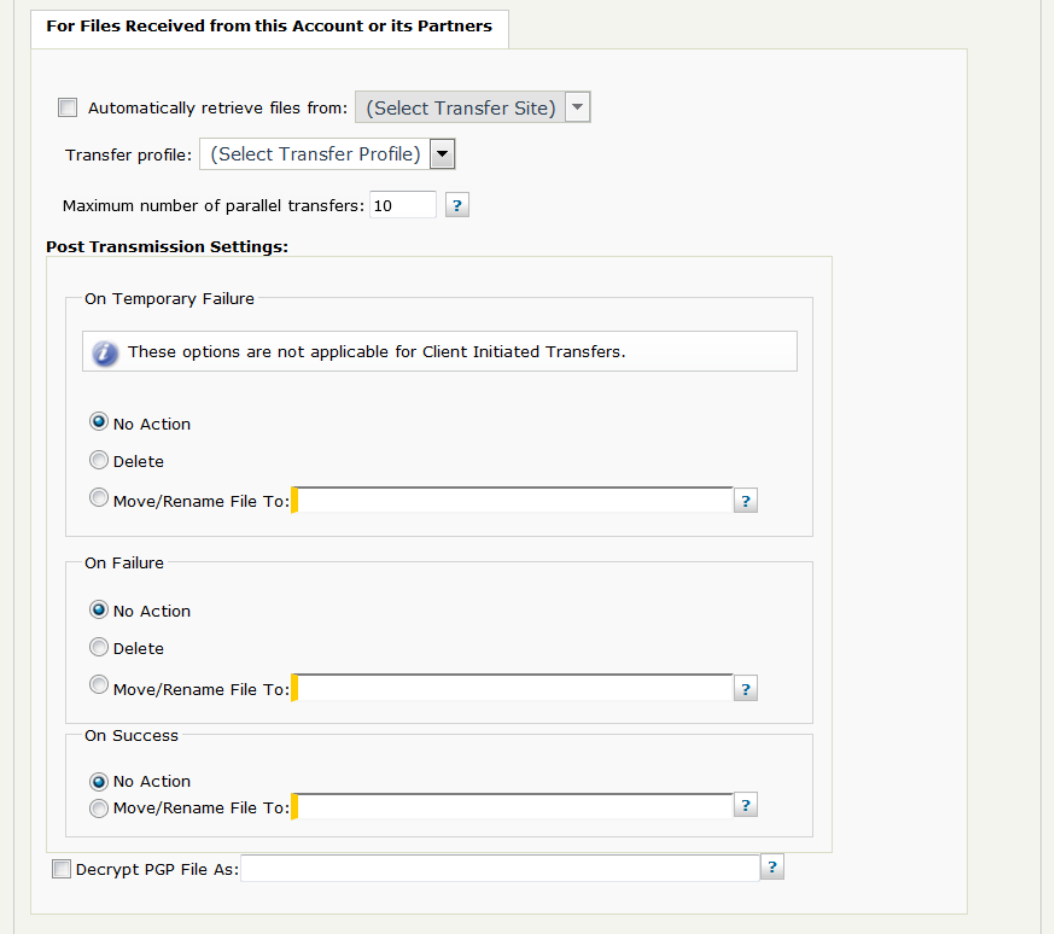
```
${account.attributes['userVars.1']}
${account.attributes['userVars.2']}
```

For example, the `account.attributes` is the selector for attributes of the account used to execute the current route - it has to be written exactly as shown.
The `userVars.` prefix must be prepended to attribute name.
All this should be written as an EL expression: `${...}`

b. Click the **Save** (✓) icon.

Configure files received settings

In the *For Files Received from this Account or its Partners* pane:



1. To set a schedule for automatic retrieval of the transferred files, select the **Automatically Retrieve Files From** checkbox and then select the respective transfer site from the drop down list.
   If you select a PeSIT transfer site, you can choose a **Transfer Profile** from the list or leave the field empty to use the default PeSIT transfer profile. For more information, see *Transfer profiles on page 640*.

   If you select an SSH transfer site, you can configure SecureTransport to keep track of the downloaded files and pull only new or the modified ones.

   A subscription that retrieves files from an AS2 transfer site does not use a schedule. To retrieve files from an AS2 transfer site, see *Considerations for subscriptions and AS2 transfer sites on page 673*.

   For a subscription that retrieves files from a Folder Monitor transfer site, to configure scheduled Folder Monitor operation, you must select **Set explicit FolderMonitor Schedule**.
2. (Optional) Click **Configure** in the *Schedule* pane to set up a future one time event or a recurring schedule.

   The *Configure Schedule* dialog box is displayed.

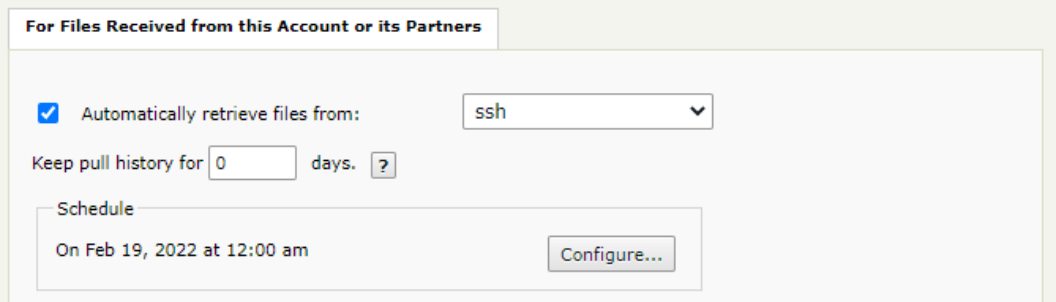
Note If you configure a schedule and save it after the scheduled start time, the task will not be executed. You must save your configured schedule before the scheduled start time.

3. Specify the desired schedule. For details, see *Set up a scheduled transfer task for a subscription on page 675*.
4. (Optional) Set **Maximum number of parallel transfers**.
   If you enter a value greater than zero, SecureTransport executes only the specified number of transfers in parallel. If the value is null or zero, the maximum number of parallel transfers is limited by system capacity.

   The maximum number of parallel transfers limit is applied cluster wide. The limit for files transferred from the client will not be exceeded. Due to limitations in Standard Cluster communication mode, the parallel pulls limit can be exceeded when there are several connections. If you want to force the limit, then the `force.standard.cluster.sit.pulls.sync=true` system property should be added to the `start_tm_console`. Adding the property to the `start_tm_console` has a performance penalty due to increased cluster communication.
5. (Optional) Click the **Retrieve Files Now** button to immediately trigger a one time file pull. For details, see *Retrieve files now* on page 676.
6. (Optional) In the *Post Transmission Settings* pane, set the failure and success options. For details, see *Post-transmission actions on page 661*.
7. To decrypt the transferred files, select **Decrypt PGP File As** and enter a file name or expression.
8. (Optional) Select or clear the **Require Trusted Signature** and **Require Encryption** options as needed for incoming transfers.
9. (Optional) Select **Keep Original As** to save the encrypted file. You can move the file to a different folder, rename the file, or both using either hard-coded text or by entering an expression.

Pull only new and changed files

A subscription that retrieves files from an SSH transfer site can be configured to pull only files that have been added or changed since the last retrieval. When this feature is enabled, SecureTransport keeps a history of the downloaded files with key information like file name, file size, file last modification date, and date of downloading. On each pull attempt, it compares the files available on the remote server with the records of the downloaded files for the past "X" number of days specified in the **Keep pull history** field. The comparison is based on the file name, size, and last modification timestamp. If all of the three attributes match, the files are considered to be identical. Identical files are not re-downloaded. Otherwise, the file is considered unique and will be downloaded.



To enable this functionality, you need to change the default **Keep pull history** value which is 0 to the desired number of days - only the records during this period will be taken into consideration. If the input field is empty, the functionality is disabled. For example, if **Keep pull history** is set to 2, SecureTransport will check if a file with the same name, size, and timestamp has been downloaded during the last two days. If there's a match, the file will not be downloaded again. However, older records are not checked, which means that if the file was downloaded more than 2 days ago, it will be pulled again.

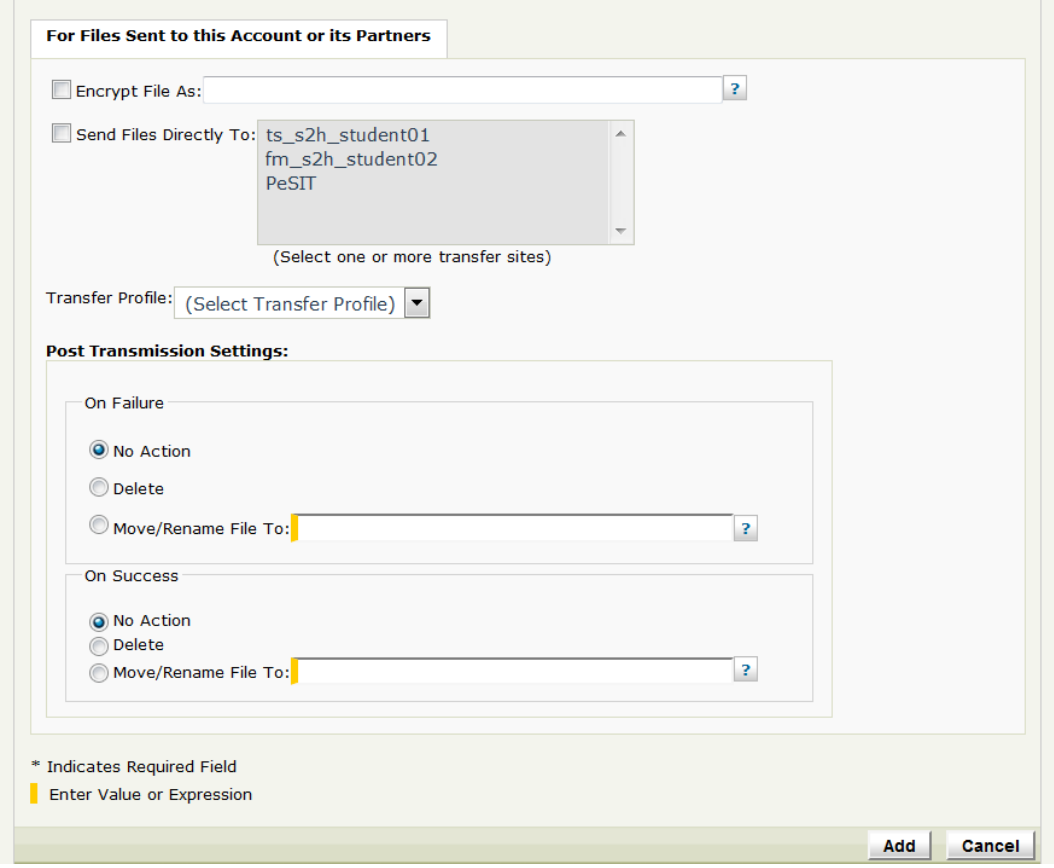
To keep the number of downloads manageable, you need to manually clear the pull history when needed via one of the following ways:

- via the REST API using *POST /subscriptions/{id}/operations*
- via the Administration Tool, from the account's **Subscription** tab. See *Unsubscribe, delete subscription folder, or clear pull history* on page 678.

You can also use the REST API *POST /subscriptions/{id}/operations?operation=pull* request to specify a different transfer site to download files from, making them part of the pull history.

Configure files sent settings

In the *For Files Sent to this Account or its Partners* pane:



1. Select or clear **Encrypt File As** for outgoing transfers. Enter a file name or an expression for the encrypted file.

   If you selected **Encrypt File As**, additional fields display. You must select either **Encrypt Using PGP Key** or **Sign using PGP Key** and select a PGP key.

2. Select or clear **Encrypt Using PGP Key** for outgoing transfers. If you turn this option on, you must select the PGP key used for encryption of outbound transfers from the list.

3. Select or clear **Sign using PGP Key** for outgoing transfers. If you turn this option on, you must select the PGP key used for signing of outbound transfers from the list.

4. (Optional) Select or clear **Use Data Compression** for outgoing transfers. If you turn this option on, you must select a data compression **Type** from the list. You must also select a **Compression Level**.

5. (Optional) Select or clear **Encode Using ASCII Armor** for outgoing transfers.

6. (Optional) Select **Keep Original As** to move the file to a different folder, rename the file, or both using either hard-coded text or by entering an expression.

7. (Optional) To set automatic sending of the files, select **Send Files Directly To** and choose one or more transfer sites from the drop-down list. Press either the Shift or Ctrl key while selecting the transfer sites to choose more than one site. All files in the outbox of the subscription folder are automatically sent to the selected transfer sites. If you select a PeSIT transfer site, you can select a **Transfer Profile** from the list or leave the field empty to use the default PeSIT transfer profile. For more information, see *Transfer profiles on page 640*.

Note Enable the `SendToSite` rule package to upload files without subscribing an account to an application. For more information, see *Manage rule packages* on page 223.

8. (Optional) Under *Post Transmission Settings*, set the failure and success options.

   Note If you configure two or more sites in **Send Files Directly To**, do not configure **Post Transmission Settings**.
   If you select an AS2 transfer site, see *Considerations for subscriptions and AS2 transfer sites on page 673*.

Complete the subscription

To complete the subscription, Click **Add**.

Considerations for subscriptions and AS2 transfer sites

You can set independently the options to enable or disable automatic sending to and receiving files from an AS2 transfer site. For example, for a particular site, you can enable the **Automatically Retrieve Files From** option and disable the **Send Files Directly To** option.

Also, you can specify different AS2 transfer sites for each of the options. For example, you can send files directly to one AS2 transfer site and automatically receive files from a different AS2 transfer site.

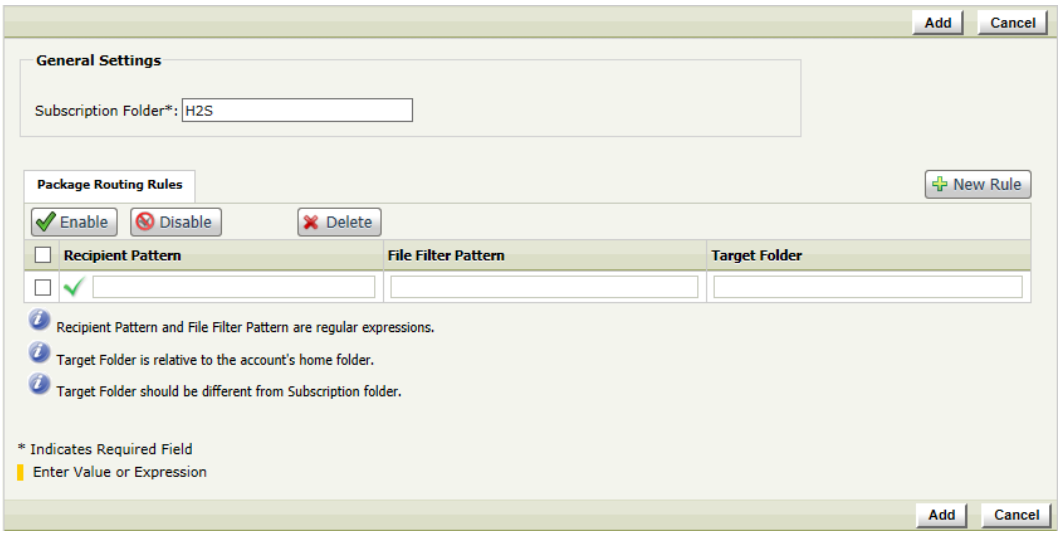
When you specify an AS2 transfer site in the **Automatically Retrieve Files From** list for a subscription, you cannot reuse the AS2 transfer site again in a different subscription.

If you specify an AS2 transfer site in the **Automatically Retrieve Files From** list for a subscription, the **On Temporary Failure** option is not displayed. This setting is not applicable for AS2 incoming transfers as they are never retried.

Human to System type application

Use a subscription to a Human to System type application to specify email addresses that represent destinations for files sent in emails from ST Web Client. When SecureTransport receives an email for one of these addresses, it process the files sent as you specify in the *Package Routing Rules* list in the subscription. SecureTransport applies all the rules that match the email.

Note An account can have at most one subscription to a Human to System type application.



1. In the **Subscription Folder** field, type the path to a folder that the application uses for temporary files or use the default folder name. The path is relative to the account home folder. The subscription folder name can contain 254 characters or less.
2. Create one or more package routing rules.
   a. Click **New Rule**.
   b. In the **Recipient Pattern** field, type a regular expression that matches the email addresses that this rule applies to. For example, `invoices@example\.com`.
   c. In the **File Filter Pattern** field, type a regular expression that matches the names of the files that this rules applies to. For example, `*\.xls`.
   d. In the **Target Folder** field, type the path to the folder that receives the files that arrive at a matching address and with a matching file name. The path is relative to the account home folder.

   Note For information about regular expressions, see *Regular expressions on page 1117*.

   The new rules are enabled by default.
3. Click **Add**.

Scheduled downloads and tasks

The SecureTransport scheduler feature allows you to schedule file downloads and tasks initiated by the server. You can schedule jobs in two ways, either per subscription or per application. You can access the scheduler page by creating or editing any application or subscription with a transfer site that supports scheduled transfers. Only file downloads can be scheduled. REST API endpoints are also available for configuring scheduled tasks for applications and subscriptions.

When you create a scheduled transfer, for example, when creating a subscription connecting an account to an application where you are transferring files from a remote site or an internal system, you can set the following configuration options:

- Start Date and Time
- Perform the task once or perform recurring tasks at configurable regular intervals.
- Do not perform the scheduled task if it falls on a holiday.

Note The scheduler cannot be used for AS2 transfer sites.

Before queuing a new task, the server checks if a previous instance of same periodic task is still pending. If there is an instance of the same periodic scheduled task is pending, the new task is not scheduled.

If the server goes down and then restarts, the scheduler does not execute any scheduled tasks missed during the server down time.

You can set up holiday dates and use them later when creating scheduled transfers or tasks.

Set up a scheduled transfer task for a subscription

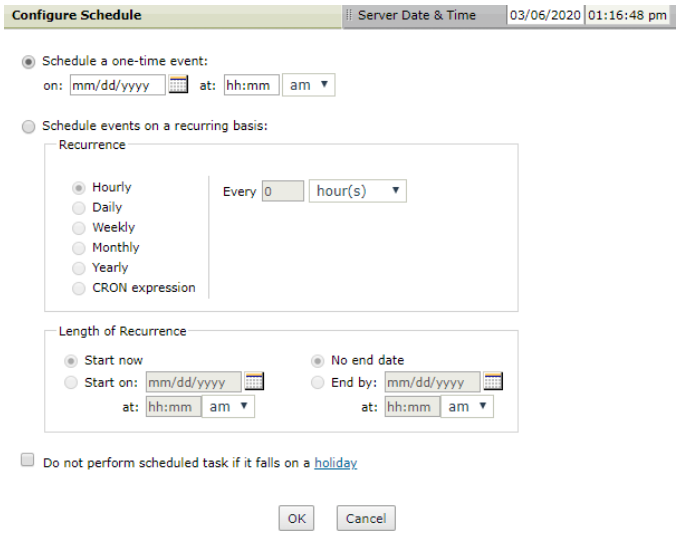
When you subscribe a specific account to an application, depending on the transfer site protocol used you can schedule server-initiated downloads from a particular transfer site.

1. Click **Accounts > User Accounts** and click the account you want to subscribe.
2. Click **Subscriptions > Subscribe to <application_name>**.

   The *Subscription to <application name>* page is displayed.
3. In the *For Files Received from this Account or its Partners* pane, select the **Automatically retrieve files from:** checkbox and choose the transfer site from which the files to be downloaded from the drop down.

   The **Schedule** pane is displayed.
4. To set the schedule conditions, click **Configure.**

   The *Configure Schedule* dialog box is displayed.



5. In the *Configure Schedule* dialog box, specify the desired conditions for the scheduled server-initiated download.

   If the schedule is set on a recurring basis, the **Recurrence** options dynamically change with respect to the recurrence condition: **Hourly**, **Daily**, **Weekly**, **Monthly**, **Yearly**, or **CRON expression**. You can add multiple cron expressions, each on a new line.
   To schedule an immediate recurrent task, select **Schedule events on a recurring basis** and then select **Start now** in the *Length of Recurrence* pane. The task will begin on the next minute.

6. Choose whether the task should be performed if it falls on a day specified as a holiday in the Holiday Schedule. Note that the Holiday Schedule functionality does not allow for executing a scheduled task on the next working day if the specified date happens to be a holiday – when this occurs, the tasks are not executed.

7. Click **OK** when finished setting the schedule.

   Note If the server goes down for some time and restarts, the scheduler does not execute any scheduled tasks missed during the server down time.

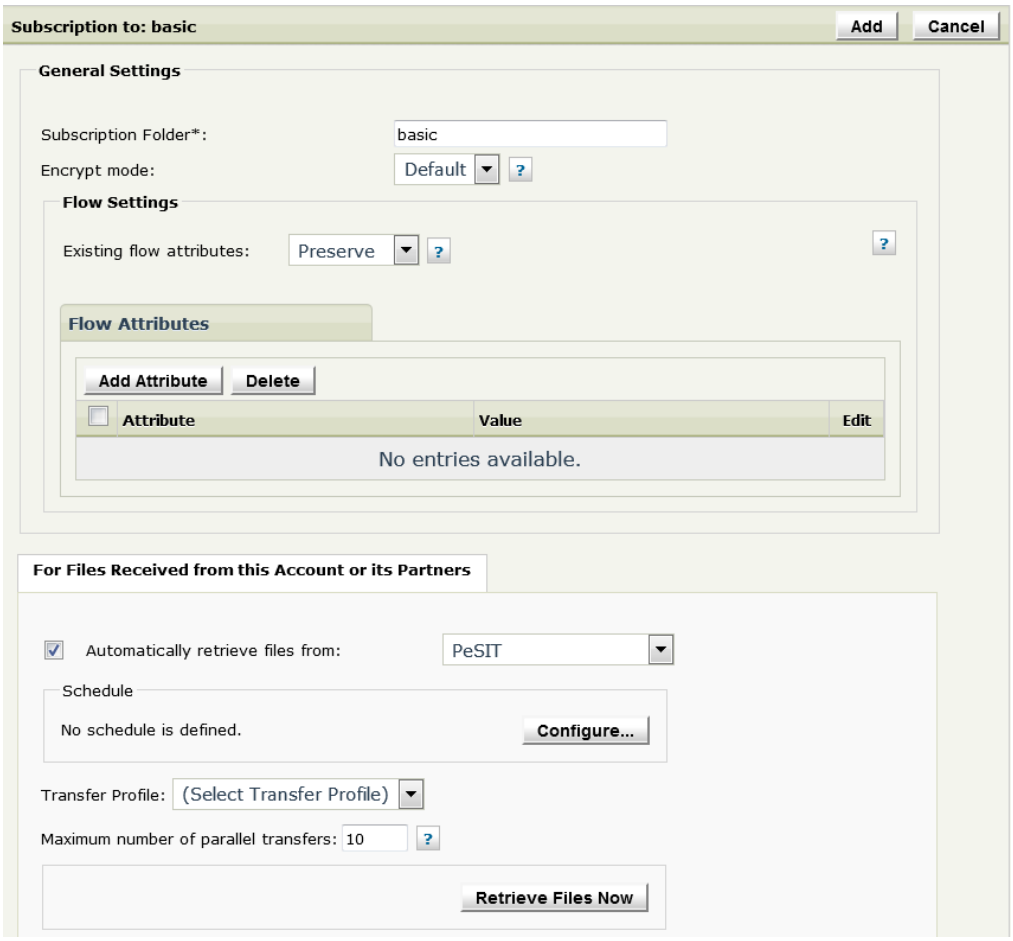
   Note If you configure a schedule and save it after the scheduled start time, the task will not be executed. You must save your configured schedule before the scheduled start time.

Retrieve files now

When you have already subscribed an account to an application, depending on the transfer site protocol used, you can initiate an immediate download from the selected transfer site by clicking the **Retrieve Files Now** button.

1. Click **Accounts > User Accounts** and click the account you want to subscribe.
2. Click **Subscriptions > Subscribe to <application_name>**.

   The *Subscription to <application name>* page is displayed.



3. Select the **Automatically Retrieve Files From** checkbox and then select the respective transfer site from the drop down list. If you select a PeSIT transfer site, you can select a **Transfer Profile** from the list or leave the field empty to use the default PeSIT transfer profile. For more information, see *Transfer profiles on page 640*.

   The **Retrieve Files Now** button is displayed.

4. Click **Retrieve Files Now** to immediately trigger a one-time file pull.

   Note When the **Retrieve Files Now** is clicked, a one-time pull event is always triggered independent of the subscription being saved. If the subscription has just been created and a one-time pull event is executed, the subscription folder will be created by the runtime if it does not exist. Retrieve files now pulls can also be triggered from the REST API by account, transfer site, and destination folder.

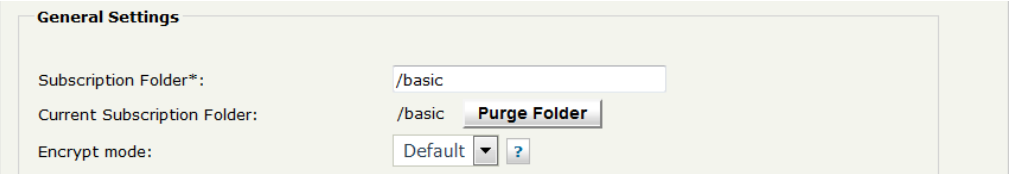
   Note When the one-time pull event is triggered, the admin daemon will try to connect to the Transaction Manager until the maximum number of retry attempts is reached as specified by the `Streaming.Event.maxRetries` server configuration parameter. The period between each retry is specified by the `Streaming.Event.idleTimeout` server configuration parameter. When the maximum number of retries is reached, the execution process

finishes. For more information on server configuration parameters, refer to *View and change server configuration parameters on page 334*

Purge a subscription folder

You can delete the contents of the subscription folder specified for an account.

1. Click **Accounts > User Accounts** and click the account containing the subscription.
2. Click **Subscriptions** and click the subscription you want to edit.
3. Click **Purge Folder** to remove the contents of the current subscription folder.



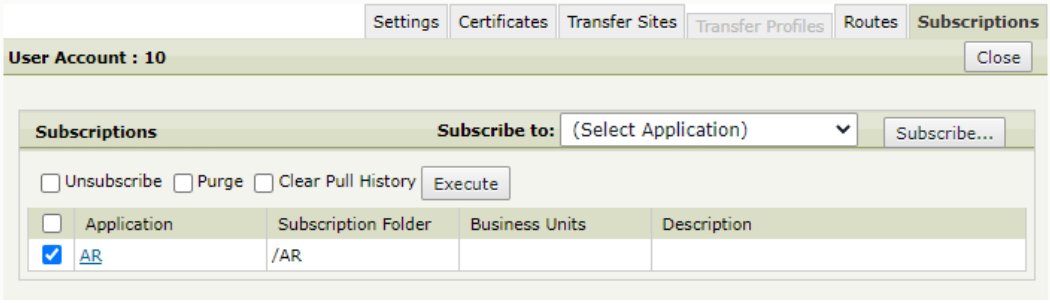
4. A message asking you to confirm the deletion of the folder contents is displayed. Click **OK** to remove the folder contents or click **Cancel** to do nothing.

> **Note** All files and directories (including other subscription directories) residing under the purged folder will be purged and deleted. The purged files and directories cannot be recovered.

Unsubscribe, delete subscription folder, or clear pull history

Use the following procedure to unsubscribe an account from an application.

1. Click **Accounts > User Accounts** and click the account containing the subscription.
2. Click the **Subscriptions** tab and select the checkbox next to the subscription you want to remove.



3. Select the desired action:
   - **Unsubscribe** - removes the subscription. Unsubscribing from an application deletes the associated pull history.
   - **Purge** - deletes the associated subscription folder

- **Unsubscribe**and **Purge** removes the subscription and deletes the associated subscription folder including any subfolders and pull history.
- **Clear Pull history** - deletes the pull history associated with the selected application.

4. Click **Execute**.
5. A message asking you to confirm the action is displayed. Click **OK** to confirm or click **Cancel** to do nothing.

Duplicate an account

You can easily create multiple accounts with similar settings by duplicating an existing account. While most account information will be replicated, certain details must remain unique to each account, such as the account name and home folder. Fields requiring unique information are marked with an asterisk (*).

Certificate Uniqueness

By default, a login certificate must be unique to a user and cannot be used on multiple accounts. This means that it cannot be duplicated when the `CertificateStores.UserCertificateStore.Keystore.uniqueCerts` configuration option is set to `true`. For more information, see *Certificate uniqueness on page 527*. It's important to note that account duplication will fail if either the partner or private certificate uniqueness option is enabled. Before starting the duplication process, make sure that these options are set to false:

- `CertificateStores.AccountLocalCertificateStore.uniqueCerts`
- `CertificateStores.PartnerCertificateStore.uniqueCerts`

Procedure

When you select **Duplicate Account**, you are guided through the different pages where you can alter the user information, the transfer site and subscription details.

1. Open the user or service account you want to use as a template. In the Settings tab, click **Duplicate Account**.

   A *New Service Account* or *New User Account* page is displayed with a **Next** button at the bottom of the page.
2. Change the account name, home folder and any other user information you want to modify, such as the password. Click **Next** to continue.

   If the account you are using as a template has a transfer site set up, the *Add Transfer Site* page is displayed.
3. Add a new transfer site, modify the existing settings, or click **Next** to continue without making any changes.

   If you did not add a transfer site, the *Subscriptions* page is displayed. Continue with step 5.

   If you added a transfer site, the *Transfer Profiles* page is displayed.

4. Add a new transfer profile, modify the existing settings, or click **Next** to continue without making any changes.

   The *Subscriptions* page is displayed.

   Note If you are duplicating a service account using a Standard Router application, only the transfer site and the certificates are copied to the new account.

5. Modify the subscription settings and click **Next** to continue.

   The new account is saved to the server and displayed on the *Settings* tab of the user account. The new account is disabled. Routes assigned to the duplicated account are also copied to the new account. If subscriptions are changed during duplication and they had routes assigned, the copied routes are assigned to the respective changed subscriptions.

Note You will not be able to change the value in the *Route* drop-down menu while duplicating an account. You can change the value afterward via the *Subscriptions* tab in the newly created user account.

Control login name case sensitivity

You can avoid login errors caused by different treatment of the case sensitivity of user names for the different user types by setting server configuration parameters that control converting the case of login names and the case sensitivity of the names of a virtual users.

- **`Users.LoginNames.normalizedCaseInsensitiveUsername`**– Controls the conversion of user names entered during login. Valid values are:
  - `lower` – The user name is converted by mapping all alphabetic character to lower case. This is the default.
  - `upper` – The user name is converted by mapping all alphabetic character to upper case.
  - `none` – The user name is not converted.
- **`Users.LoginNames.virtualUserCaseSensitive`** – Controls whether the user name of a virtual user is case sensitive. Valid values are `true`, the default, or `false`.

The case sensitivity of login names is specified as follows, by user type:

- **Real users** – User names are case-sensitive on UNIX-based systems and case-insensitive on Windows systems.
- **Virtual users** – Case sensitivity depends on the parameter `Users.LoginNames.virtualUserCaseSensitive`. When the value of this parameter is `true`, login names must match configured user names exactly. When the value of this parameter is `false` and the value of `Users.LoginNames.normalizedCaseInsensitiveUsername` is `lower`, the login name is converted and there must be no upper case letters in the configured user name. When the value of this parameter is `false` and the value of `Users.LoginNames.normalizedCaseInsensitiveUsername` is `upper`, the login name is converted and there must be no lower case letters in the configured user name.

When the value of this parameter is `false` and the value of `Users.LoginNames.normalizedCaseInsensitiveUsername` is `none`, login names must match configured user names exactly.

- **LDAP users** – Case sensitivity depends on the **LDAP Common case** setting on the *LDAP Server* page. This setting specifies that an LDAP user name must be converted into upper or lower case before it is submitted for authentication. If the value of **LDAP Common case** is either `Upper` or `Lower`, the LDAP user name authentication is case insensitive. If the value of **LDAP Common case** is `None`, then case sensitivity is assumed. In other words, you must set **LDAP Common case** to either `Lower` or `Upper` to indicate that LDAP performs case insensitive match during login, even if it does not require normalization of the input string.
- **SiteMinder** – Case sensitivity depends on the `Siteminder.UserAttributesMap.commonCaseAttr` parameter on the *Server Configuration* page. Valid values are `disabled`, which causes case sensitive authentication, and `enabled`, which causes case insensitive authentication with the user name converted according to the value of `Users.LoginNames.normalizedCaseInsensitiveUsername`.

Password Reset

The SecureTransport Web Client users can reset their passwords if allowed by the administrator. If users have an email address configured in their SecureTransport account, they can reset their passwords from the *Login* page. If not, they must contact the SecureTransportadministrator to reset the password.

Note This functionality does not apply to LDAP or SSO users.

This functionality is controlled and configured by the following server configuration options, available on SecureTransport edge and backend servers in the Server configuration page in the Admin user interface:

PasswordReset.Enabled

Specifies if password reset is enabled. The default value for the setting is true.

PasswordReset.Interval

Specifies the minimum interval (in minutes) between two password reset requests for the same email. The value must be a positive integer. The default value for the setting is 60.

PasswordReset.LinkExpirationInterval

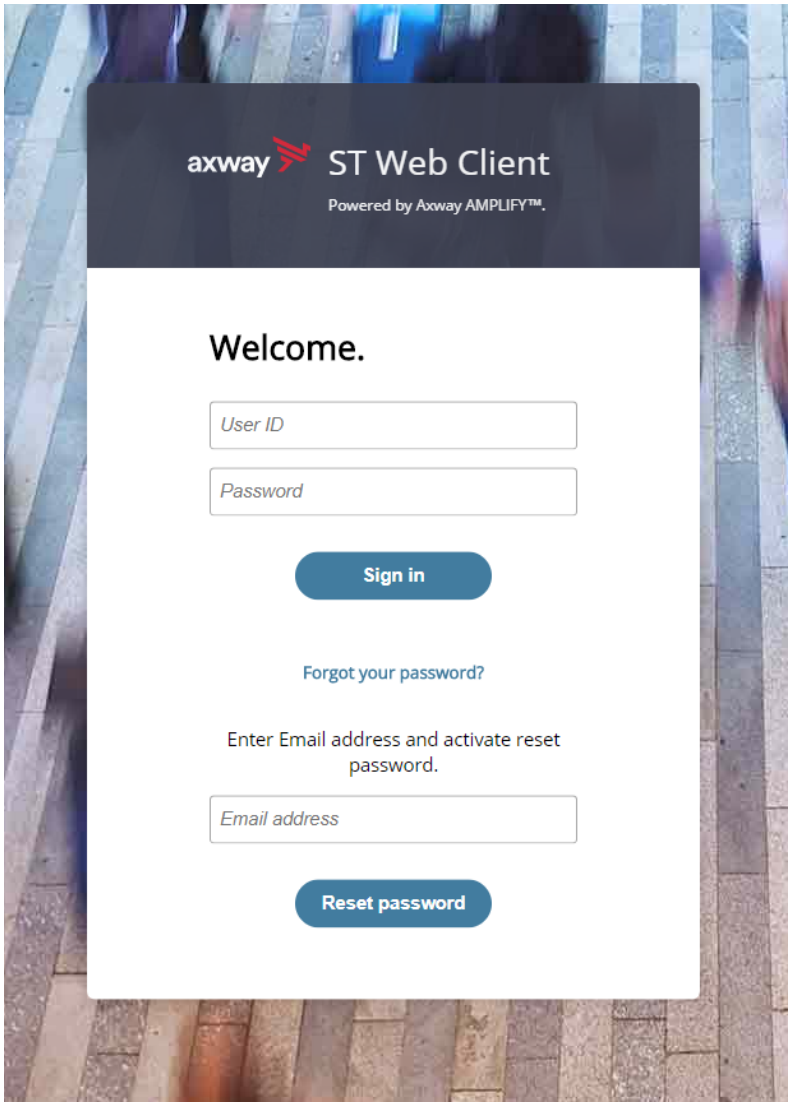
Specifies the time (in minutes) until the reset password link expires. The value must be a positive integer. The default value for the setting is 60.

PasswordReset.RequireUsername

If set to true, users will be asked for a username and an email while requesting a password reset. Otherwise, only email is required. Possible values are true/false. The default value for the setting is false.

Note Set all the password reset configuration options on all SecureTransport edge and backend servers in your setup to have identical values to avoid unexpected behaviour in a streaming setup. (Valid note for all server configuration options: PasswordReset.Enabled, PasswordReset.Interval, PasswordReset.LinkExpirationInterval, PasswordReset.RequireUsername)

To reset a password, an end user must click on the **Forgot your password?** link on the SecureTransport Web Client login page. The following window will be displayed:



A valid and unique email address must be provided. Client password reset will not work if emails assigned to user accounts in SecureTransport are not unique.

SecureTransport sends an email with password reset instructions to the provided email address. Once the user opens the link in the email, they are prompted to enter and confirm the new password in the displayed form. For a list of supported special characters, see Change password.

If the secret question service is enabled, the user must provide an answer to the secret question as part of the password reset process.

The user must fill in all the fields, save the new password, and then log in.

Token

A token will be generated for authentication during the whole password reset process.

The token is encrypted with the SecureTransport Secret and bears its creation-time stamp and the account's email.

A token is expired if time, indicated by the `PasswordReset.LinkExpirationInterval` configuration option, has passed or if a password reset has occurred after the creation of the token.

The link sent via email will have the following structure: `https://<st_ip>/passwordReset?token=<encryptedString>`.

If the secret question feature is enabled in SecureTransport, the users must answer a secret question, which they have preciously set, before they can reset their password.

For more information on Secret Question Functionality, see *Configure a secret question* on page 683.

Configure a secret question

Enabling and configuring the optional secret question feature provides a secure challenge and response mechanism for resetting passwords. It also eliminates the security risks of replacing passwords with temporary ones and sending passwords via email.

If the secret question feature is enabled and their system administrator requires them to do so, end users must select and answer a secret question during their initial login. If the secret question feature is enabled and they are not required to select a secret question, end users may optionally select and answer a secret question. For additional information, refer to the *ST Web Client User Guide*.

As a system administrator, you can:

- *Enable or disable the secret question feature* on page 684
- *Set minimum length for the Secret question answer* on page 684
- *Set maximum number of answer attempts* on page 684
- *Configure a list of secret questions* on page 685
- *Require a user to select a new secret question* on page 685

If an end user forgets their password, they must:

1. Submit a password reset request through their email using the ST Web Client forgotten password mechanism.

2. Click the reset password link in the forgotten password email.
3. Answer the secret question correctly.
4. Provide and verify their new password.

For additional information on the end user password reset process, refer to the *ST Web Client User Guide*.

Enable or disable the secret question feature

By default the secret question feature is disabled. To enable the secret question feature:

1. Navigate to **Operations > Server Configuration**.

   The *Server Configuration* page is displayed.
2. Search for the `Users.SecretQuestion.Enabled` configuration parameter.
3. Click the **Edit** ( ) icon in the *Edit* column.
4. Change the `Users.SecretQuestion.Enabled` configuration parameter to **true**.
5. Click the **Save** ( ) icon in the *Edit* column.

To disable the secret question feature:

Repeat steps 1 through 5, but set the configuration parameter to **false**.

For more information on changing server configuration parameters, refer to *View and change server configuration parameters* on page 334.

Set minimum length for the Secret question answer

To configure a minimum length for the Secret question answer:

1. Navigate to **Operations > Server Configuration**.

   The *Server Configuration* page is displayed.
2. Search for the `Users.SecretAnswer.MinLength` configuration parameter.
3. Click the **Edit** ( ) icon in the *Edit* column.
4. Enter the desired minimum number of characters in the *Value* field. The default value is **0**.
5. Click the **Save** ( ) icon in the *Edit* column.

Note The `Users.SecretAnswer.MinLength` configuration option is available only on SecureTransport Server.

Set maximum number of answer attempts

To configure the maximum number of answer attempts:

1. Navigate to **Operations > Server Configuration**.

   The *Server Configuration* page is displayed.

2. Search for the `Users.SecretQuestion.MaxAttempts` configuration parameter.
3. Click the **Edit** ( ) icon in the *Edit* column.
4. Enter the desired number of answer attempts in the *Value* field. The default value is **0**.
5. Click the **Save** ( ) icon in the *Edit* column.

For more information on changing server configuration parameters, refer to *View and change server configuration parameters* on page 334.

Configure a list of secret questions

To configure a list of secret questions:

1. Navigate to **Operations > Server Configuration**.

   The *Server Configuration* page is displayed.
2. Search for the `Users.SecretQuestions` configuration parameter.
3. Click the **Edit** ( ) icon in the *Edit* column.
4. Update the `Users.SecretQuestions` list as desired. The secret questions must be separated by a hard return. The default secret questions are:
   - **What make was your first car or bike?**
   - **What is your father's middle name?**
   - **What is your mother's maiden name?**
   - **What is your school's mascot?**
   - **What is the name of your favorite fictional character?**
   - **What is your favorite teacher's name?**
   - **Where did you go on your first date?**
   - **What is your dog's name?**
   - **What is your dream occupation?**
5. Click the **Save** ( ) icon in the *Edit* column.

For more information on changing server configuration parameters, refer to *View and change server configuration parameters* on page 334.

Require a user to select a new secret question

To require a user to select and answer a new secret question, edit their user account and select **Require user to set new secret question on next login**. On the next login, the user must select and answer a new secret question before they can access the user interface. For additional information on editing user accounts, refer to *Edit user account settings* on page 515.

Advanced account administration 12

In addition to creating user and service accounts, you can control account access through additional tools such as account export and import, account templates, delegated administration, business units, and administrative roles.

The following topics describe advanced account administration:

- *Export and import accounts* on page 686 - Describes account export and import.
- *Manage administrator accounts* on page 700 - Describes managing administrator accounts.
- *Delegated administration* on page 707 - Describes delegated account administration.
- *Administrative roles* on page 711 - Describes administrative roles.
- *Account templates* on page 717 - Describes account templates.
- *Site templates* on page 737 - Describes site templates.
- *System users* on page 742 - Describes system users.
- *Business units* on page 746 - Describes business units.
- *Display active users* on page 758 - Provides how-to instructions for displaying active users.
- *Client-initiated and server-initiated transfers* on page 761 - Describes client-initiated and server-initiated transfers.

Export and import accounts

SecureTransport 5.5 supports import from the following releases only, patched to the latest patch or service pack – 5.4, 5.3.6, 5.3.5, 5.3.3, 5.3.1, 5.3.0, and 5.2.1. Accounts can be imported from export files produced by the same or another SecureTransport deployment.

Overview

SecureTransport provides a way to export or import all account information, such as account templates, user accounts, service accounts, business units, administrators, and site templates.

Exported account template, user account and service account information includes: user settings, transfer sites, transfer profiles, certificates, certificate requests, subscriptions, applications, business units, route packages, route package templates and their adjacent routes and steps, and those certificates that apply to all of the system.

Accounts can be exported to use as a template, to create a backup, to move from a test to a production environment, or to move from one platform to another.

Tools and access

You can use the command line interface or the Administration Tool to export and import the account information. The following access is required:

- for import/export via the Administration Tool, you must be a master administrator or a delegated administrator with the appropriate privileges. For more information, see *Delegated administration on page 707*.
- for import/export via command line interface, you must have access the server.

For detailed instructions on exporting and importing accounts, see *Export and import accounts: step-by-step instructions on page 691*.

Information is exported to and imported from an XML file. This file can be edited and re-imported. Sensitive information such as private keys and passwords are encrypted during the export process, and you are asked to create a password to protect the sensitive information. When you import the account information, you are asked for the password to allow the sensitive information to be decrypted.

Account XML schema

SecureTransport provides an XML schema for importing and exporting account information. You can use this schema when creating an XML file that can be read by SecureTransport. You can also export an account and use the exported XML file as a template. The schema is located in the `<FILEDRIVEHOME>/conf/xmlExport.xsd` file. For instructions on how to modify it, see *Edit an XML file on page 687*.

Edit an XML file

Read the following information before editing an XML file:

- To change the password for the account, delete the `encryptedPassphrase` element and replace it with a `passphrase` element. Type the new account password using plain text.

  ```
  <passphrase>user3</passphrase>
  ```

  The password is encrypted during the import process.
- You can add or modify the information for a transfer site in its `site` element. Each setting that applies to all transfer sites has an element named for the setting. The information specific to the transfer protocol is represented by the `customProperties` element using the format `<entry key="`*`fieldname`*`">`*`value`*`</entry>` where *`fieldname`* is the name of the field in the transfer site, such as `port`, and *`value`* is the information entered for that field, such as `801`.

- You can add or modify the information for a transfer profile in its `Idf` element:

Field	Element	Valid values
Transfer Profile Name	name	Any valid string
Files To Send	sendMapping	Any valid string
Receive File As	receiveMapping	Any valid string
Acknowledge transfer	sendingAcknowledgmentEnabled	`false` `true`

File Label	fileLabelOption	`DONT_SEND`
		`SEND_FILENAME`
		`SEND_FILENAME_AND_PATH`

All files	multiSelect	`false` `true`

Transfer Mode	transferMode	`ASCII`
		`BINARY`
		`EBCDIC`

Record Format	recordFormat	`0` for Fixed `128` for Variable
Record Length	recordLength	Any valid positive integer

- To indicate that a transfer profile is the default, include the `<default>true</default>` element in the `Idf` element. Only one transfer profile can include this element.
- If you add or modify a subscription, make sure that the application is set up on the server where you are importing the XML file or that you are importing the appropriate application information in the same XML file.
- To modify an application name in a subscription, edit the following element:

  `<applicationReference>MySub</applicationReference>`
- You can change the account information of an existing account, or you can add new accounts to the file.
- When editing an account, you can modify the account information and use the existing `id` attribute in the `Account` element.
- When adding an account, include the following elements in a new `completeAccount` element. Do not include the `id` attribute.

```
<account authByEmail="false" unlicensed="false"
    isUnlicensedAllowedToReply="true" disabled="false" >
    <name>partner1</name>
    <type>user</type>
    <usrid>1001</usrid>
    <grpid>1003</grpid>
    <homeFolder>/home/users/partner1</homeFolder>
    <homeFolderAccessLevel>PUBLIC</homeFolderAccessLevel>
    <email>partner1@example.com</email>
    <phone>800-555-0199</phone>
    <htmlTemplateFolderPath>/html/skin/ric</htmlTemplate
FolderPath>
    <notes>Include ad hoc file transfer functions.</notes>
    <deliveryMethod>CUSTOM</deliveryMethod>
    <enrollmentTypes>CHALLENGED_LINK</enrollmentTypes>
    <implicitEnrollmentType>EXISTING_ACCOUNT</implicitEnrollmentType>
    <customAttributes>
        <customProperties>
            <entry key="encryptMode">unspecified</entry>
            <entry key="routingMode">reject</entry>
            <entry key="transferType">E</entry>
            <entry key="transfersWebServiceAllowed">false</entry>
        </customProperties>
        <localCertificates>
        </localCertificates>
        <partnerCertificates>
        </partnerCertificates>
        <userCertificates>
        </userCertificates>
    </customAttributes>
</account>
```

The elements correspond to the fields in the account *Settings* pane:

Field	Element	Valid values
Attributes		
Allow this account to login by email	authByEmail	`false` `true`

(none)	unlicensed	Is this an unlicensed user account?
		`false`
		`true`

Field	Element	Valid values

Allow reply to packages	isUnlicensedAllowedToReply	`false`
		`true`
		Always `true` of licensed accounts.

(none)	disabled	Is this user account disabled?
		`false`
		`true`

Elements		

Delivery Method	deliveryMethod	`DISABLED`
		`DEFAULT`
		`ANONYMOUS`
		`ACCOUNT_WITHOUT_ENROLLEMENT`
		`ACCOUNT_WITH_ENROLLMENT`
		`CUSTOM`

Enrollment Types	enrollmentTypes	If deliveryMethod is CUSTOM:
		`ANONYMOUS_LINK`
		`CHALLENGE_LINK`
		`EXISTING_ACCOUNT`
		`ENROLL_UNLICENSED`
		`ENROLL_LICENSED`

Implicit Enrollment Type	implicitEnrollmentType	One of the valid enrollment types. Do not include when the deliveryMethod is `DEFAULT` or the field value is None.

Home Folder Access	homeFolderAccessLevel	`PRIVATE`
		`PUBLIC`
		`BUSINESSUNIT`

Custom properties		
Encrypt Mode	encryptMode	`unspecified` `enabled`

Field	Element	Valid values

PeSIT Routing Mode	routingMode	`accept`
		`reject`
		`ignore`

Account Type	transferType	`E`
		`I`
		`N`

Transfer Mode	transfersWebServiceAllowed	`false` `true`

Export and import accounts: step-by-step instructions

You can export and import accounts using the Administration Tool or a command-line utility. This document contains detailed instructions for both methods and explains how SecureTransport behaves when importing existing contacts.

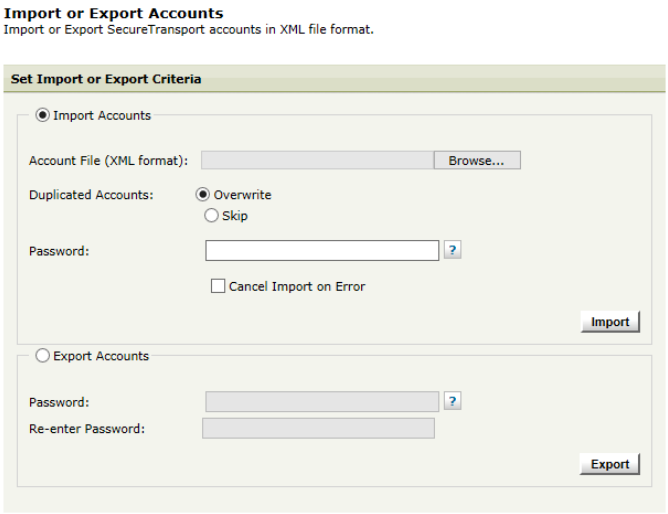
Export accounts using the Administration Tool

You can export accounts using the SecureTransport Administration Tool. When you use the *Import or Export Accounts* page, all the account information on the server is exported. This includes user accounts, service accounts, account templates, certificates, application instances, business units, administrators, administrative roles, site templates, route packages, and route package templates. To export a single account from the Administration Tool, see *Export a single user or service account on page 519*. To control which account information is exported, see *Export accounts from the command line on page 692*.

The exported file is written to the `<FILEDRIVEHOME>/var/tmp/export_accounts.xml` file. This file is overwritten every time you export account information.

1. Select **Accounts > Import/Export**.

   The *Import or Export Accounts* page is displayed.



2. Select **Export Accounts**.
3. Enter a password in the **Password** field, then type the same password in the **Re-enter Password** field. The password must contain at least eight characters.
4. Click **Export**. SecureTransport creates an export file in `<FILEDRIVEHOME>/var/tmp/export_accounts.xml` and displays a message indicating that the export was successful.
5. To save the exported account information to a new location, click **Download Exported Accounts**. A dialog box displays prompting you to **Save** or **Open** the XML file.

   Note You can download the exported file multiple times to the same or a new location. The Export Complete message with the **Download Exported Accounts** button remains, enabling you to download again, until you change tabs, select an option in the navigation bar at the left, or click **Back** twice.

6. (Optional) To refresh the *Import or Export Accounts* page, select **Accounts > Import/Export** or click **Back** twice.

Export accounts from the command line

Using the command line, you can export a single account or all the accounts. Exported account information includes: user accounts, administrators, administrative roles, transfer sites, site templates, transfer profiles, partner certificates, certificate requests, applications, subscriptions, and routes.

Run the `xml_export` utility in the `<FILEDRIVEHOME>/bin` directory from the command line to export account information into an XML file.

The form of the command is:

`./xml_export [`*`options`*`] [`*`fileNameAndPath`*`]` on a UNIX-based system

or

`xml_export [`*`options`*`] [`*`fileNameAndPath`*`]` on a Windows system

where *options* can be:

- -help means display the usage information and exit.
- -acc=*accountName* where *accountName* is the name of the account template, user or service account to export.
- -adm=*adminName* where *adminName* is the name of the administrator account to export.
- -role=*adminRoleName* where *adminRoleName* is the name of administrative role to export.
- -bu=*buName* where *buName* is the name of the business unit to export.
- -st=*stName* where *stName* is the name of the site template to export.
- -crt=*certName* where *certName* is the name of the certificate to export.
- -app=*appname* where *appName* is the name of the application to export.
- -pwd=*passwordFile* where *passwordFile* is a file containing the password used to encrypt or decrypt sensitive information stored in the XML file. The text in this file is not encrypted. You can create the file using any text editor. If you do not use this option, the utility prompts you to type the password in the command window.
- -route=*routeName* where routeName is the name of the route package template or global route package or orphan simple route to export.

and *fileNameAndPath* is the name and location of the XML file to write containing the exported data, such as /user/XMLExport/ST_Acct_Export.XML. If *fileNameAndPath* is not specified, xml_export writes the XML output to the standard output.

To export specific information, use only the option for the information you want. To export all items of one type, set the option to *, such as -acc="*". To export all accounts, administrator accounts (including delegated administrators), administrative roles, business units, site templates, applications, and global certificates, do not use any of the -acc, -adm, -role, -bu, -st, -crt, or -app options.

Exporting a single account using the command line

When you export a single account the following items are not exported:

- Applications
- Business units
- Global certificates
- Site templates
- Route package templates

It is best to use the single account export to create an XML template for account import only.

1. On a UNIX-based system, change to the <FILEDRIVEHOME>/bin directory.
2. Type the following command:

   ./xml_export -acc=*accountName fileNameAndPath* on a UNIX-based system

or

`xml_export -acc=accountName fileNameAndPath` on a Windows system

where *`accountName`* is the name of the account template, user account, or service account you want to export and *`fileNameAndPath`* is the XML file name and location where you want to store the exported data.

3. When prompted, type a password for the exported information. This password is requested when you import the account from the file.
4. Confirm the password by typing it again when prompted.

   The XML file is created in the specified location.

Export information using a specific option

You can use the command line to export only the information for a specific option such as global certificates or application instances.

Export global certificates

1. On a UNIX-based system, change to the `<FILEDRIVEHOME>/bin` directory.
2. Type the following command:

   `./xml_export -crt=certName FileNameAndPath` on a UNIX-based system

   or

   `xml_export -crt=certName FileNameAndPath` on a Windows system

   where *`certName`* is the name of the global certificate you want to export or * to export all global certificates and *`FileNameAndPath`* is the XML file name and location where you want to store the exported data. Multiple certificates may use the same name.
3. When prompted, type a password for the exported information. This password is requested when you import the global certificates from the file.
4. Confirm the password by typing it again when prompted.

   The XML file is created in the specified location.

Export an application

1. On a UNIX-based system, change to the `<FILEDRIVEHOME>/bin` directory.
2. Type the following command:

   `./xml_export -app=appName FileNameAndPath` on a UNIX-based system

   or

   `xml_export -app=appName FileNameAndPath` on a Windows system

where *appName* is the name of the application you want to export or * to export all the application instances and *FileNameAndPath* is the XML file name and location where you want to store the exported data.

3. When prompted, type a password for the exported information. This password is requested when you import the application from the file.
4. Confirm the password by typing it again when prompted.

   The XML file is created in the specified location.

Export a business unit

1. On a UNIX-based system, change to the `<FILEDRIVEHOME>/bin` directory.
2. Type the following command:

   `./xml_export -bu=`*`buName FileNameAndPath`* on a UNIX-based system

   or

   `xml_export -bu=`*`buName FileNameAndPath`* on a Windows system

   where *buName* is name of the business unit you want to export or * to export all the business units and *FileNameAndPath* is the XML file name and location where you want to store the exported data.
3. When prompted, type a password for the exported information. This password is requested when you import the business unit from the file.
4. Confirm the password by typing it again when prompted.

   The XML file is created in the specified location.

Export a site template

1. On a UNIX-based system, change to the `<FILEDRIVEHOME>/bin` directory.
2. Type the following command:

   `./xml_export -st=`*`stName FileNameAndPath`* on a UNIX-based system

   or

   `xml_export -st=`*`stName FileNameAndPath`* on a Windows system

   where *stName* is name of the site template you want to export or * to export all the site templates and *FileNameAndPath* is the XML file name and location where you want to store the exported data.
3. When prompted, type a password for the exported information. This password is requested when you import the site template from the file.
4. Confirm the password by typing it again when prompted.

   The XML file is created in the specified location.

Export a route package template, global route package, or orphan simple route

1. On a UNIX-based system, change to the `<FILEDRIVEHOME>/bin` directory.
2. Type the following command:

   `./xml_export -route=`*`routeName FileNameAndPath`* on a UNIX-based system

   or

   `xml_export -route=`*`routeName FileNameAndPath`* on a Windows system

   where `routeName` is the name of the route package template, global route package or orphan simple route you want to export or * to export all the route package instances and `FileNameAndPath` is the XML file name and location where you want to store the exported data.
3. When prompted, type a password for the exported information. This password is requested when you import the route from the file.
4. Confirm the password by typing it again when prompted.

   The XML file is created in the specified location.

Export all the information using the command line

You can generate an XML file that contains all the account, global and account certificate, application, business unit, delegated administrator, and site template information to use as a backup or to move information from a test environment to a production environment.

1. On a UNIX-based system, change to the `<FILEDRIVEHOME>/bin` directory.
2. Type the following command:

   `./xml_export `*`FileNameAndPath`* on a UNIX-based system

   or

   `xml_export `*`FileNameAndPath`* on a Windows system

   where *`FileNameAndPath`* is the XML file name and location where you want to store the exported data.
3. When prompted, type a password for the exported information. This password is requested when you import the information from the file.
4. Confirm the password by typing it again when prompted.

   The XML file is created in the specified location.

Note To save and restore delegated administrator accounts correctly, you must export and import both server configuration and accounts.

Import accounts using the Administration Tool

SecureTransport 5.5 supports import from the following releases only, patched to the latest patch or service pack – 5.4, 5.3.6, 5.3.5, 5.3.3, 5.3.1, 5.3.0, and 5.2.1. Accounts can be imported from export files produced by the same or another SecureTransport deployment.

You can import account information using the SecureTransport Administration Tool. The account information is imported as an XML file containing user settings, transfer sites, transfer profiles, account certificates, certificate requests, subscriptions, and route packages. You can also import business units, administrators, administrative roles, site template settings, applications, route package templates, and certificates. You must know the password assigned to the file when it was exported by SecureTransport. If you are creating an XML file from scratch, you must assign a password to the file.

Note On SecureTransport installations with an embedded database, the import of accounts containing multiple objects may overload the Audit Log. Before you start importing, be sure to disable audit logging by changing the `AuditLog.Enabled.Import` configuration option value to `false`. After the account import completes, change it back to `true`.

Use the following procedure to import accounts using the Administration tool:

1. Select **Accounts > Import/Export**.

   The *Import or Export Accounts* page is displayed.

2. Select **Import Accounts**.

3. Type the name of the XML file you are importing in the **File Account** field or click **Browse** and locate the file in your system.

   The system automatically validates the XML schema. If the schema is invalid, a warning message is displayed. For more information, see *Account XML schema on page 687*.

   If the XML document is valid, the import process starts.

4. In **Duplicated Accounts**, select **Overwrite** to overwrite the previous account settings or **Skip** to skip such accounts.

   If an account_export.xml, containing route templates instantiated by accounts with route packages, is imported with **Skip** option selected, to target SecureTransport Server, already configured with route templates, accounts and route packages, the corresponding imported route templates, accounts and route packages are rejected during the import if there are already objects with the same names in the target SecureTransport Server.

5. Type the password created for the exported account file in the **Password** field.

6. To stop the import process when an error occurs, select **Cancel Import on Error**.

7. Click **Import**.

   The import process begins. When the import completes, a status message is displayed.

Account import will fail on the first error, and SecureTransport will not attempt to process it further, which means that there may still be more issues. Only one error is logged per import attempt; you can view it in the `<FILEDRIVEHOME>/var/tmp/rejected_import_records.xml` file. This error needs to be fixed before you try to import accounts again.

Import accounts from the command line

SecureTransport 5.5 supports import from the following releases only, patched to the latest patch or service pack – 5.4, 5.3.6, 5.3.5, 5.3.3, 5.3.1, 5.3.0, and 5.2.1. Accounts can be imported from export files produced by the same or another SecureTransport deployment.

You can import accounts to move from one platform to another or to create a large number of accounts without creating them one at a time. Using an exported account as a template, you can create a list of new accounts that can be imported into SecureTransport.

Note On SecureTransport installations with an embedded database, the import of accounts containing multiple objects may overload the Audit Log. Before you start importing, be sure to disable audit logging by changing the `AuditLog.Enabled.Import` configuration option value to `false`. After the account import completes, change it back to `false`.

Run the `xml_import` utility in the `<FILEDRIVEHOME>/bin` directory from the command line to import account information from an XML file. The utility has the following options and parameters:

The form of the command is:

`./xml_import [`*`options`*`]` *`FileNameAndPath`* on a UNIX-based system

or

`xml_import [`*`options`*`]` *`FileNameAndPath`* on a Windows system

where *`options`* can be:

- `-dup=[overwrite|skip]` where `overwrite` overwrites duplicate account entries and `skip` does not import duplicate accounts. If no option is specified, the default setting is `overwrite`.
- `-err=[continue|exit]` where `continue` continues importing the accounts when an error occurs and `exit` stops the utility when an error occurs. If no option is specified, the default setting is `exit`. Errors are written to the `admin.log` file
- `-pwd=`*`passwordFile`* where *`passwordFile`* is a file containing the password used to encrypt or decrypt sensitive information stored in the XML file. You can use this file instead of typing the password from the command line. The text in this file is not encrypted. You can create the file using any text editor.
- `-sync=[y|n]` where `y` synchronizes the imported accounts with all Servers in a Standard Cluster (SC) or Enterprise Cluster (EC) after the import and `n` only imports the accounts to the Server where the command is run. If no option is specified, the default setting is `y`.

  Use `-sync=n` to reduce the time to import large numbers of accounts or accounts with many transfer sites or other features. Then restart the Transaction Manager and the Administration Tool server on the other Servers in the Standard Cluster or Enterprise Cluster to synchronize.

  Note Synchronization requires that the Administration Tool server is running on all Servers in the cluster.

and *FileNameAndPath* is the location and file name of XML file containing the accounts you want to import, such as `/user/XMLExport/ST_Acct_Export.XML`.

Import accounts from an XML file

1. Make sure the Administration Tool server is running on all Servers in the Standard Cluster or Enterprise Cluster.
2. On a UNIX-based system, change to the `<FILEDRIVEHOME>/bin` directory.
3. Type the `xml_import` command.
4. Type the password for the XML file at the prompt.

   The accounts are imported into SecureTransport.

Note To restore administrator accounts exported from SecureTransport 5.1.0 SP3 or SecureTransport 5.2.1 SP3, SP4, or SP5 correctly, you must import server configuration before you import the accounts.

Set the location of the overwrite file

If you set the `dup` option to `overwrite`, and a specific account, certificate, template, or application cannot be imported, the information is written to the following location:

`<FILEDRIVEHOME>/var/tmp/rejected_import_records.xml`

To store the overwrite information in a different directory relative to `<FILEDRIVEHOME>`, set the value of the `Directories.Directory.exportHome.path` parameter on the *Server Configuration* page to the relative path name for the directory. The default value is `/var/tmp`.

To store the overwrite information in a different directory not relative to `<FILEDRIVEHOME>`, clear the value of `Directories.Directory.exportHome.relative` and set the value of `Directories.Directory.exportHome.path` to the absolute path name for the directory.

Import existing accounts and manage file ownership

When importing existing accounts that have a different UID or GID in the XML file, SecureTransport will not automatically overwrite the home folder ownership - all folders and files will remain with the previous UID/GID. This behavior is dictated by the `Import.Filesystem.Ownership` configuration option, and more precisely, its default value: `NONE`. You can edit this option to set a different change ownership mode for the filesystem content when importing existing accounts. The other supported values are:

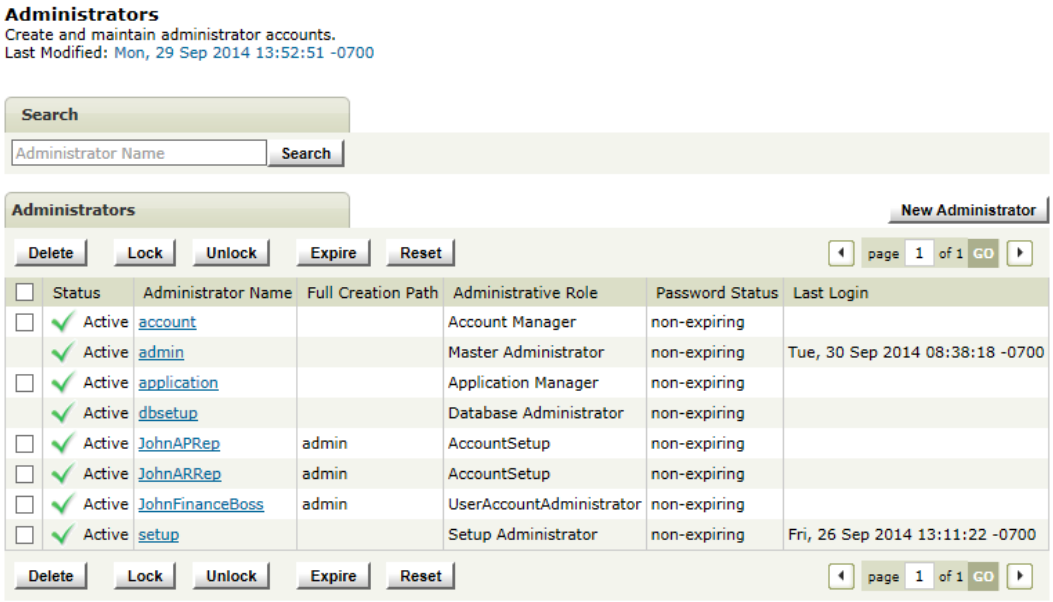
- `RECURSIVE` - changes the ownership of the user home folder and all its files and subfolders to the UID/GID specified in the XML file.
- `NONRECURSIVE` - changes the ownership of the user home folder only to the UID/GID specified in the XML file.

Manage administrator accounts

Use the *Administrators* page to manage administrative accounts. You can create, edit, delete, and lock administrator accounts. You can also expire and reset administrator account passwords. An administrator can also reset their own expired passwords.

If a custom hierarchical administration exists in an organization, you can specify different privileges for each administrator. Use the *Change Password* page to change your password even if you cannot edit accounts.

To display the *Administrators* page, select **Accounts > Administrators** in the Administration Tool. To show only administrators whose names match a character string, type the string in the field in the *Search* pane and click **Search**.



For each administrator, you can view the information described in the following table.

Field	Description
Status	Reports the current status of the administrator account: • **Active** – The account is neither locked nor expired. • **Expired** – Manual action or the expiration interval set the account as Expired. • **Locked** – Manual action or login failures set this account as Locked. The account might also be Expired.
Administrator Name	The name given to the Administrator account

Field	Description
Full Creation Path	Applies only to delegated administrators. Shows the path for the parent administrator. For example, the path might look like: `admin/deladmin1/subdeladmin1`.
Administrative Role	Role assigned to the administrator account. The predefined administrative roles are: • **Master Administrator** – Access to all menus, tabs and pages of the Administration Tool. Cannot be modified. • **Database Administrator** – Access to the *Database Settings* page only. Only the `dbsetup` administrator can have this role. Cannot be modified. • **Setup Administrator** – Access through the custom **Configure** menu to pages required to perform the post-installation tasks and setup operations described in the *SecureTransport Getting Started Guide*. Cannot be modified. • **Account Manager** – Access restricted to the *User Accounts*, *Unlicensed Users Accounts, Service Accounts*, *Templates*, and *Business Units* pages on the **Accounts** menu and all pages on the **Access** menu. • **Application Manager** – Access restricted to the pages on the **Service Accounts** and **Applications** menus. • **Delegated Administrator** – Configurable access to menus and data based on the function the administrator has within an organization.
Password Status	The password status is reported differently depending on whether or not password expiration is enabled on the *Admin Settings* page. If password expiration if enabled, this field shows one of the following: • the expiration date • the date the password expires • a reminder to change the password at the next login • non-expiring If password expiration if not enabled, this field shows one of the following: • the date the password was changed • a reminder to change the password at the next login • no change recorded Depending on your policy for these accounts, you can use the Reset function to mark the last password change time as the current time, or use the Expire function to force a password change at next login.
Last Login	Reports the last login attempt recorded. This is either the date and time of the last successful login, or if the last recorded activity was a failed login, the number of failures and the date and time.

The following topics provide how-to instructions for managing administrator accounts:

- *Add an administrator account* on page 702 - Provides how-to instructions for adding an administrator account.
- *Edit an administrator account* on page 703 - Provides how-to instructions for editing an administrator account.
- *Delete an administrator account* on page 705 - Provides how-to instructions for deleting an administrator account.
- *Lock an administrator account* on page 705 - Provides how-to instructions for locking an administrator account.
- *Unlock an administrator account* on page 705 - Provides how-to instructions for unlocking an administrator account.
- *Expire an administrator account password* on page 705 - Provides how-to instructions for expiring an administrator account password.
- *Reset an expired administrator account password* on page 706 - Provides how-to instructions for resetting an expired administrator account password.
- *Change administrator password* on page 706 - Provides how-to instructions for changing administrator password.

Add an administrator account

If you have access to the *Administrators* page, you can define multiple administrators with varied access privileges for SecureTransport administration. Master administrators and delegated administrators with the **Manage Administrators** privilege have access to the *Administrators* page by default.

1. Select **Accounts > Administrators**.

   The *Administrators* page is displayed.

2. Click **New Administrator**.

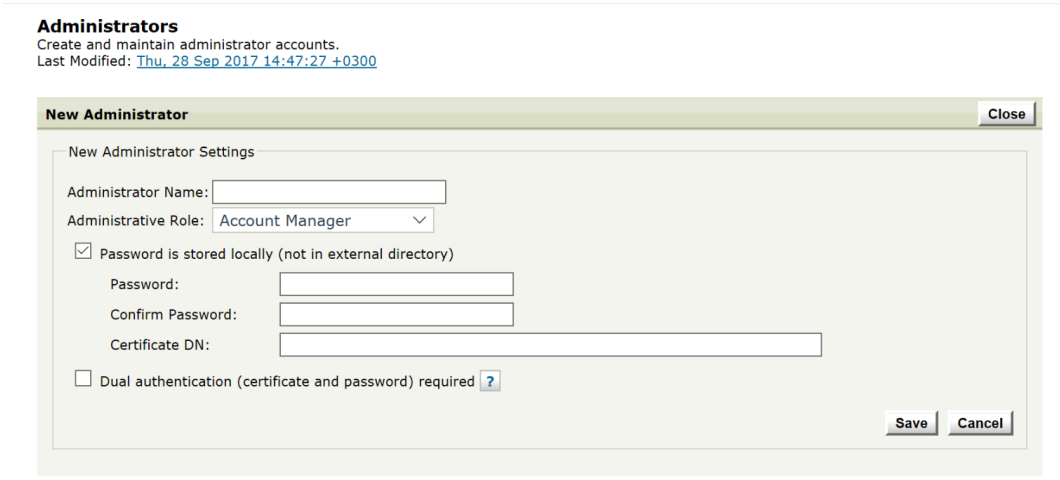
   The *New Administrator* page is displayed.

3. In the **Administrator Name** field, enter a name for the administrator. Administrator names are case-sensitive.
4. Select the **Administrative Role** for the new administrator.
5. Select **Password is stored locally (not in external directory)** to store the administrator password locally and not in an external directory. If this option is selected, completing the **Password** field is mandatory. If this option is not selected, the only mandatory parameters for creating an administrator are **Administrator Name** and **Administrative Role**.

   Note Uncheck this option in order for the current administrator to be able to log in, using an external authentication agent (SSO or authentication plug-in).

6. In the **Password** field, enter a password for the administrator. Passwords are case-sensitive.
7. In the **Confirm Password** field, type the password again to confirm it.

8. If you have enabled administrators to login using a client certificate on the *Admin Settings* page, the **Certificate DN** field and **Dual authentication** checkbox are displayed. If client certificates are required or to specify one for this administrator, complete the fields.
    a. In the **Certificate DN** field, type the Subject field value from the certificate.
    b. To require the administrator to use both a certificate and password, select **Dual authentication**. If you select this option, you must type the Distinguished Name in the **Certificate DN** field.
    c. In case the password stored locally is not checked, the Dual authentication will still be visible, allowing current administrator to be logged using dual-factor authentication by an external
    d. In case the password stored locally is not checked, the Dual authentication will still be visible, allowing current administrator to be logged using dual-factor authentication by an external authentication agent (in this case the plug-in).
    Note: This option will have no effort for SSO-authenticated administrators.



9. Click **Save** to add the administrator account.

    The new administrator is displayed in the list of administrators on the *Administrators* page.

Edit an administrator account

Note If you are using a Firefox browser, disable the auto complete function prior to editing an administrator account settings.

If you have access to the *Administrators* page, you can edit an Administrator account. Master administrators and delegated administrators with the **Manage Administrators** privilege are granted access to the *Administrators* page by default.

Account password stored locally

If the administrator account was created with **Password is stored locally (not in external directory)** selected and the account password is stored locally, use the following instructions to edit administrator account settings.

1. Select **Accounts > Administrators**.

   The *Administrators* page is displayed.

2. Click the administrator entry you want to edit.

   The *Edit Administrator* page is displayed.

3. Make any desired changes in the *Administrator Account Status* pane. You can lock the account, expire the administrator account password, or reset an expired password.

4. Make any desired changes to the fields in the *Edit Administrator Settings* pane.

   You can change the **Administrative Role** and the administrator password. If you have enabled administrators to login using a client certificate on the *Login Settings* page, you can change the **Certificate DN** field and **Dual authentication** checkbox. If the **Administrative Role** is set to Delegated Administrator, you can also modify the **Delegated Administrator Settings**.

5. Click **Save** to apply the changes.

Account password stored externally

If the administrator account was created with **Password is stored locally (not in external directory)** not selected and the account password is stored externally, use the following instructions to edit administrator account settings. This option will allow the administrator to be authenticated by an external authentication agent (Identity provider).

1. Select **Accounts > Administrators**

   The *Administrators* page is displayed.

2. Click the administrator entry you want to edit.

   The *Edit Administrator* page is displayed.

3. Make any desired changes in the *Administrator Account Status* pane. You can only lock and unlock the account.

4. Make any desired changes to the fields in the *Edit Administrator Settings* pane. You can only change the **Administrative Role**
   Note: If you have enabled administrators to log in using a client certificate on the **Login Settings** page, you can change the **Dual authentication** checkbox.

5. Click **Save** to apply the changes.

Delete an administrator account

If you have access to the *Administrators* page, you can remove an existing administrator account. Master administrators and delegated administrators with the **Manage Administrators** privilege are granted access to the *Administrators* page by default.

1. Select **Accounts > Administrators**.

   The *Administrators* page is displayed.
2. Select the checkbox for one or more administrators you want to delete.
3. Click **Delete**.

Lock an administrator account

You can lock an administrator account to remove or modify it or if you are not ready to make it active. Unlock the account when you want the administrator to have access to the server. You cannot lock the account of an administrator who is currently logged in.

1. Select **Accounts > Administrators**.

   The *Administrators* page is displayed.
2. Select the checkbox for one or more administrators you want to lock.
3. Click **Lock**.

   The **Status** column shows that the accounts are locked.

Unlock an administrator account

Use the following procedure to unlock an administrator account.

1. Select **Accounts > Administrators**.

   The *Administrators* page is displayed.
2. Select the checkbox for one or more administrators you want to unlock.
3. Click **Unlock**.

Expire an administrator account password

You can force the immediate expiration of the password for an administrator account. The administrator must set a new password upon the next log in. You cannot expire the password of an administrator who is currently logged in.

If you need to change the password for the administrate account that is currently logged in, edit the account or select **Change Password** from the **Accounts** menu.

1. Select **Accounts > Administrators**.

   The *Administrators* page is displayed.

2. Select the checkbox for one or more administrators for which you want to expire the passwords.
3. Click **Expire**, then click **Save** to apply the change.

Note When the password for the edited administrator is stored externally you can not expire their password.

Reset an expired administrator account password

You can cancel the password expiration and restore the current administrator account password. You can use Reset to cancel an expired administrator password until the administrator changes the password.

1. Select **Accounts > Administrators**.

   The *Administrators* page is displayed.

2. Select the checkbox for one or more administrators for which you want to reset the passwords.
3. Click **Reset**, then click **Save** to apply the change.

Note When the password for the edited administrator is stored externally, you cannot expire their password and therefore you cannot reset it.

An administrator can reset their own expired password using either the Administration Tool or the REST API 2.0 resource *PATCH /myself*.

Change administrator password

The *Change Password* page will not be displayed when authentication for the administrator is completed by an external agent.

Note If you are using a Firefox browser, disable the auto complete function prior to editing an administrator account password.

You can change your administrator password even if you cannot edit your administrator account by using the *Change Password* page. Use this page to view the last date and time the password was changed and to enter a new password.

1. Select **Accounts > Change Password**.

   The *Change Password* page is displayed.

2. Type the new password you want to use in the **Password** field.
3. Retype the new password in the **Confirm Password** field.
4. Click **Save**.

Delegated administration

SecureTransport provides a customizable administrator type called a *delegated administrator*. The delegated administrator works with specific user groups referred to as business units. User accounts, service accounts, account templates, unlicensed user accounts, and applications are divided into business unit groups and each user or service account, unlicensed user, and account template is assigned to only one business unit.

Each delegated administrator is assigned one or more business units that determine the user accounts, service accounts, unlicensed user accounts, account templates, and applications managed by that administrator. Tracking information is also displayed based on the business unit assigned.

When you log in to the SecureTransport Administration Tool as a delegated administrator, you see a subset of the menus and pages normally available. You are allowed to view the file transfer tracking information, accounts, and applications that are assigned to your business unit.

As a delegated administrators with the **Manage Administrators** privilege, you can create other delegated administrators and perform the following actions:

- Delegate to the new administrator any privileges that you have
- Assign your business unit or any child business unit to the new administrator

Maker and Checker

With SecureTrasnport version 5.3.8 and later, there are two additional available "roles" of the delegated administrator, defined by specific permissions: *Maker* and *Checker*.

- The Maker is a delegated administrator who can create and update user accounts. Accounts created by the Maker will remain in "Pending" verification status until further processing by a Checker.
- The Checker is a delegated administrator who can view in read-only mode all settings associated with an account. The Checker has the responsibility to review and accept or reject the newly created account by the Maker. In fact, these are the only actions the Checker privileges grant: the rest of the Checker permissions are read-only.

The concept of the Maker and Checker is to separate the responsibilities and duties of account creation and account approval. These two roles complement each other and the Checker acts as a second level of user account approval.

When you create a delegated administrator, you can either assign **Read Only** or **Checker Rights** or **Maker Rights**; or you could use any combination of the other available privileges.

Privilege	Description
Read Only	Allows the administrator to view the pages only. This administrator cannot make any changes. Use Read Only for auditing.

Privilege	Description
Checker Rights	Allows the administrator to inspect all settings of user, service and template accounts in an assigned business unit. The Checker administrator can also approve or reject accounts created by a Maker administrator in the assigned business unit.
Maker Rights	Allows the administrator to create user, service and template accounts in an assigned business unit. The Maker administrator can update all account settings before submitting the accounts for approval by Checker administrator.
Create Users	Allows the administrator to create new accounts for an assigned business unit outside of the Maker-Checker user creation flow.
Update Users	Allows the administrator to modify existing accounts for an assigned business unit outside of the Maker-Checker user creation flow.
HelpDesk Rights	Allows the administrator to change the password of users in an assigned business unit. The administrator can also enable or disable a user in the assigned business unit.
Audit Log Rights	Allows the delegated administrator to access Audit Log entries of actions performed by all administrators. When deselected, the administrator can access only the audit log entries of actions performed by their account and no one else's.
Manage Administrators	Allows the administrator to create, modify, and delete delegated administrators for an assigned business unit.
Manage Business Units	Allows the administrator to create, modify, and delete business units.
Manage Applications	Allows the administrator to create, modify, and delete applications other than Shared Folder type applications for an assigned business unit.
Manage Shared Folders Applications	Allows the administrator to create, modify, and delete Shared Folder type applications for an assigned business unit.
Manage Route Package Templates	Allows the administrator to create, modify, and delete route package templates. **Note** In order these privileges to take effect, the appropriate administrative role should be updated to allow access to the Routes Menu.

Privilege	Description
Manage 'External Script' Step	Allows the administrator to create, modify, and delete any **External Script** steps in a route belonging to route package or route package template. **Note** In order for these privileges to take effect, the appropriate administrative role should be updated to allow access to the Routes Menu.
Manage 'Run as root external scripts'	Allows the administrator to modify the **Run as root administrator** External Script property. **Note** In order for these privileges to take effect, the administrator should have privilege to manage External Script step.
Manage Login Restriction Policies	Allows the administrator to create and maintain login restriction policies. They can also create and manage login restriction policy entries.

When each delegated administrator delegates privileges and assigns business unit to delegated administrators he creates, the result is a hierarchy of delegated administrators where those higher in the hierarchy can have greater responsibility and more privileges than those below them.

For example, a finance delegated administrator with permission for the finance business unit can create an audit delegated administrator who can view the Administration Tool pages and two other delegated administrators to administer business units within finance. The following diagram shows the hierarchy:

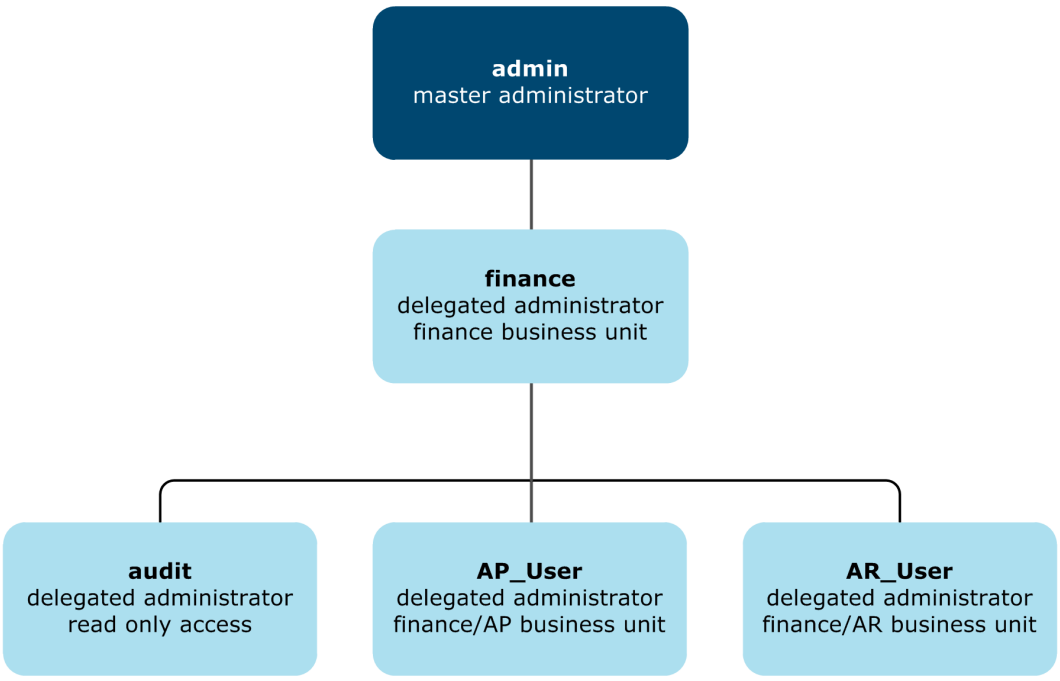


Figure 6. Example delegated administration hierarchy

The following topic describes how to create a delegated administrator:

- *Create a delegated administrator* on page 710 - Provides how-to instructions for creating a delegated administrator.

Create a delegated administrator

Use the following procedure to create a delegated administrator.

1. Create or edit an administrator and set the **Administrative Role** to Delegated Administrator. The panel area expands with various additional Delegated Administrator Settings.
2. Select a **Parent Administrator**. This is the administrator who hierarchically stands on the higher level to the delegated administrator you are creating.
3. Select the business units you want to assign to the administrator, if required. Business Units can be added or modified through the *Business Units* page. Also, here you can only assign business units that are assigned to the Parent Administrator.

   Note General rules of inheritance apply here: if the Business unit assigned to Delegated Administrator has one or more child business units, all those child business units will be also assigned to this Delegated Administrator. If the business unit does not have child, but another Delegated Administrator adds child on a later stage, the newly added child will be automatically assigned to the first Delegated Administrator. If the child Business Unit is removed from the parent Business Unit but continues to exist, the Delegated Administrator will not be assigned to it anymore.

4. Depending on the duties and responsibilities of your new delegated administrator, select either of the following options: **Read Only**, **Checker Rights** or **Maker Rights**.
   - Select **Read Only** to allow the delegated admin read-only access to the user management screens within the selected Business Units. This option deselects and disables editing of all the permission options that follow.
   - Select **Checker Rights** to allow the delegated admin Checker-only privileges. This option disables all additional permissions for user account creation and modification.
   - Select **Maker Rights** to allow the delegated administrator Maker-only privileges. This option preselects the **Create Users** and **Update Users** permissions. You can still edit HelpDesk and Manage rights.
5. If you do not any of the options in the previous step, you can still assign any of the available privilege options to the new delegated administrator.
6. Click **Save** to add the new administrator.

Administrative roles

When you create or edit an administrator account you can set an administrative role that defines the account's privileges and permissions in the Administration Tool. You can create multiple administrators whose account management capabilities are based on administrative roles you create. Each role can have different account management capabilities. Use the *Administrative Roles* page on the SecureTransport Server to create the roles and assign administrative privileges.

For each role, you control:

- Role type
- Permission to bounce servers
- Access to SecureTransport menus and submenus

The following topics describe and provide how-to instructions for managing administrative roles:

- *Predefined administrative roles on page 711* - Lists the predefined administrative roles.
- *Add an administrative role on page 714* - Provides how-to instructions for adding an administrative role.
- *Edit an administrative role on page 717* - Provides how-to instructions for editing an administrative role.

Predefined administrative roles

The following administrative roles are predefined:

- **Master Administrator** – Access to all menus, tabs, and pages of the Administration Tool. Cannot be modified.
- **Database Administrator** – Access to the *Database Settings* page only. Access to the *Setup Oracle* page is not included when SecureTransport is running on the embedded database. Only the `dbsetup` administrator can have this role. Information is maintained in the file system so that `dbsetup` can log in to the Administration Tool when the database is not running. Cannot be modified.

  Note If, as the `dbsetup` administrator, you change your password, have your password expired, or have your account disabled or enabled while there is no connection to the database, that change is not recorded in the database. The next time you log in as the `dbsetup` administrator with the database connected, the change is overwritten by the information from the database.

- **Setup Administrator** – Access through the custom **Configure** menu to pages required to perform the post-installation tasks and setup operations described in the *SecureTransport Getting Started Guide*. Cannot be modified.
- **Account Manager** – Access restricted to the *User Accounts*, *Unlicensed User Accounts, Service Accounts*, *Templates*, and *Business Units* pages on the **Accounts** menu and all pages on the **Access** menu.

- **Application Manager** – Access restricted to the pages on the **Service Accounts** and **Applications** menus.
- **Delegated Administrator** – Configurable access to menus and data based on the function the administrator has within an organization. This administrator type is created by a master administrator or a parent delegated administrator.

Default credentials

To improve security, an administrator must change their default password after the first time they log in.
On fresh installation, all predefined administrator accounts have default passwords set to *expire*, which makes it necessary to change the password at next login. A master administrator can change all administrator passwords. An administrator can reset their own expired password by using either the Administration Tool or the REST API 2.0 resource *PATCH /myself*.

Access rights and restrictions

The following table illustrates the access rights and restrictions of the default restriction levels for SecureTransport administrators. You can modify the access rights for the Account Manager, Application Manager, and Delegated Administrator roles:

Menus and Pages	Master Administrator	Setup Administrator	Account Manager	Application Manager	Delegated Administrator
Operations					
Server Control	P	P	—	—	—
Cluster Management	P	—	—	—	—
Server Usage Monitor	P	—	—	—	—
File Tracking	P	—	—	—	P
Server Log	P	P	—	—	—
Audit Log	P	P	—	—	P
Server Configuration	P	—	—	—	—
Support Tool	P	—	—	—	—

Menus and Pages	Master Administrator	Setup Administrator	Account Manager	Application Manager	Delegated Administrator
Setup					
Certificates	P	P	—	—	—
FTP Settings	P	—	—	—	—
AS2 Settings	P	—	—	—	—
SSH Settings	P	—	—	—	—
Admin Settings	P	—	—	—	—
PeSIT Settings	P	—	—	—	—
AdHoc Settings	P	—	—	—	—
Database Settings	P	P	—	—	—
Axway Sentinel	P	—	—	—	—
Server License	P	P	—	—	—
Command Logging	P	—	—	—	—
Transfer Logging	P	—	—	—	—
Holiday Schedule	P	—	—	—	—
Miscellaneous	P	—	—	—	—
File Archiving	P	—	—	—	—
TM Settings	P	—	—	—	—
Network Zones	P	—	—	—	—
Authentication - All submenus	P	—	—	—	—
Account					

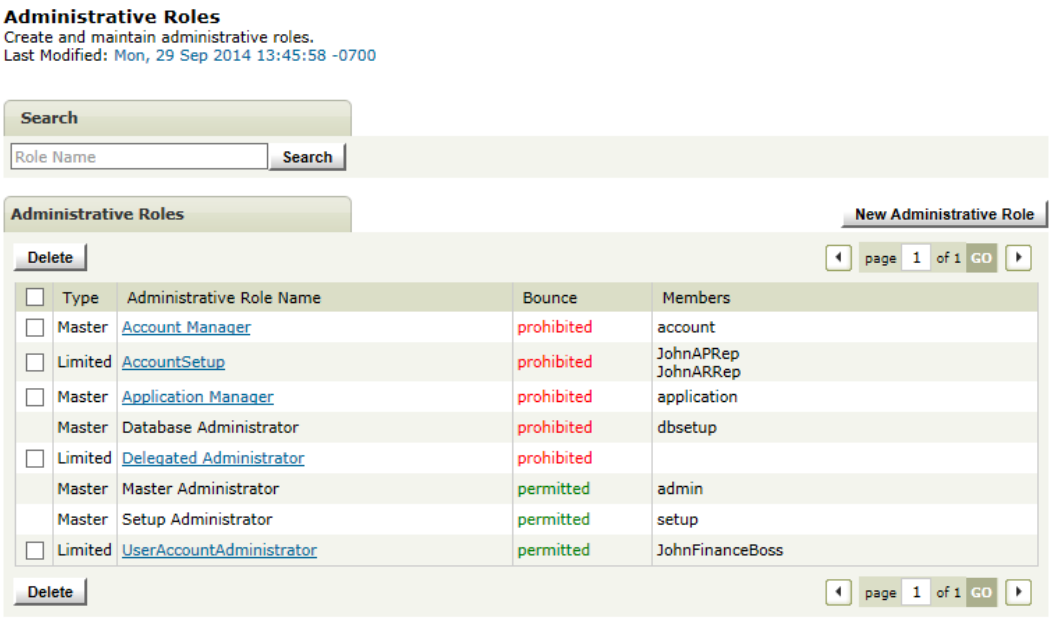
Menus and Pages	Master Administrator	Setup Administrator	Account Manager	Application Manager	Delegated Administrator
User Accounts	P	—	P	—	P
Unlicensed Users	P	—	P	—	P
Service Accounts	P	—	P	P	P
Import/Export	P	—	—	—	P
Administrators	P	—	—	—	P
Change Password	P	—	—	—	P
Manage Roles	P	—	—	—	—
Account Templates	P	—	P	—	P
Site Templates	P	—	P	—	P
System	P	—	—	—	P
Business Units	P	—	P	—	P
Active Users	P	—	—	—	—
Access—All submenus	P	—	P	—	—
Application—All submenus	P	—	—	P	P
Routes—All submenus	P	—	—	—	—

Add an administrative role

Use the following procedure to add administrative role.

1. Select **Accounts > Manage Roles**.

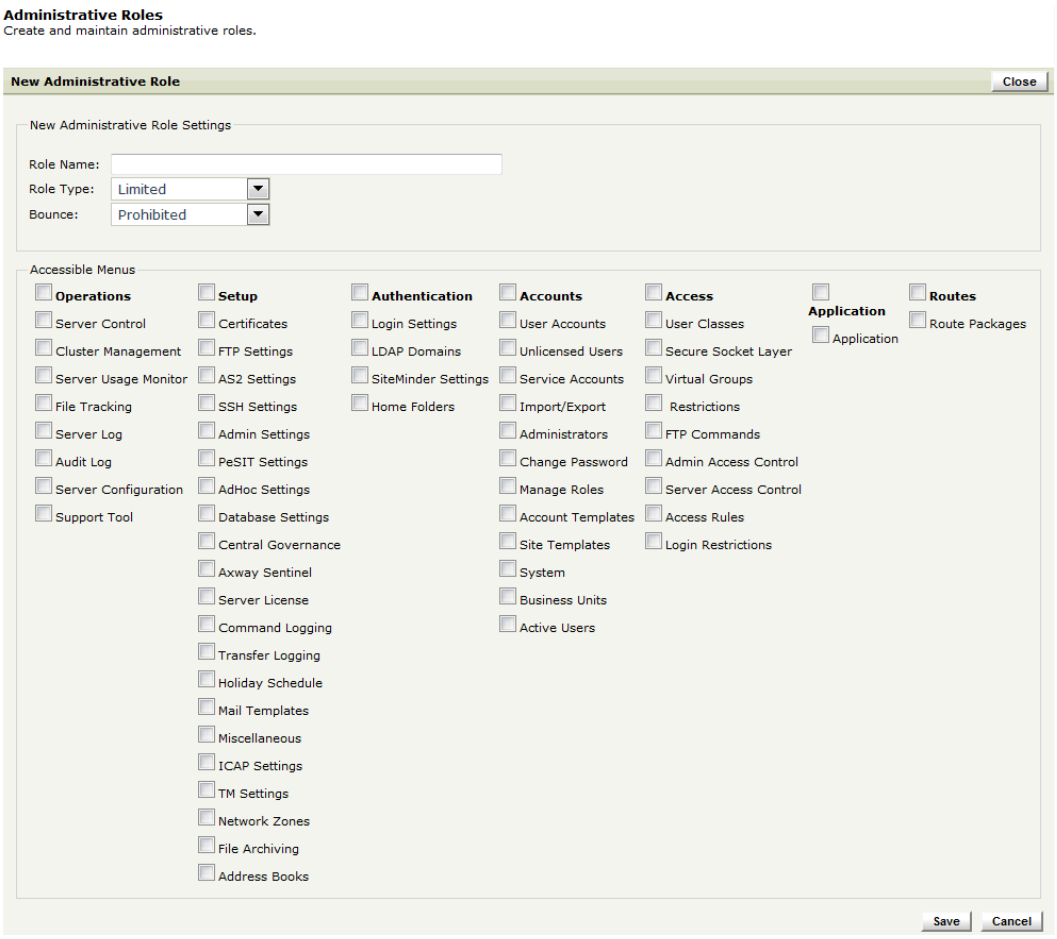
   The *Administrative Roles* page is displayed.



If you do not see roles with Master in the Type column, you are logged on as a user with a limited role.

2. Click **New Administrative Role**.

   The *New Administrative Role* window is displayed.



3. Type the **Role Name**.
4. Select the **Role Type**.
    - Select **Master** to give this role complete privileges over the Administration Tool menus selected under **Accessible Menus** and access to all accounts and business units. Only an administrator whose role has Master type can select this.
    - Select **Limited** to enable limiting the business unit access and privileges for administrators with this role. The Delegated Administrator role has **Role Type** set to **Limited**. A user with a limited role and **Manage Roles** access cannot access Master roles or his own role. **Limited** is the default setting.
5. Select values from the **Bounce** drop-down list.
    - Select **Permitted** to enable administrators at this level to bounce (manually signal running server processes to reload their configurations) servers.
    - Select **Prohibited** to deny these same privileges. **Prohibited** is the default setting.

6. Under **Accessible Menus**, do the following:
   - Select the menus that the administrative role can access. Select the checkbox for the column heading to select all the menus in the column.
   - Clear the menus that the administrative role cannot access. Clear the checkbox for the column heading to clear all the menus in the column.
7. Click **Apply**.

   The new administrative role is added to the *Administrative Roles* page.

Edit an administrative role

Use the following procedure to edit an administrative role.

1. Select **Accounts > Manage Roles**.

   The *Administrative Roles* page is displayed.
2. Click the name of the administrative role you want to edit.

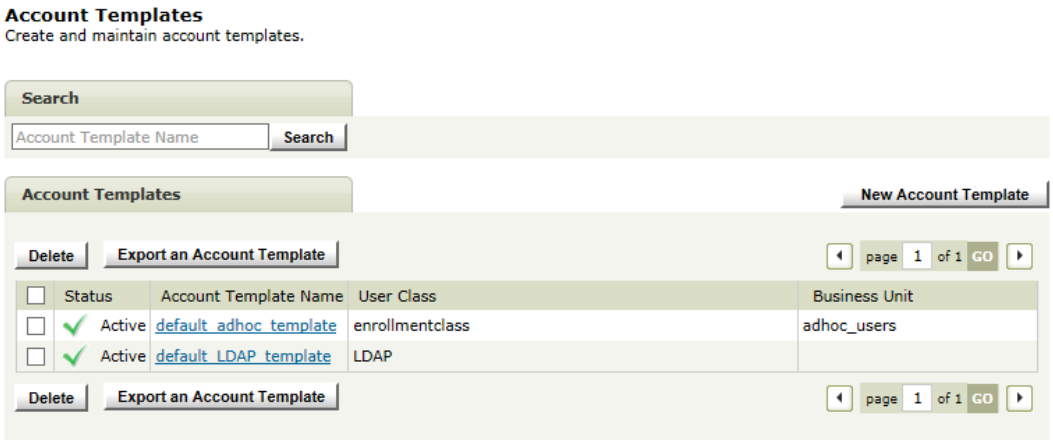
   The *Edit Administrative Role* dialog box is displayed.

   Note You cannot edit the Master Administrator, Setup Administrator, or Database Administrator roles or your own role.
3. Edit the values as necessary, and then click **Apply**.

   You are returned to the *Administrative Roles* page.

Account templates

You can create an account template that can be used by LDAP, SSO, or other external user repositories. Use account templates to avoid duplicating users between your user repository and SecureTransport. Account templates let SecureTransport use the user names and passwords that exist in external user repositories such as LDAP, SSO, Active Directory, or a third party database. SecureTransport does not need to synchronize with any external source. By using the value set in the User Class field, the account templates map the user in real time to SecureTransport.



The template uses the user class as a type of dividing mechanism, allowing you to create different templates for different user functions based on the User Class. A user can be assigned a User Class based on the following items: User Type, User Name, User Group, and From address (the IP address or host name of the logged in user). Create a specific user class for each external repository that you are using.

Templates can also be assigned a business unit.

The following topics describe how to assign external users to account templates and how to manage account template. They also list the account template required values.

- *Account templates and external users* on page 718 - Describes how to assign external users to account templates.
- *Account template required values* on page 719 - Lists the account template required values.
- *Manage account templates* on page 719 - Provide how-to instructions for managing account templates.

Account templates and external users

SecureTransport assigns external users to the account template using the following steps:

1. Determine the User Class based on the already known values for the UID, GID, User Type, and IP address.
2. Compare the determined User Class with the defined User Class in all enabled external account templates. Since the User Class in the template can contain wildcards there might be more than one template that matches the User Class of the currently logged user. In this case, the templates are sorted alphabetically, and the first one is selected.
3. If the User Class matches the User Class of a template, SecureTransport tries to determine the new UID, GID, and Home Folder values as defined in the template. The templates can contain expressions in the supported expression language to dynamically select the UID, GID, or Home Folder. The result is one of the following:
    - If the system fails to determine even one of the required attributes (UID, GID, or Home Folder) from the template the user is *not* assigned to that template and the login fails.
    - If the system manages to determine all of the required attributes, the currently logged external user is assigned to the selected template.
    - If the User Class does not match any of the User Classes in the account template, the server treats the user as a regular external user.
4. If the user is assigned to a template, the UID, GID, and Home Folder are determined from the template, and the values used to determine the User Class are ignored. If the home folder of the user is missing, it is automatically created with the correct permissions.
5. After the external user is mapped to a template the user is automatically assigned a User Type of Virtual, the same as a regular user account.

Account template required values

Each template has several required values: **Account Template Name**, **User Class**, **UID**, **Group ID**, and **Change Home To**.

There are three ways to configure a template once you have specified the **User Class** and **Account Template Name**, and **Business Unit**:

- **Hardcoded values** – The values of the **UID**, **Group ID**, **Change Home To**, and **Notes** fields are explicitly specified in the template. In this scenario every external user mapped to the template uses the same home folder and has the same UID and Group ID (GID).
- **Expressions** – In this scenario the values of the **UID**, **Group ID**, **Change Home To**, and **Notes** fields are specified expressions in the supported expression language. Usually the values of the attributes are different for each external user.
- **Mixed** – Some values are hardcoded and some are expressions or text that includes hardcoded values and expressions. For example you can specify the **Change Home To** as `/tmp/users/${stenv['loginname']}` which means that the home folder of every external user is determined by adding the path `/tmp/users/` and the login name.

Directory path names in an account template are case sensitive.

Note If you configure the home directory of an account template that is used for LDAP or SSO users to include the user name in the home directory, and LDAP or SSO user whose user name contains one or more of the characters `<`, `>`, `#`, and `\` or begins or ends with a space character cannot log in to SecureTransport. This is due to operating system limitation on file names. To allow such LDAP or SSO users to use an account template, use the UID or some other user-unique value to name the home directory.

Manage account templates

The following topics provide examples and how-to instructions for managing account templates:

- *Add an account template* on page 720
- *Enable an account template* on page 726
- *Disable an account template* on page 727
- *Certificates for an account template* on page 727
- *Configure transfer sites for an account template* on page 727
- *Configure transfer profiles for an account template* on page 728
- *Configure routes for an account template* on page 728
- *Configure subscriptions for an account template* on page 731
- *Examples of expressions in an account template* on page 733
- *Export an account template* on page 736

Add an account template

Use the following procedure to add an account template.

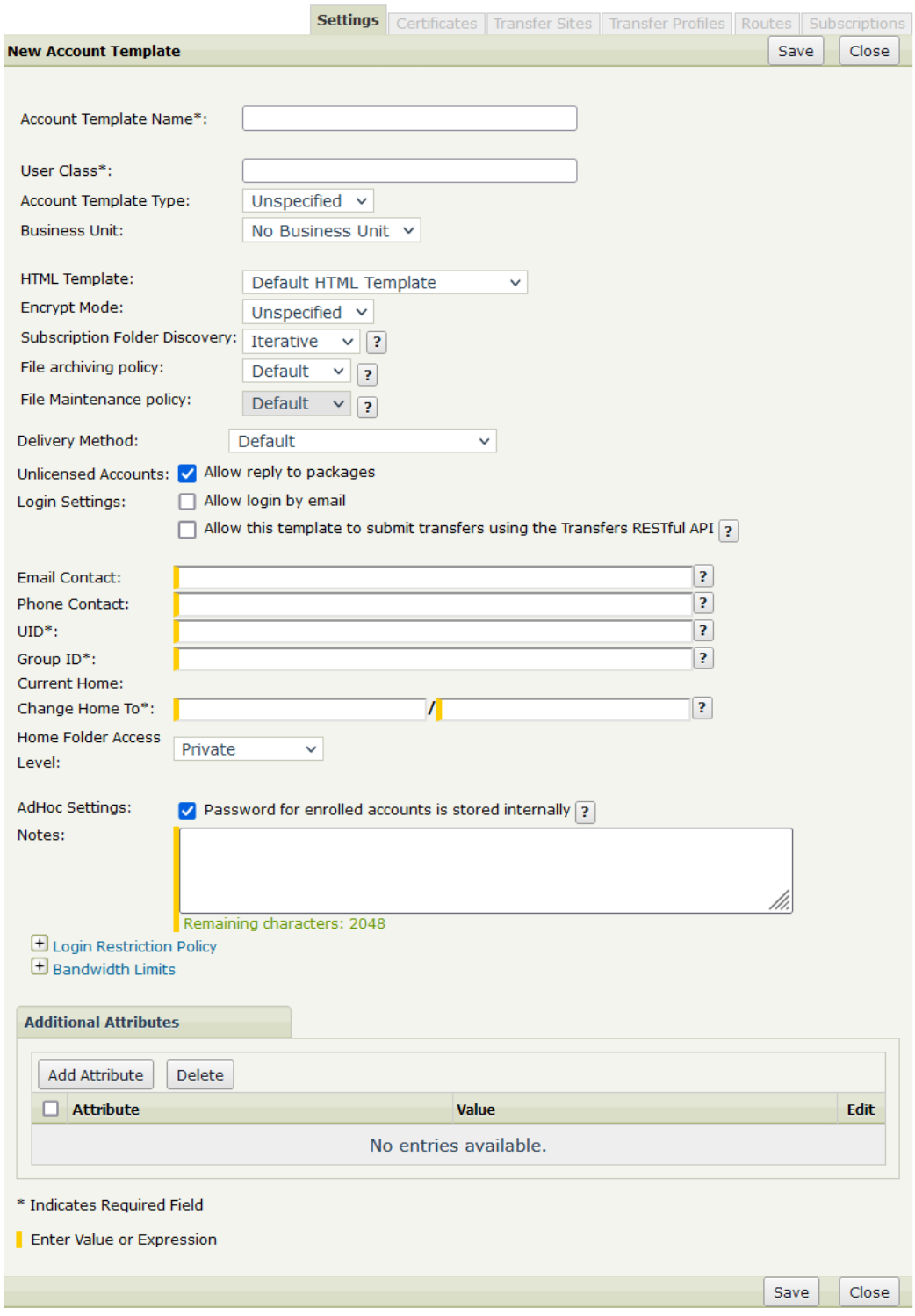
1. Select **Accounts > Account Templates**.

   The *Account Templates* page is displayed.

2. Click **New Account Template** to open a new account template.

   The *New Account Template* page is displayed.

   Note The *Address Book Settings* are only displayed if the Address Book feature is enabled (the value of the `AddressBook.Enabled` configuration option is set to **true**). For Address Book account level configuration instructions, refer to *Address Book account level configuration* on page 249.



Use hardcoded values, expressions in the supported expression language, or a combination of both to complete the fields for the account settings. Required fields are marked by an asterisk. Fields that accept either hardcoded values or expressions are indicated by a vertical yellow bar.

3. Enter a name for the template in the **Account Template Name** field.

4. Enter a pattern that uses question mark (`?`) to match one character and asterisk (`*`) to match any string of characters in the **User Class** field. This account template is associated with users in all classes whose names are matched by the pattern. For example, to associate the template with all users, enter `*`.
5. To place users in a **Business Unit**, select a business unit from the list. Leave the setting as `No Business Unit` if users are not part of a business unit.
6. To specify an HTML template to be used when users log in using the web client, select a value from the **HTML template** drop down.
7. Select **Encrypt Mode**.

   This field can enable repository encryption for users associated with this template.

   - **Unspecified** (default) – Repository encryption is enabled based on the `EncryptClass` user class evaluation.
   - **Enabled** – Repository encryption is enabled for users associated with this template.
8. Select **Subscription Folder Discovery**
   This field determines the subscription folder discovery mode. For accounts with multiple subscriptions, the number of subscriptions and the target folder depth may impact performance.

   - **Iterable** (default): Subscription folder discovery is performed by iteration over all of the account's subscriptions while trying to match the target folder.
     Tip: choose this mode when the number of subscriptions is small and the target folder depth is large.
   - **Recursive**: Subscription folder discovery is performed by recursive traversal of the target folder hierarchy - the target folder is checked first and if no match is found, then its parent folder is checked. The process continues until a match is found or there are no more folders to check.
     Tip: choose this mode when the number of subscriptions is large and the target folder depth is shallow.
9. Select **File archiving policy**.

   This field determines the file archiving policy.

   - When **Default** is selected, then the following apply:
     1. If the account is assigned to a business unit, it will inherit its policy.
     2. Otherwise, the global archiving policy applies.
   - When **Enabled** is selected, file archiving will be enabled for this account.
   - When **Disabled** is selected, file archiving will be disabled for this account.

     Note If the global file archiving policy is disabled, or if this account is assigned to a business unit with **Allow File Archiving Policy modifying** unchecked, then this option cannot be modified.
10. Select **File Maintenance policy**. When file maintenance is enabled, there are specifics in constructing the account home folder.

This field determines the file maintenance policy.

- When **Default** is selected, then the following apply:
  1. If the account is assigned to a business unit, it will inherit its policy.
  2. Otherwise, the *File Maintenance application* on page 834 applies.
- When **Custom** is selected, the panel expands with a *Custom settings* pane that allows you to modify the global *File Maintenance application* on page 834. The customized policy applies to the accounts assigned to this account template only.
- When **Disabled** is selected, file maintenance will be disabled for this account.

  Note If the global file maintenance policy is disabled, or if this account is assigned to a business unit with **Allow File Maintenance Policy modifying** unchecked, then this option cannot be modified.

11. The **Delivery Method** value controls the options that ST Web Client displays in the *User Access* window.
    - **Disabled** – The user cannot send files using ad hoc file transfers.
    - **Default** – Use the delivery method specified in the account template, if any, or in the **Default Package Delivery Method** field of the *AdHoc Setting* page.
    - **Anonymous** – The sender can choose Anonymous or Challenge.
    - **Account Without Enrollment** – The sender can choose Anonymous, Challenge, or Existing Account.
    - **Account With Enrollment** – The sender can choose Anonymous, Challenge, Existing Account, Enroll Unlicensed, or Enroll Licensed.
    - **Custom** – Select the allowed enrollment types in the **Enrollment Types** field. The sender can chose any of the selected enrollment types.
12. For a custom delivery method, select one or more allowed enrollment types in the **Enrollment Types** field:
    - **Anonymous** – The ad hoc file recipient receives a link to retrieve the files and is not enrolled as a user. The ST Web Client option is **Send attachment link only**.
    - **Challenge** – The ad hoc file recipient receives a link and must answer correctly a challenge question specified by the sender to retrieve the files. The recipient is not enrolled as a user. The ST Web Client option is **Protect attachment link with security question**.
    - **Existing Account** – Do not enroll ad hoc file recipients. Only existing users can receive files. The ST Web Client option is **Send to existing users only**.
    - **Enroll Unlicensed** – If the ad hoc file recipient does not have a user account, the recipient must enroll and create an account before retrieving the files. The ad hoc file recipient becomes an unlicensed user who can only reply once to the email and retrieve the files. Other user attributes are defined by the enrollment template. The ST Web Client option is **Allow recipients to enroll as new Unlicensed Users**.

- **Enroll Licensed** – If the ad hoc file recipient does not have a user account, the recipient must enroll and create an account before retrieving the files. The ad hoc file recipient becomes a SecureTransport user with all the attributes specified in the default enrollment template. The ST Web Client option is **Allow recipients to enroll as new Full Licensed Users**.

13. The **Implicit Enrollment Type** value controls which option ST Web Client selects initially in the *User Access* window. The choices depend on the enrollment types enabled by the **Delivery Methods** and **Enrollment Types** fields.
14. Select **Allow reply to packages** in **Unlicensed Accounts** to allow an unlicensed user associated with this template to reply to emails.
15. Specify **Login Settings**.
    a. Select **Allow this account to login by email** to allow the user to log in using with the value of the **Email Contact** field as well as the **Login Name**.
    b. Select **Allow this account to submit transfers using the ST RESTful API** to enable calls from the SecureTransport REST file transfer API authenticated with the credentials from this account. When this option is selected, the account will be allowed to trigger server initiated transfers using the Transfers RESTful API resource and retrieve the tracking information for these transfers.
16. Enter a value or expression for the **Email Contact**.

    When this email address is the recipient of an ad hoc file transfer email sent from ST Web Client, SecureTransport determines that this user is the recipient. If the user is allowed to log in by email, this is the value used in the **User Name** field of the login page.

    Note You can access the SSO email attribute that was previously mapped to `fdxEmail` with the expression `${sess.STSESSION_SSO.email}`.

    Note Accessing Single Sign-On (SSO) attributes is not possible when using SSO with Kerberos authentication protocol. It is only possible with SAML.

17. Enter a value or expression for the **Phone Contact**.
18. Enter a value or expression for the numeric user ID of the user in the **UID** field.

    On Windows platforms, this field is named **Real User** and is optional.

    Note You can access the SSO UID attribute that was previously mapped to `fdxUid` with the expression `${sess.STSESSION_SSO.uid}`.

    Note Accessing Single Sign-On (SSO) attributes is not possible when using SSO with Kerberos authentication protocol. It is only possible with SAML.

19. Enter a value or expression for the numeric group ID for the user account in the **GID** field. The account uses the system access rights and privileges valid for this user group on the system.

    Note You can access the SSO GID attribute that was previously mapped to `fdxGid` with the expression `${sess.STSESSION_SSO.gid}`.

    Note Accessing Single Sign-On (SSO) attributes is not possible when using SSO with Kerberos authentication protocol. It is only possible with SAML.

20. Enter values or expressions for the home folder in the **Change Home To** fields for the account as an absolute path. When File Maintenance is enabled, consider the following important factors:
    - The base folder must be different than the global one. Otherwise, file maintenance will be performed on the whole global directory.
    - When the account home folder is constructed using an EL expression, the File Maintenance application cannot calculate the real path of the subscription folder and will delete it if it's left empty after the maintenance.

      Note You can access the SSO username attribute with the expression `${sess.STSESSION_SSO.userName}`.

      Note Accessing Single Sign-On (SSO) attributes is not possible when using SSO with Kerberos authentication protocol. It is only possible with SAML.
21. Select **Access Level**. The home folder access level determines whether and which other accounts are able to publish to the home folder of the current account.
    - **Private** – The access level is private. Only the current account is able to publish files to its home folder.
    - **Business Unit** – Account home folder access is limited to the account's business unit. The current account and all accounts in the current account's business unit can publish to this account's home folder.
    - **Public** – Access to the account is public. All accounts are able to publish to this account's home folder.

      Note Access level is applicable only when Advanced Routing feature is used. For more information see *Advanced Routing* on page 864.
22. Select **Password for enrolled accounts is stored internally** in **AdHoc Settings** to generate the account's password during enrollment. If **Password for enrolled accounts is stored internally** is not selected, no password will be generated and stored in the SecureTransport database. When a new account with external password is enrolled, SecureTransport will send out an email notification; but will not send a temporary password.

    Note For SSO end-users you need to uncheck this option.
23. Enter a value or expression for the text description of the user account in the **Notes** field.
24. Select the **Login Restriction Policy**. The Login Restriction Policy defines rules for allow or deny login to users based on the client IP or host and other conditions. For additional information, refer to *Login restrictions* on page 810.

    If a Login Restriction Policy is selected as the global default policy, it will be the inherited default selection for the user account.

    If a Login Restriction Policy is not selected as the global default policy and the Business Unit has a Login Restriction Policy selected, it will be the inherited default selection for the user account.

    If neither a global default Login Restriction Policy or a Business Unit Login Restriction Policy is selected, then the policy selected for the users account will be in effect.

    Note The default inherited Login Restriction Policy can be overridden by selecting a Login Restriction Policy from **On Account Template**.

25. In the *Bandwidth limits* pane select either **Bandwidth Limits Policy** to apply:
    - Default – the current account template inherits its bandwidth limits from the parent business unit or the global bandwidth
    - Custom – the panel expands with two additional options for you to configure: **Inbound limit** and **Outbound limit** (both values in kb/s per user)
    - Disabled – no bandwidth limits are applied to the users assigned to the current account template
26. To add an attribute, click **Add Attribute**. For details, see *Additional attributes* on page 759.
    a. Enter the attribute and value in the **Attribute** and **Value** fields.

       Add Attribute enables the administrator to add custom properties (Key=Value). Also the administrator will be able to access the custom properties (named Attributes) using in any fields in Advanced Routing.

       Some examples of Attributes are:

Attribute	Value
userVars.1	internalEmail@axway.com
userVars.2	ReportsMonitor

       To access attributes, see the following examples:

       ```
       ${account.attributes['userVars.1']}
       ```

       ```
       ${account.attributes['userVars.2']}
       ```

       For example, the `account.attributes` is the selector for attributes of the account used to execute the current route - it has to be written exactly as shown.

       The `userVars.` prefix must be prepended to attribute name.

       All this should be written as an EL expression: `${...}`

    b. Click the attribute Save () icon.
27. Once you have completed the information in the *Settings* pane, click **Save** to create the account template.
28. To enable the account template, click **Enable Account Template**.

    Select the **Certificates**,**Transfer Sites**, **Transfer Profiles**, or **Subscriptions** tabs to add additional information to the template. Those pages are similar to the pages for an account, but permit expressions in some fields.
29. To return to the *Account Templates* page, click **Close** or select **Accounts > Account Templates**.

Enable an account template

Once you have created the template, you must enable it to use it.

1. Select **Accounts > Account Templates**.

   The *Account Templates* page is displayed.
2. Click the name of the template you want to enable to view the template settings.
3. Click **Enable Template** to make the template active.
4. To return to the *Account Templates* page, click **Close**.

Disable an account template

You can also disable an already created template.

1. Select **Accounts > Account Templates**.

   The *Account Templates* page is displayed.
2. Click the name of the template you want to enable to view the template settings.
3. Click **Disable Template** to make the template active.
4. To return to the *Account Templates* page, click **Close** or select **Accounts > Account Templates**.

Certificates for an account template

Like a user account, an account template can have partner certificates and private certificates. It cannot have login certificates.

For more information, see *Manage login certificates* on page 527.

Configure transfer sites for an account template

Use the following procedure to configure transfer sites for an account template.

1. With the account template open, select **Transfer Sites** and click **Add New**.

   You must define the transfer site completely. Transfer sites in an account template do not support site templates.
2. Type a **Site Name**.
3. Select a **Site Type**.
4. In the **Add Transfer Site** box, select the **Transfer Protocol**.

   To comply with AS2 protocols, it is not available.
5. Type values or expressions for the required fields and the optional fields needed to define the transfer site.

   Transfer sites in an account template do not support server-initiated downloads, so the fields used for them are not displayed.

   You can use expressions in the fields indicated by a vertical yellow bar.

6. To use expressions for the checkboxes, select **Use Expression Language**, and, in each field that replaced a checkbox, type `true` for selected or `false` for cleared, or an expression that evaluates to true or false.
7. Click **Add** to save the transfer site.

For example, to select **Use FTPS** for the transfer site depending on the whether the `target` variable contains the string `class`, type the following in the **Use FTPS** field:

```
${stenv['target'].matches('.*class.*')}
```

This expression tests the value of `target` and returns `true` if it contains the string class, `false` if not.

Note If an account template and its transfer site are defined using expressions, you cannot restart failed transfers for that account template using the **Resubmit** button on the *File Tracking* page.

Configure transfer profiles for an account template

An account template can have transfer profiles. You can use expressions in the **Files To Send** and **Receive Files As** fields.

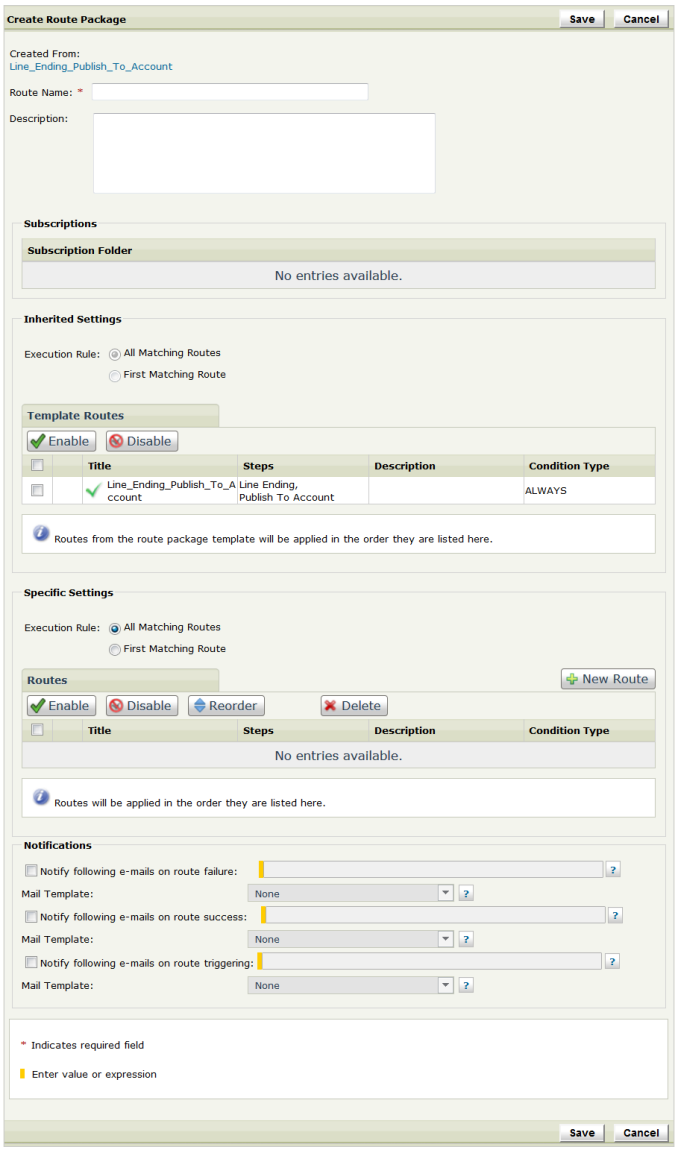
For more information, see *Transfer profiles* on page 640.

Configure routes for an account template

Prior to configuring a route for an account template, the account template should have an Advanced Routing application instance subscription. For account template subscription information, refer to *Configure subscriptions for an account template* on page 731 and to *Subscribe to Advanced Routing application* on page 888. Additionally, route package templates must be available for assignment. For information on creating and managing route package templates, refer to *Manage Route Package Templates* on page 875.

1. With the account template open, select **Routes**, select a route package template, and click **Assign Route**.

   The *Create Route Package* page is displayed. You can navigate to the *Edit Route Package Template* page for the selected route package template by clicking the **Created From** link.



2. In the **Route Name** field, type the desired name of the route. The route name can contain 254 characters or less.

   Note You cannot use the following characters in the route name: `* < > ? " / \ | :`

3. (Optional) Enter a **Description**.
4. In the *Subscriptions* pane:
   a. Click **Assign** to assign an available Advanced Routing application folder to the route.

      The Available Subscriptions page is displayed.

   b. On the *Available Subscriptions* page, select the checkbox for a folder from the

Subscriptions Folder list and click **OK** to assign a folder to the route.

The assigned folder is now listed in the *Subscriptions List*.

To unassign an application folder, select the checkbox for the folder and click **Unassign**.

5. In the *Inherited Settings* pane:
    a. Select the checkbox for a Template Route and click **Disable** to disable an enabled inherited route.
    b. Select the checkbox for a Template Route and click **Enable** to enable a disabled inherited route.

        Note The inherited Execution Rule cannot be changed.

6. In the *Specific Settings* pane:
    a. Determine the **Execution Rule**. Select either **All Matching Routes** (default) or **First Matching Route**.

        When **All Matching Routes** is selected, all matching Routes are executed. When **First Matching Route** is selected, only the first matching Route is executed.

    b. Click **New Route**.

        The *New Route Entry* page is displayed. For Route configuration information, refer to *Manage Routes* on page 881.

        You can also enable, disable, reorder, and delete Routes in the *Specific Settings* pane. For information on enabling, disabling, reordering, or deleting Routes, refer to *Manage Route Package Templates* on page 875.

7. In the *Notifications* pane:
    a. Select **Notify following e-mails on route failure** to be notified on route failure and enter a notification email address, a list of mail addresses, or an expression. For additional email configuration information, refer to *Set up email notifications via SMTP* on page 202.
    b. Select the **Mail Template** from the menu to be used for route failure notifications. For email template configuration information, refer to *Mail templates* on page 195.
    c. Select **Notify following e-mails on route success** to be notified on route success and enter a notification email address, a list of mail addresses, or an expression.
    d. Select the **Mail Template** from the menu to be used for route success notifications.
    e. Select **Notify following e-mails on route triggering** to be notified on route triggering and enter a notification email address, a list of mail addresses, or an expression.
    f. Select the **Mail Template** from the menu to be used for route triggering notifications.
8. Click **Save**.

Configure subscriptions for an account template

1. With the account template open, select **Subscriptions**, select an application, and click **Subscribe**.

   Subscriptions in an account template do not support applications of type Standard Router, so they are not include in the drop-down list.

2. In the *Flow Settings* pane, select the **Existing flow attributes**.

   If **Preserve** is selected, the attributes defined in the *Flow Attributes* pane will be applied only for newly received files which do not have associated flow attributes.

   If **Overwrite** is selected, the attributes defined in the *Flow Attributes* pane will overwrite any existing attributes for incoming files (for example, files published to this folder from another subscription folder).

   When **Append** is selected, only the attributes which are not defined for incoming files will be applied. Existing attributes will be preserved.

3. In the *Flow/Subscription Attributes* pane:

   a. To add an attribute, click **Add Attribute**. For additional information, refer to *Flow and subscription attributes* on page 198.

   Add Attribute enables the administrator to add custom properties (Key=Value). Flow attributes can be used for expression evaluation in Advanced Routing only when the application operates with files. Subscription attributes are bound to the subscription, therefore, they can be used for expression evaluation in all Advanced Routing fields.

   Note Subscription attributes can be accessed using the following expression: `${subscription.attributes['userVars.ATTRIBUTE_NAME']}`. Flow attributes can be accessed using the following expression: `${flow.attributes['userVars.ATTRIBUTE_NAME']}`.

   Some examples of Attributes are:

Attribute	Value
userVars.1	internalEmail@axway.com
userVars.2	ReportsMonitor

   To access attributes, see the following examples:

   ```
   ${account.attributes['userVars.1']}
   ```

   ```
   ${account.attributes['userVars.2']}
   ```

   For example, the `account.attributes` is the selector for attributes of the account used to execute the current route - it has to be written exactly as shown.

The `userVars.` prefix must be prepended to attribute name.

All this should be written as an EL expression: `${...}`

b. Click the attribute Save () icon.

4. Type values or expressions for the required fields and the optional fields needed to define the subscription.

   Subscriptions in an account template do not support server-initiated downloads, so the fields used for them do not appear.

   You can use expressions in the fields indicated by a vertical yellow bar.

5. To use expressions for additional fields including the checkboxes, select **Advanced Expressions**.

   The fields and checkboxes are replaced by fields with a vertical yellow bar.

   - In each field that replaces a checkbox, type 1 for selected (true) or 0 for deselected (false) or an expression that evaluates to 0 or 1.
   - In the other fields, type a value or an expression that evaluates to the value required.

     In the **Compression Type** field for applications that have the **Encrypt File** option such as application of type Basic Application and Site Mailbox, type one of the following values that represent available compression algorithms or an expression that evaluates to one them:

Type	Value
Use preferred (algorithm obtained from PGP key)	-1
Uncompressed	0
ZIP	1
ZLIB	2
BZIP2	3

     In the **Compression Level** field, type an integer between 1 and 9, where 1 represents the least compressed but fastest level and 9 represents the most compressed but slowest level or an expression that evaluates to an integer between 1 and 9. The values that correspond to the levels available when **Advanced Expressions** is not selected are:

Level	Value
Fast	2
Normal	5
Good	7

Level	Value
Best	9

6. Click **Add** to add the subscription to the account template.

Examples of expressions in an account template

You can use expressions on the *Settings* pane, *Transfer Sites* pane, and *Subscription*s pane when creating an account template. The following examples show some of the expressions you can use.

The following table gives examples of expressions in account settings:

Field	Expression	Description
UID	`${sess['STSESSION_LDAP_DIR_uidNumber']}`	Returns the UID from the LDAP session.
GID	`${sess['STSESSION_LDAP_DIR_gidNumber']}`	Returns the GID from the LDAP session.
Home Folder	`${sess['STSESSION_LDAP_DIR_homeDirectory']}`	Returns the home folder specified in the LDAP session.
When you have attribute maps configured, you can use the following named variable expressions instead:		
UID	`${stenv.useruid}`	Returns the UID.
GID	`${stenv.usergid}`	Returns the GID.
HomeDir	`${stenv.homedir}`	Returns the home folder.
If the account template is for licensed or unlicensed users enrolled after receiving notification of an ad hoc file transfer:	If the account template is for licensed or unlicensed users enrolled after receiving notification of an ad hoc file transfer:	If the account template is for licensed or unlicensed users enrolled after receiving notification of an ad hoc file transfer:
Email Contact	`${stenv.recipient_email}`	Returns the email address for recipient of the ad hoc file transfer.

Field	Expression	Description
Home Folder	`${stenv.recipient_email}`	Returns the email address for recipient of the ad hoc file transfer to create a unique home folder.

The following table gives examples of expressions in transfer sites:

Field	Expression	Description
Upload Folder	`/upload/${stenv.loginname}`	Returns the subfolder based on the user login name in the upload folder.
User Name	`${stenv.loginname}`	Returns the user login name.
Certificate	`x509_${stenv.loginname}`	Returns the user login certificate.
With Advanced Expressions selected, you can use the following complex expressions:		
Upload Folder	`/${stenv['target'].replace('^(.*)_(.*)_(.*)$','$2')}` or `/upload/${stenv['target'].matches('.*\\.((jpg)|(gif)|(txt))$') ? stenv['target'].replace('^.*\\.((txt)\|(jpg)\|(gif))$','$1') : 'general/'}`	Returns the upload folder based on the match and replace expression criteria.
Enable SSL	`${stenv['target'].matches('^(ssl).*')}`	Returns a 0 (false) or 1 (true) based on the match criteria.

Note You can also use regular expressions such as `${stenv.target}` only to return the file name or `${filename(stenv.target)}-${random}` to change the file name.

The following table gives examples of expressions in subscriptions:

Field	Expression	Description
Subscription Folder	`mailbox_el_${stenv.useruid}`	Returns the folder using the UID.
	`${flow.attributes['userVars.ATTRIBUTE_NAME']}`	Returns the folder using the attribute name.

Field	Expression	Description
Receive Options Decrypt PGP File As:	`${stenv.loginname}_ ${embedded}`	Returns a file name based on the login name and the embedded file name.
Keep Original As:	`archive/${date ('yyyy.MM.dd')}/ ${filename (stenv.transformation_ input)}`	Returns the location and file name based on the date and the PGP file name.
With Advanced Expressions selected, you can use the following complex expressions:		
Send Options Send Files Directly To:	`${stenv.loginname}_ftp`	Returns a location based on the user login name.
Receive Options Decrypt PGP File:	`${stenv['target']. matches('.*((\\.pgp)| (\\.gpg)|(\\.asc))')}`	Returns a 0 (false) or 1 (true) based on the match criteria.
As:	`${empty embedded ? filename (stenv.transformation_ input).replace('(\\.pgp)| (\\.gpg)|(\\.asc)') : embedded}`	Returns the file name to which the decrypted file is saved.
Keep Original:	`1`	The value `1` represents true. SecureTransport recognizes the field as being selected.
As:	`archive/${date ('yyyy.MM.dd')}/ ${filename (stenv.transformation_ input)}`	Returns the file name based on the original PGP file name.
Use Data Compression	`${extension(filename (stenv.transformation_ input)) .matches('(\\.jpg)| (\\.mov)') ? 0 : 1}`	Returns a `0` when the file extension is `.jpg` or `.mov`. These file types are already compressed and do not require compression.

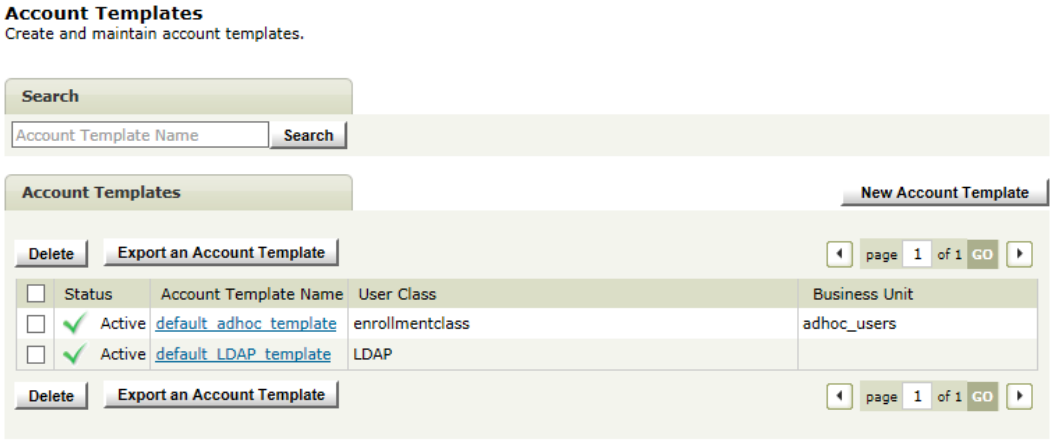
Field	Expression	Description
Compression Type	2	Compresses the file using ZLIB.
Compression Level	5	Compresses the file using the Normal setting.

Export an account template

You can export an account template to an XML file.

1. Select **Accounts > Account Templates**.

   The *Account Templates* page is displayed.



2. In the first column, select the account template to export.
3. Click **Export an Account Template**.

   The *Export Account* page is displayed.
4. Type a password in the **Password** field. This password is used to encrypt the sensitive information contained in the template account. You must use this password when you import the template account to decrypt the sensitive information.
5. Retype the password in the **Re-enter Password** field.
6. Click **Export**.
7. When the XML file with the exported account template is ready, click **Download Exported Accounts** and save the file to your local computer.

Site templates

You can create a site template for a Connect:Direct or a file services interface transfer site. You can use a Site templates to provide the information needed for many accounts in one place instead of creating a transfer site for each account. When you associate a transfer site with the template, that transfer site gets its properties from the site template. When you change the site template, the associated sites are changed.

You can associate the same template with multiple transfer sites in different accounts. If your account has more than one Connect:Direct or file services interface transfer site, you can reuse the same site template for each transfer site, or set up different site templates for each transfer site.

Site template properties are the same as those of a transfer site for the same protocol.

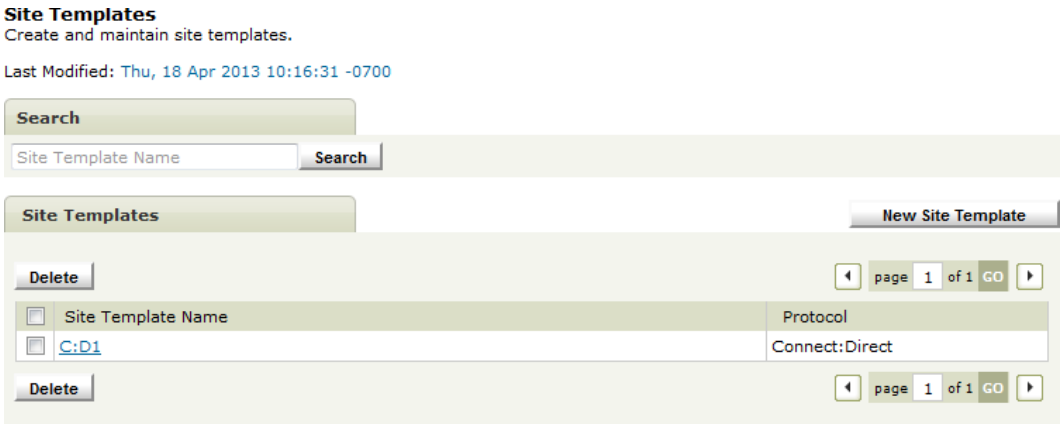
Note Support for the NDM protocol through a Connect:Direct transfer site does not replace or append your Connect:Direct license.

The following topics describe how to manage site templates and how to use a site template to define a transfer site:

- *Manage site templates* on page 737 - Provides how-to instructions for managing site templates.
- *Use a site template to define a transfer site* on page 741 - Provides how-to instructions for using a site template to define a transfer site.

Manage site templates

From the *Site Templates* page, you can create, search for, view, modify, and delete site templates.



The following topics provide how-to instructions for managing site templates:

- *Create a site template* on page 738
- *Search for a site template* on page 740
- *View a site template* on page 740

- *Modify a site template* on page 740
- *Delete a site template* on page 741

Create a site template

You can create site templates for Connect:Direct and file services interface sites.

A site template contains the same fields as the corresponding transfer site. Within those fields you can provide hardcoded values such as a server name or port number, or you can provide a placeholder parameter. You can specify an optional default value as part of a placeholder parameter. This default value can be changed after the site template is applied to a transfer site. When you select a site template while defining a transfer site, you can provide values tailored for a specific transfer site in each field that specified a placeholder parameter.

Note You can create multiple site templates, but each site template must have a unique name.

SecureTransport provides a placeholder parameter you can use when you do not want to hard code values in the fields for the site template. A placeholder parameter is associated only with the site template where it is used. You can, however, specify different placeholder parameters for each site template you create. A placeholder parameter consist of two parts, the placeholder name and an optional default value. The format for entering a placeholder parameter is:

`%{PlaceholderName|DefaultValue}`

where `PlaceholderName` is the name of the placeholder parameter and `DefaultValue` is an optional default value assigned to a specific field.

Note If you are using the optional default value, it must be separated from the placeholder name by the | character.

For example, the placeholder parameter `%{ServerPort|456}` has both a placeholder name and a default value. In this example, the placeholder parameter is entered in the **Local server port** field and a server port number of 456 is used by all transfer sites that apply this site template and accept the default value.

Placeholder names can only contain the following characters: a-z, A-Z, 0-9, and the underscore (_). The first character of a placeholder name cannot be a digit. The default value can use any characters, but to include a right brace (`}`), precede it with a backslash (`\}`).

Placeholder examples:

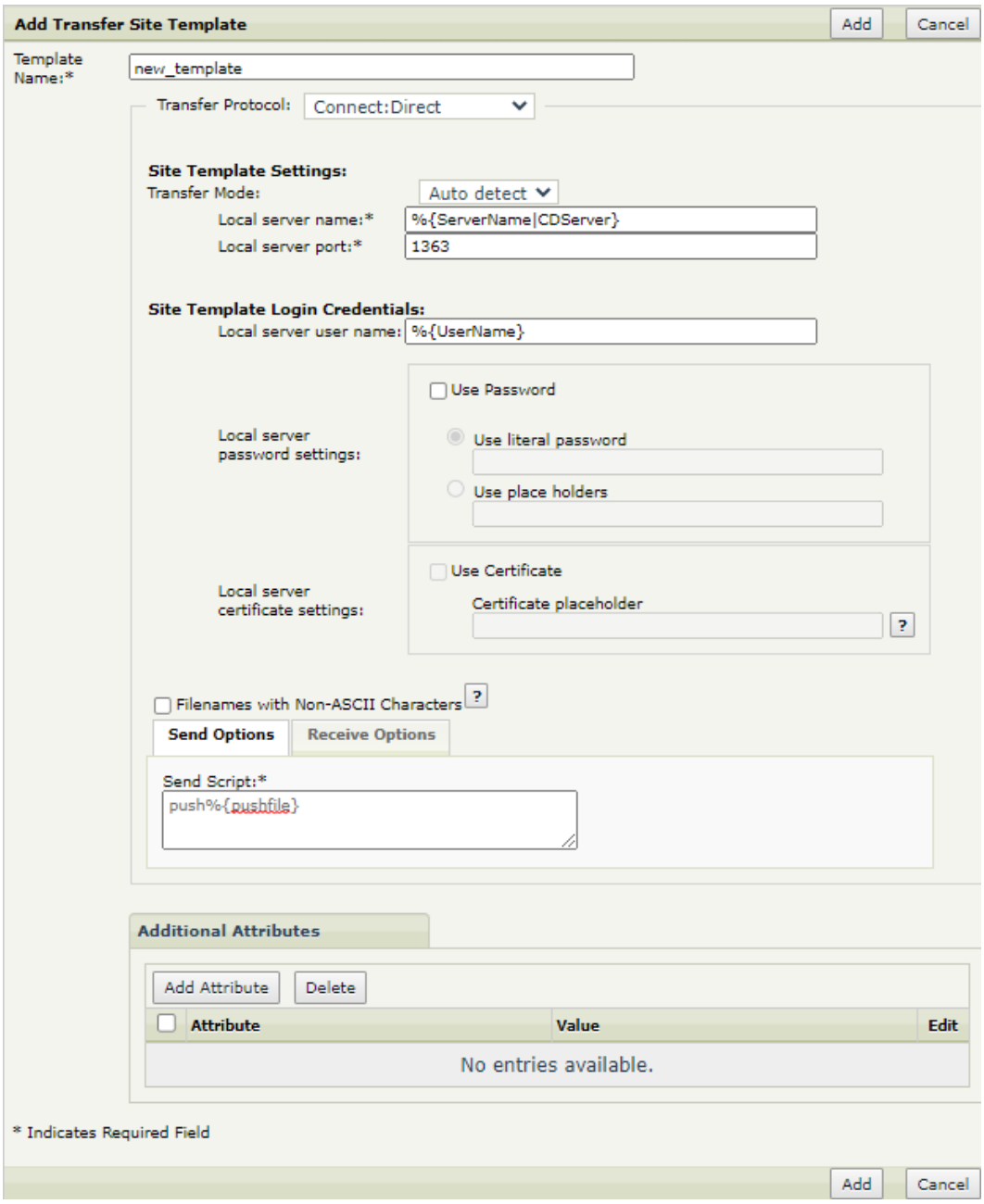
```
%{ServerPort|456}

%{certificate|st_cd_server_certificate_hr}

pull %{pullTransferFile}
```

Note The placeholder parameter cannot include regular expressions.

1. Select **Accounts > Site Templates**.

   The *Site Templates* page is displayed.

2. Click **New Site Template** to create the template.



3. Type a site template name in the **Template Name** field.
4. Select Connect:Direct or the file services interface protocol from the **Transfer Protocol** list.
5. Complete the remaining fields for the transfer protocol you selected.

   Note You can use placeholder parameters for all scripts and fields. For more information about the fields, see *Connect:Direct transfer sites* on page 550 and *File services interface transfer sites* on page 554.
   All the fields for the site template must have a value or a placeholder parameter.

6. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

   Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.

7. Click **Add** to save the site template. Click **Cancel** to close the page without saving the site template.

Search for a site template

Use the following procedure to search for a site template.

1. Select **Accounts > Site Templates**.

   The *Site Templates* page is displayed.

2. In the *Search* pane, type all or part of a site template name and click **Search**. Wildcards are not accepted.

   All site templates that match the search criteria are displayed.

Note You can use individual characters to search for a template, such as typing "1" to see all site templates that contain "1" in the name. If you have site templates named NDM1, 1NDM, and NDM 1, all three site templates are displayed. If you have a fourth site template named NDM2, it is not displayed.

View a site template

Use the following procedure to view a site template.

1. Select **Accounts > Site Templates**.

   The *Site Templates* page is displayed.

2. Click the site template name. The *Edit Transfer Site Template* page is displayed. Use this page to view the site template settings.

3. Click **Save** or **Cancel** to return to the *Site Templates* page.

Modify a site template

Use the following procedure to modify a site template.

1. Select **Accounts > Site Templates**.

   The *Site Templates* page is displayed.

2. Click the site template name. The *Edit Transfer Site Template* page is displayed.

3. Modify any fields you want to change.

4. Click **Save** to return to the *Site Templates* page. Click **Cancel** to close the site template without saving the changes.

Delete a site template

Use the following procedure to delete a site template.

1. Select **Accounts > Site Templates**.

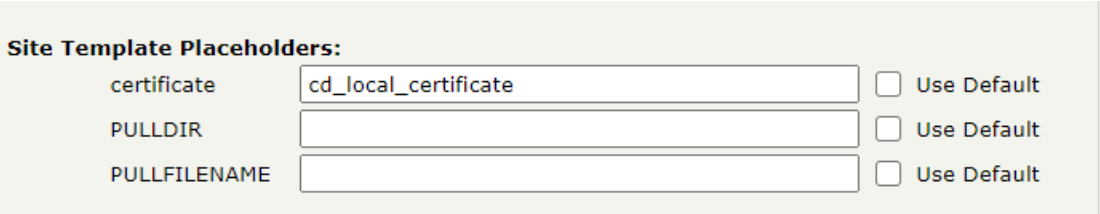
   The *Site Templates* page is displayed.

2. Select one or more site templates.
3. Click **Delete** to remove the site templates. A dialog box displays asking you to confirm the deletion. Click **OK** to continue or **Cancel** to stop.

Use a site template to define a transfer site

Use the following procedure to create a transfer site from a template.

1. Create or edit a Connect:Direct or file services interface protocol site as described in *Create a transfer site* on page 628 or *Edit a transfer site* on page 629.
2. Select the desired template from the **Site Template** drop-down list.
3. If the selected template uses placeholder parameters, they are listed at the bottom of the page under **Site Template Placeholders**.

   If a default value was provided by the site template, you can modify it for this account.

   (Optional) To have the placeholder default values automatically filled in when the site template is updated, select the **Use Default** checkbox for one or more placeholder parameters.

   When you modify a site template, the changes are automatically applied to all transfer sites that use that template. For example:

   - Add or remove a placeholder parameter adds or removes it to or from all associated transfer sites.
   - Modify an optional default value of a placeholder parameter updates it in all associated sites that have the Use Default checkbox selected for that placeholder.
   - Delete the default value of a placeholder parameter clears it in all associated sites that have the Use Default checkbox selected for that placeholder.

4. Click **Add** or **Save** to save your changes and close the *Add Transfer Site* or *Edit Transfer Site* page. Click **Cancel** to close the page without saving your changes.

System users

You can allow users who already have an account on the computer running SecureTransport to log in without creating a SecureTransport account by configuring the settings on the *Password Files* page in the Administration Tool. This page is only applicable for real users in SecureTransport.

The following topics describe real users and provide how-to instructions for managing password files:

- *Real users* on page 742 - Describes real system users.
- *Manage password files* on page 743 - Provides how-to instructions for managing password files.

Real users

Note Real users cannot be granted access to SecureTransport on non-root installations.

Real users are the users defined at the operating-system level. Access rights to the server file system for real users are based on the underlying operating system file access rights. Real users can be defined locally on the server (for instance, on a UNIX-based platform in `/etc/passwd`, or on Windows as a computer-specific local user) or on a network resource (NIS for UNIX or on a domain controller for Windows).

Set a home folder for each real user to ensure that the user is not logged into a randomly-selected directory when logging in to SecureTransport.

Note Real users can view the complete file system of the SecureTransport Server, regardless of the location of the home folder.

The following topics describe real users on UNIX and on Windows:

- *Real users on UNIX* on page 742
- *Real users on Windows* on page 742

Real users on UNIX

UNIX real users are the users defined in `/etc/passwd`, or in NIS. Real users are created at the system level. They can login using `telnet` or `rlogin`, in addition to FTP access only if their rights and permissions give them access.

Real users on Windows

Windows real users are created locally on the server or on the domain controller with the system controls. For Windows Server, the system controls are accessed through **Control Panel > User Accounts > Add or remove user accounts**.

For more information on Windows users, refer to the Microsoft documentation.

Note Real users set up on the SecureTransport Edge are unable to log into either a SecureTransport Edge or SecureTransport Server. You must create a user account for each real user set up on a SecureTransport Edge to allow log ins.

Note If the account home folder prefix is on a shared network, specify a real user that has access to it. The real user must be part of the domain, not a local user for one of the cluster nodes; otherwise the other nodes in the cluster cannot impersonate it to access the shared location.

Note The specified real user needs to added in a password vault file. For more information, refer to *Add a user to a password vault* on page 745.

When SecureTransport is running on a Windows platform, the *Password Files* page provides an additional option to specify password vaults. A password vault stores user names and passwords of real users on Windows, is used to mimic virtual users on Windows, and is applicable only for Windows. See *Manage password files* on page 743

Manage password files

Use the *Password Files* page to add, enable, disable, and delete password entries for real users.

By default the *Password File List* contains a disabled entry for real users. On a UNIX-based system, you cannot delete this entry or add an entry. On Windows, you can delete the entry for real uses and add password vault entries. SecureTransport stores password vaults in the SecureTransport database.

The following topics provide how-to instructions for managing password files and password vaults:

Add password file entry

Use the following procedure to a password file entry.

1. Select **Accounts > System**.

   The *Password Files* page is displayed.

2. Click **Add Password File**.

   A new row is displayed in the table.

3. In the **Type** list, select `Real Users` or `Password Vault`.

   Note On UNIX-based platforms, `Real Users` is the only option in the **Type** list. On Windows platforms, there are two options: `Real Users` and `Password Vault`.

4. If you selected `Password Vault`, in the **Location** column, enter a Windows file path for the password vault.

   SecureTransport stores the password vault entries in the database and uses the value of the **Location** field to identify the password vault. SecureTransport does not create a file for the password vault.

5. Click the Save icon ( ) in the last column of the list.

Enable or disable password file entries

Once you have created the password file entry, you can enable it. You can disable entries you want to keep but not use.

1. Select **Accounts > System**.

   The *Password Files* page is displayed.

2. Select the entries to enable or disable.

3. Click **Enable** or **Disable**.

Edit a password file entry

Use the following procedure to edit a password file entry.

1. Click the Edit icon ( ) in the last column of the entry to edit.

2. Make changes in the **Type** or **Location** columns.

3. Click the Save icon ( ) in the last column.

Delete password file entries

If you no longer want to keep the password file entry, you can delete it.

1. Select **Accounts > System**.

   The *Password Files* page is displayed.

2. Select the entries to delete.
3. Click **Delete**.

   SecureTransport displays a confirmation dialog.

4. Click **OK** to delete the entries.

Add a user to a password vault

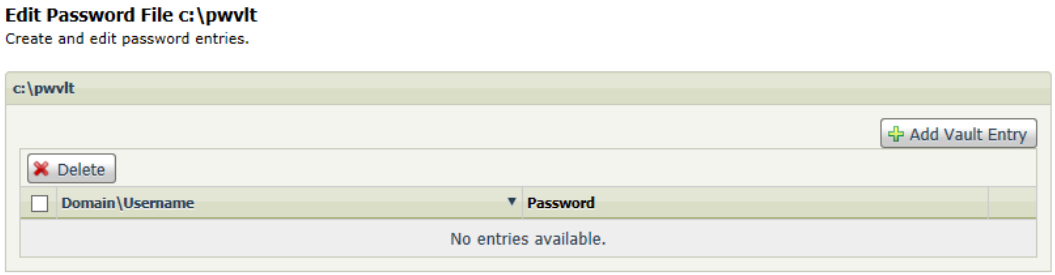
Use the following procedure to add a user to password vault.

1. Select **Accounts > System**.

   The *Password Files* page is displayed.

2. Click **Edit File** in the **File Action** column for the password vault entry.

   The *Edit Password File* page is displayed.



3. Click **Add Vault Entry**.

   A new row is displayed in the table.

4. In the **Domain\Username** column, type the domain or computer name and user name using one of the following formats:

   Domain\Username or *Computer\Username*

   where *Username* is the valid domain or computer user.

5. In the **Password** column, type the password of the user.
6. Click the Save icon ( ) in the last column of the list.

Note If the password is incorrect, or the specified user does not have the relevant Windows local or domain permissions, the addition of the user fails with an appropriate error message. If the reason is wrong permissions, contact the domain administrator.

Edit a user in a password vault

Use the following procedure to edit a user in a password vault.

1. Select **Accounts > System**.

   The *Password Files* page is displayed.

2. Click **Edit File** in the **File Action** column for the password vault entry.

   The *Edit Password File* page is displayed.

3. Click the Edit icon ( ) in the last column of the user to edit.
4. Make changes in the **Domain\Username** or **Password** columns.
5. Click the Save icon ( ) in the last column of the list.

Note If the password is incorrect, or the specified user does not have the relevant Windows local or domain permissions, the addition of the user fails with an appropriate error message. If the reason is wrong permissions, contact the domain administrator.

Delete users from a password vault

Use the following procedure to delete users from a password vault.

1. Select **Accounts > System**.

   The *Password Files* page is displayed.

2. Click **Edit File** in the **File Action** column for the password vault entry.

   The *Edit Password File* page is displayed.

3. Select the entries to delete.
4. Click **Delete**.

   SecureTransport displays a confirmation dialog.

5. Click **OK** to delete the users.

Purge a password vault

Use the following procedure to purge a password vault.

1. Select **Accounts > System**.

   The *Password Files* page is displayed.

2. Click **Purge File** in the **File Action** column for the password vault entry.

   SecureTransport immediately deletes the password vault from the database and deletes the entry from the *Password Files* page.

Business units

Use business units to encapsulate certain information necessary for transfers into a single entity. When you create accounts and templates, you can specify a business unit to represent a particular set of information about the transfer. When you create a delegated administrator, you can specify business units (including users assigned to that business unit) to be managed by the delegated administrator. For more information, see *Delegated administration* on page 707.

The information contained in a business unit includes business unit name, base folder, a parent business unit, whether administrators are allowed to modify the base folder or the home folder, and which HTML template to use when users belonging to this business unit log in using the web client.

Use the *Business Units* page to see the available business units, search the list, delete business units, and invoke the editing process.

Use the *Business Units Settings* page to edit settings for a business unit.

Note Only master administrators and delegated administrators with permissions for managing business units can create and delete business units and modify business a business unit's properties.

The following topics provide how-to instructions for managing business units:

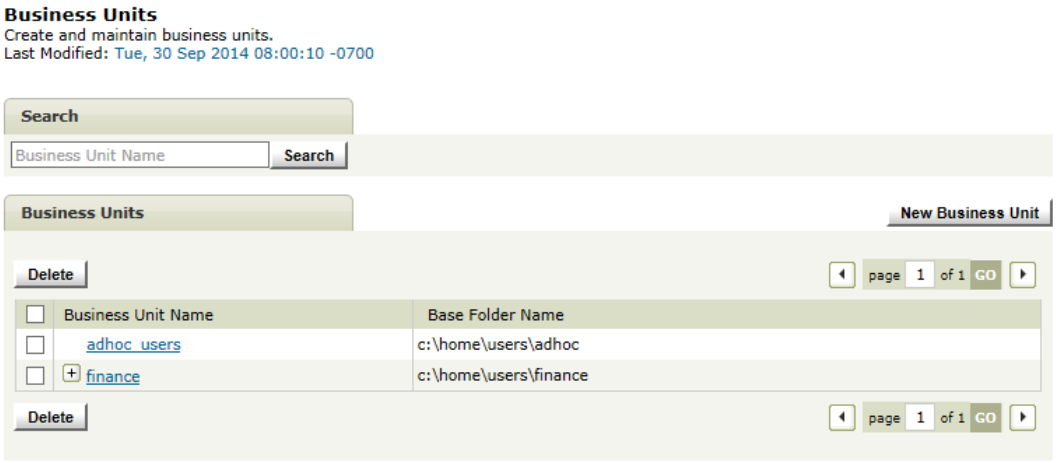
- *See available business units* on page 747
- *Create or edit a business unit* on page 748
- *Delete a business unit* on page 758

See available business units

Use the following procedure to display a list of business units.

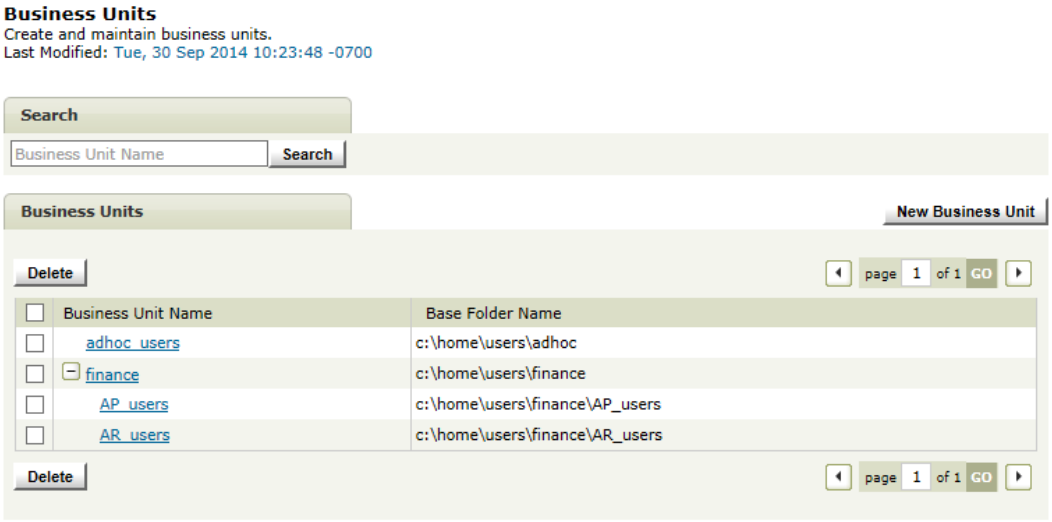
1. Select **Accounts > Business Units**.

   The *Business Units* page is displayed. Business units that have child business units associated with them are called *parent business units* and are displayed with a plus sign (+).



2. (Optional) To display child business units, click the plus sign next to a parent business unit.

The child business units are displayed under their respective parent units.



Note Delegated administrators do not see business units as parents and children. All business units associated with a delegated administrator are displayed at the same level on the page.

Create or edit a business unit

Use the following procedure to create or edit a business unit.

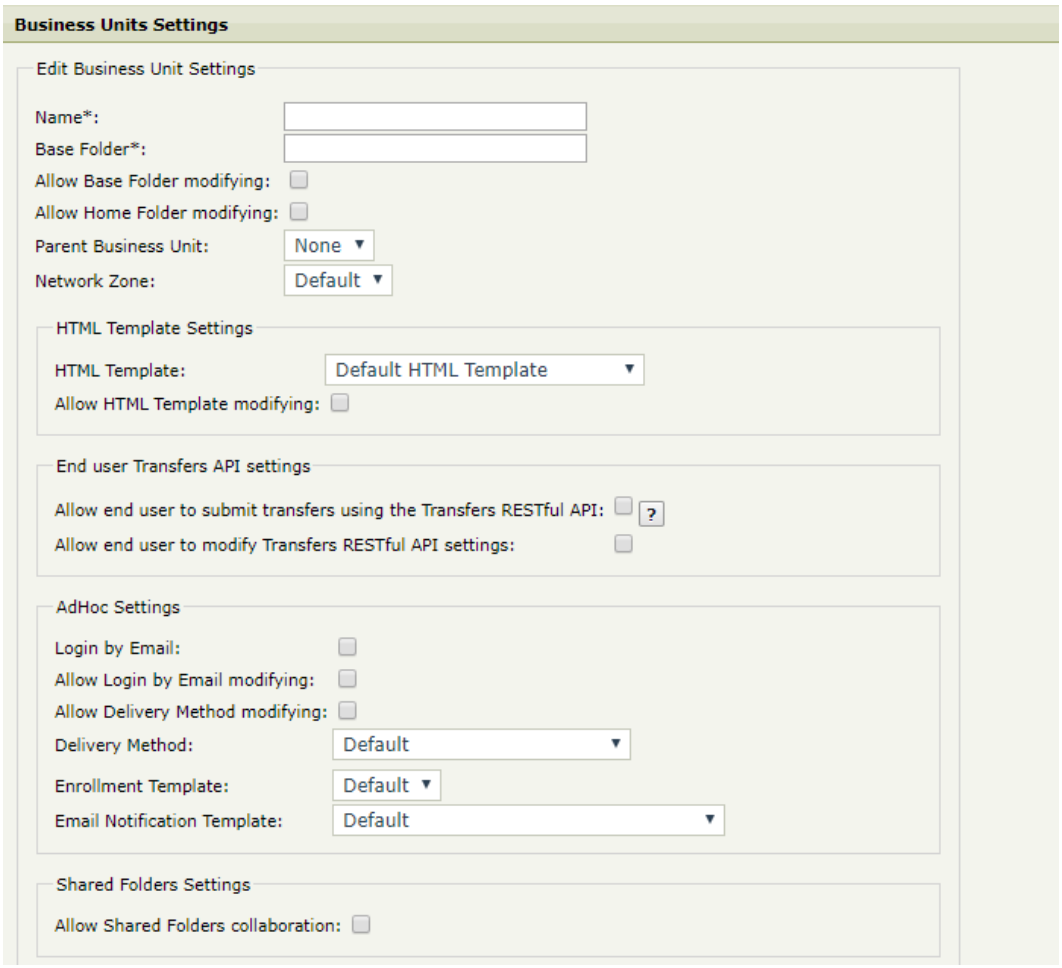
1. Select **Accounts > Business Units**.

   The *Business Units* page is displayed.

2. Click **New Business Unit** or click the name of the business unit you want to edit.

   The *Business Units Settings* page is displayed.

   Note The *Address Book Settings* pane is only displayed if the Address Book feature is enabled (the value of the `AddressBook.Enabled` configuration option is set to **true**).



3. If you are creating a business unit, enter a value in the **Name** field. Business unit names are not case sensitive.
4. In the **Base Folder** field, specify a folder that is to contain the home directories of new accounts belonging to this business unit.
5. Select the **Allow Base Folder modifying** checkbox to allow administrators to change the base folder when creating an account.
6. Select the **Allow Home Folder modifying** checkbox to allow administrators to change the account name suffix when creating an account.
7. In the **Parent Business Unit** drop-down list, select the name of the parent business unit. If you do not want this business unit to be a child, select `None`, the default.

   Note If the Business unit assigned to Delegated Administrator has child Business unit, the child Business Unit will be also assigned to this Delegated Administrator. If Business unit does not have child, but another Delegated Administrator adds child, the newly added child will be automatically assigned to the first Delegated Administrator. If the child Business Unit is removed from the parent Business Unit but continues to exist, the Delegated Administrator will not be linked anymore.

8. In the **Network Zone** field, select the network zone that defines the public URL prefix for users in this business unit.
   Select **Default** to use the setting in the default network zone, or choose a specific network zone to use the setting defined for that zone

   For more information, see *Manage the communication across Transaction Manager, protocol and proxy servers* on page 230.
9. In the *HTML Template Settings* pane:
   a. From the **HTML Template** drop-down, select the HTML template you want to use for accounts and account templates that belong to this business unit.
   b. Select the **Allow HTML Template modifying** checkbox to allow administrators to change the HTML template when editing or creating an account for this business unit.
10. In the *End user Transfers API* settings pane:
    a. Select **Allow this account to submit transfers using the Transfers RESTful API** to enable calls from the SecureTransport REST file transfer API authenticated with the credentials from accounts in the business unit. When this option is selected, the account will be allowed to trigger server initiated transfers using the Transfers RESTful API resource and retrieve the tracking information for these transfers.
    b. Select **Allow end user to modify Transfers RESTful API settings** to allow a delegated administrator to modify this fields for users in the business unit.
11. In the *AdHoc Settings* pane:
    a. Select **Login by email** to allow users in the business unit to log in using the value of the **Email Contact** field as well as the **Login Name**.
    b. Select the **Allow Login by Email modifying** checkbox to allow administrators to change the **Allow this account to login by email** field when editing or creating an account for this business unit.
    c. Select the **Allow Delivery Method modifying** checkbox to allow administrators to change the delivery method values when editing or creating an account for this business unit.
    d. Select the **Delivery Method** . The value controls the options that ST Web Client displays in the *User Access* window. See *Default package delivery method* on page 90.
    e. The **Implicit Enrollment Type** value controls which option ST Web Client selects initially in the *User Access* window. The choices depend on the enrollment types enabled by the **Delivery Methods** and **Enrollment Types** fields.
    f. Select the **Enrollment Template** for this business unit. When a user is enrolled based on an ad hoc file transfer from a user in this business unit, the selected account template is used. You specify the default enrollment template on the *AdHoc Settings* page.
    g. Select the **Email Notification Template** for this business unit. When a user is enrolled based on an ad hoc file transfer from a user in this business unit, the selected email notification template is used. You specify the *Provide the necessary information as described in the following table:* on page 88 on the *AdHoc Settings* page.
12. In the *Shared Folders Settings* pane, select the **Allow Shared Folders collaboration** checkbox to allow shared folders collaboration.
    This option is inherited from the business unit by the children of the business unit. When checked

user accounts may collaborate using, creating, and sharing folders based on the following criteria:

a. If the user accounts are not in the same BU but they have common ancestor then the business unit setting of the lowest common ancestor is used for deciding if sharing is allowed or not.

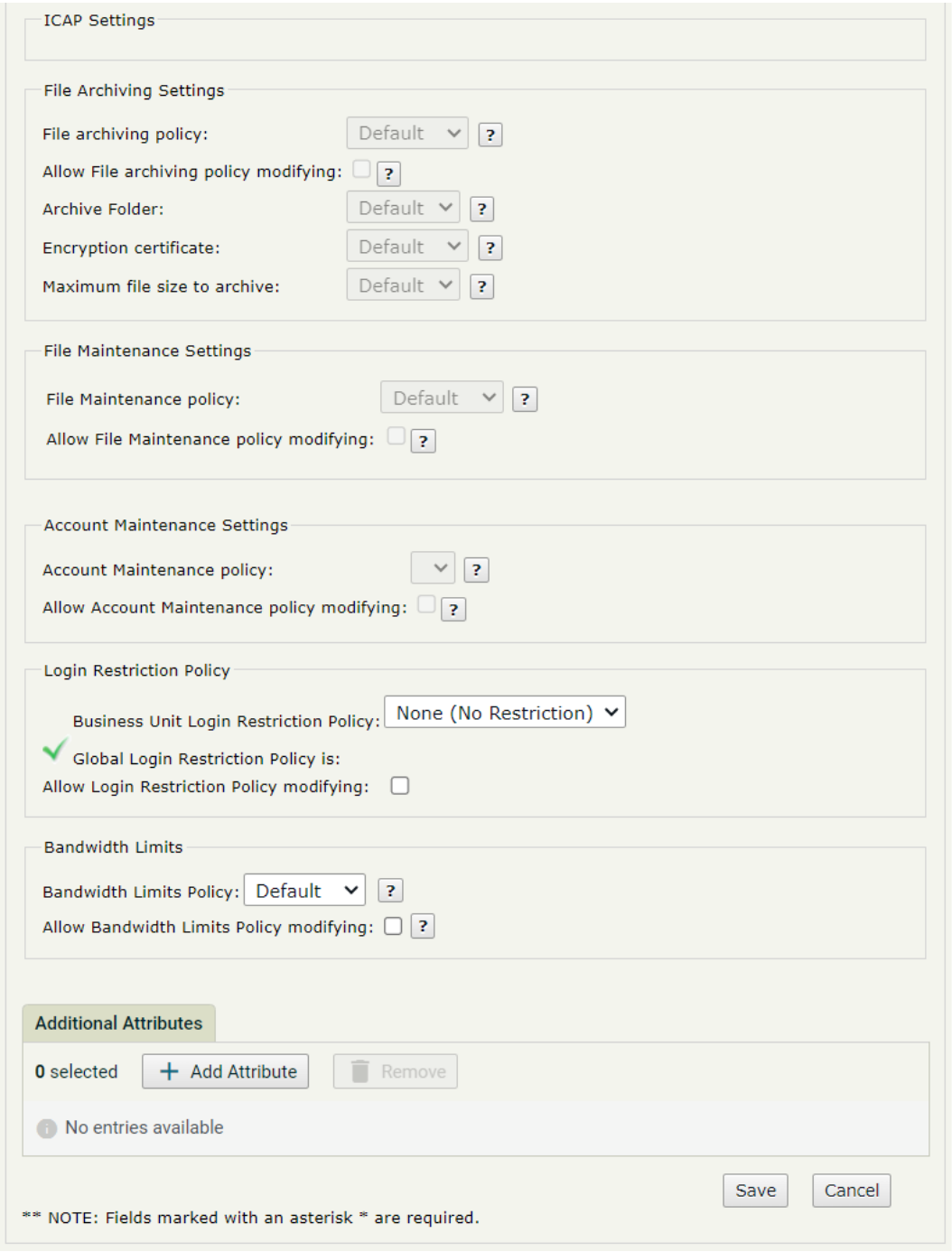
b. If the user accounts are in one and the same business unit then the shared folder setting of the business unit is used for deciding if sharing is allowed or not.

c. If the user accounts have business units assigned but there is no common ancestor then the global setting is used for deciding if sharing is allowed or not. See *View and change server configuration parameters* on page 334 for information on setting global parameters.

d. If the owner account or the collaborator account (or both of them) has no assigned business unit then the global server setting is used for deciding if sharing is allowed or not. See *View and change server configuration parameters* on page 334 for information on setting global parameters.

13. To enable of disable ICAP servers for a specific Business Unit, select **Enable ICAP scan with server 'servername'**, from the list of all ICAP servers in the **ICAP Settings**, and the specified ICAP server will be enabled for this particular Business Unit.
Note: Importing Legacy Business Unit Accounts (from any version before 5.4) will not import the

Business Unit ICAP server selection/s. They must be manually activated after the import.



For ICAP server configuration information, see *ICAP settings* on page 207.

14. In the *File Archiving Settings* pane:
    a. Select the **File archiving policy** from the menu.
        - When **Default** is selected, business unit inherits either its parent's policy or the *File archiving global configuration* on page 226 if it is a top level business unit.
        - When **Enabled** is selected, file archiving will be enabled for all accounts in this business unit.
        - When **Disabled** is selected, file archiving will be disabled for all accounts in this business unit.

            Note This option will be disabled if the **Enable File Archiving** option from the global *File Archiving* page is turned off.

    b. Use **Allow File archiving policy modifying** to enable or disable modification of the File Archiving policy *File archiving policydrop-down list* on page 506.
        - When **checked**, all the accounts that are assigned to this business unit can have their own file archiving policy.
        - When **unchecked**, the corresponding option in account settings page will be disabled and accounts will inherit the business unit policy.

    c. Select the **Archive Folder** from the menu.
        - When **Default** is selected, the business unit inherits its parent's folder. If it's a top level business unit, the global archive maintenance policy applies.
        - When **Custom** is selected, the business unit defines its archive folder.

            Note When you select this option, you must also provide archive folder absolute path. This option will be disabled if the **Enable File Archiving** option from the global *File Archiving* page is turned off.

    d. Select the **Encryption certificate** from the menu. The encryption certificate must be a local x.509 certificate. See *Manage local certificates and certificate signing requests* on page 48.
        - When **Default** is selected, the business unit inherits either its parent's encryption certificate or the global encryption certificate if it is a top level business unit.
        - When **Disabled** is selected, archived files for accounts in this business unit won't be encrypted.
        - When **Custom** is selected, a dedicated encryption certificate can be selected for this business unit.

    Note This option will be disabled if the **Enable File Archiving** option from the global *File Archiving* page is turned off.

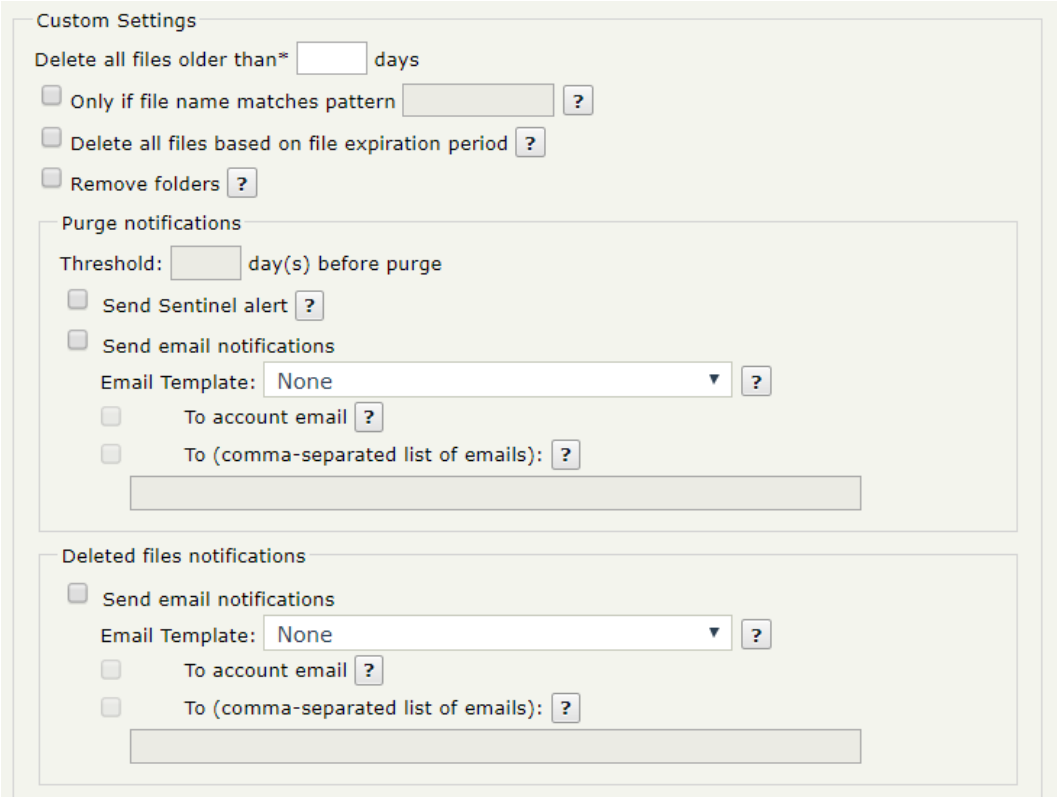
    Note The certificate cannot be deleted or overwritten when it is in use.

    Note When you delete or overwrite a certificate which previously was used for encryption, all files encrypted with this certificate will be useless and cannot be restored.

    e. Select the **Maximum file size to archive** from the menu.

- When **Default** is selected, business unit inherits either its parent's policy or the global file size limit policy if it is a top level business unit.
- When **Custom** is selected, file size limit will be enabled for all accounts from this business unit.
- When **Disabled** is selected, file size limit will be disabled for all accounts from this business unit.

15. (Optional) In the *File Maintenance Settings* pane:
    a. Set the **File Maintenance policy**:
        - When **Default** is selected, the business unit inherits the policy defined at a higher level. A top-level business unit inherits the global policy defined via the File Maintenance application. A child BU inherits the policy of the parent BU.
        - When **Disabled** is selected, File Maintenance will be disabled for this business unit, all accounts under it and all child BUs.
        - When **Custom** is selected, the panel expands with a *Custom settings* pane that allows you to set an individual policy for that business unit. This policy applies to all child BUs and accounts that are directly under that business unit.

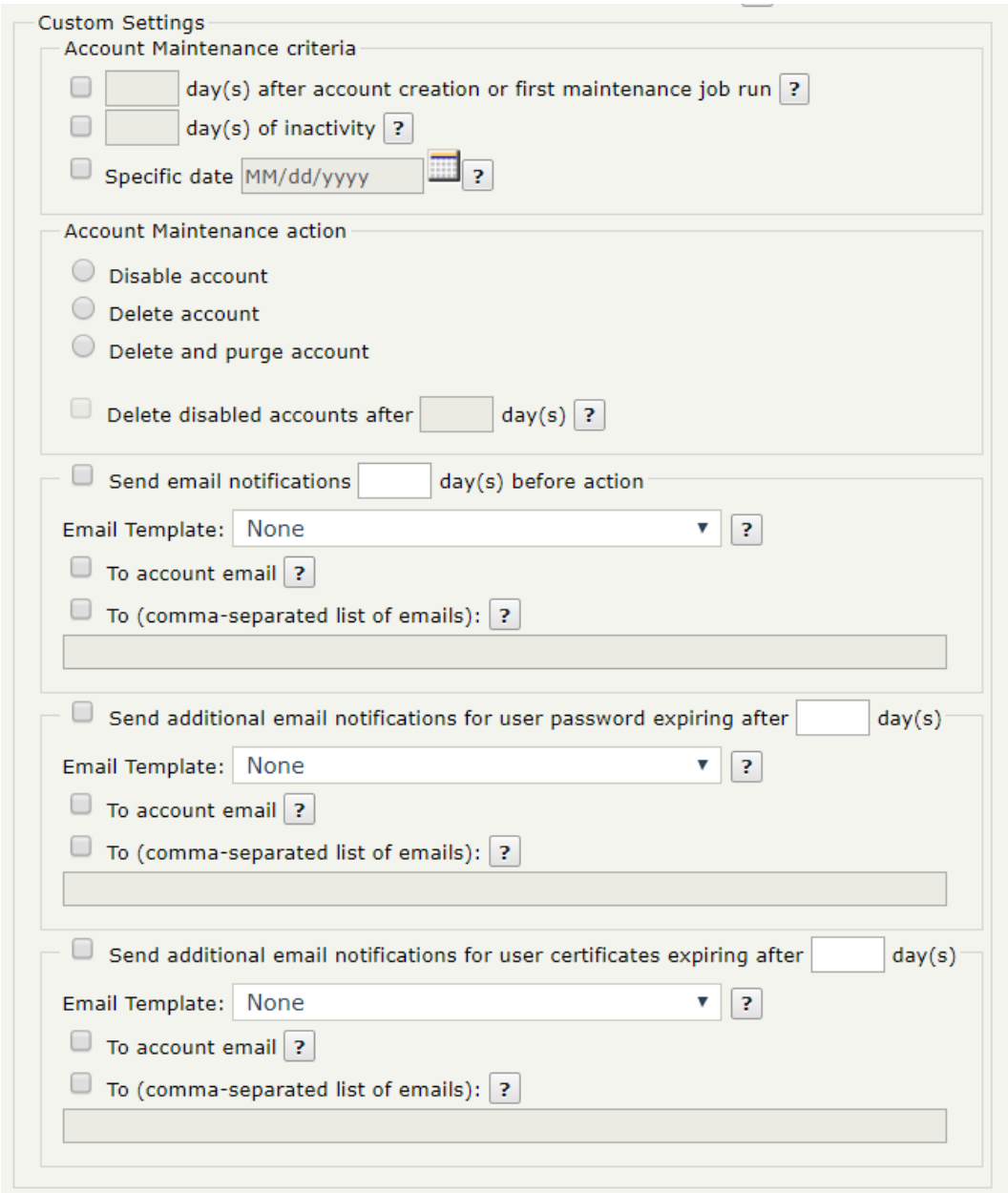


    b. Use the **Allow File Maintenance policy modifying** checkbox to enable the modification of the File Maintenance policy at child BU or *File Maintenance policydrop-down list* on page 506.

- When checked, you can set an individual policy on child business units and accounts that are directly assigned to that business unit.
- When unchecked, the inherited policy cannot be modified at lower levels.

Note The *File Maintenance Settings* are inactive until you create a File Maintenance application.

16. (Optional) In the *Account Maintenance Settings* pane:
    a. Select the **Account Maintenance policy** from the menu.
        - When **Default** is selected, the business unit inherits its parent's policy. In case of a top level business uni, the global Account Maintenance policy applies.
        - When **Disabled** is selected, Account Maintenance is disabled for this business unit.
        - When **Custom** is selected, the panel expands with a **Custom settings** pane that allows you to modify the existing *Account Maintenance application* on page 823. The customized policy applies to accounts in this business unit only. Only at Business Unit level, you can select a specific date on which Account Maintenance will be performed for all accounts under this business unit.



b. Use the **Allow Account Maintenance policy modifying** checkbox to enable or disable modification of the Account Maintenance policy at *Create a user account* on page 503.

- When checked, all the accounts that are assigned to this business unit can have their own Account Maintenance policy.
- When unchecked, the corresponding option in the user account settings page is disabled and the accounts inherit their business unit policy.

Note The *Account Maintenance Settings* will be disabled if a global Account Maintenance policy is not defined.

17. (Optional) When the Address Book feature is enabled, the *Address Book Settings* pane is displayed. To configure the business unit Address Book settings:
    a. Select the Address Book source.
        - **Default** - The business unit inherits either its parent's Address Book policy or the global Address Book policy if it is a top level business unit.
        - **Custom** - A custom Address Book policy configuration will be set for this business unit only and the following will be configurable:
            - Enable or disable Address Book sources for the business unit.
            - Specify the parent groups for Address Book sources.
            - Specify the domain for LDAP Address Book sources.
            - Specify **All Business Units** or **User's own business unit** for local and custom Address Book sources.
        - **Disabled** - The Address Book policy is set to disabled for this business unit.

    b. Specify whether or not to Allow collaboration with non-Address Book recipients. If Address Book functionality is disabled, this setting does not affect user collaboration.
        - When **checked**, accounts that use the Address Book policy defined on the business unit level will be allowed to send email packages and share folders with users that do not exist the defined Address Book.
        - When **unchecked**, accounts that use the Address Book policy defined on the business unit level will be allowed to send email packages and share folders only with users that exist in the defined Address Book.

        This business unit setting overrides the global Address Book policy setting for collaboration. This setting can be overridden on the account level if **Allow modifying of the 'Allow Address Book Collaboration' setting** is checked.

    c. Select **Allow modifying of the Collaboration setting** to enable modifying the **Allow Address Book collaboration** setting at the account level.

    d. Select Allow Address Book source settings to enable modifying of the Allow Address Book Policy modification at the account level. See *Address Book business unit level configuration* on page 248.

18. In the *Login Restriction Policy* pane:
    a. Select the **Business Unit Login Restriction Policy** from menu.

- If **None (No Restriction)** is selected, the Global Login Restriction Policy (if configured) is the default Business Unit Login Restriction Policy.
- If one of the configured Login Restriction Policies is selected, it becomes the default Business Unit Login Restriction Policy.

    b. Select **Allow Login Restriction Policy modifying** to enable modifying of the Login Restriction Policy at the account level.

19. In the *Bandwidth limits* pane:
    a. Select a **Bandwidth Limits Policy** to apply:

- **Default** – the current business unit inherits its bandwidth limits from the parent business unit or the global bandwidth
- **Custom** – the panel expands with two additional options for you to configure: **Inbound limit** and **Outbound limit** (both values in kb/s per user)
- **Disabled** – no bandwidth limits are applied to the users assigned to the current business unit

    b. Select **Allow Bandwidth Limits Policy modifying** and the bandwidth limits on the account template and accounts level will be applicable to the accounts assigned to the current business unit. Deselect this option and bandwidth limits on the account template and accounts level will not override the business unit bandwidth limits.

20. (Optional) Add **Additional attributes** as *attribute:value* pairs. Expression Language is not supported. For details, see *Additional attributes* on page 759.
21. Click **Save**.

Delete a business unit

Use the following procedure to delete a business unit.

1. Before you delete a business unit, make sure that it is not associated with any account, administrator, application, or route and that it does not have any child business units.
2. Select **Accounts > Business Units**. The *Business Units* page is displayed.
3. Using the checkboxes, select the business units to delete. To select or clear all the checkboxes, select or clear the checkbox in the table header.
4. Click **Delete**. A confirmation window is displayed.
5. Click **OK** to delete the selected business units.

Display active users

Users who log in to the SecureTransport client are called *active users*. Use the *Active Users* page to view and search within a list of active users.

1. Select **Accounts > Active Users**.

   The Active Users list is displayed. A line for each user includes the user login name, the last time the user sent an ad hoc file transfer, and the last time the user accessed the server.
2. In the *Search* pane, enter a user login name and click **Search**.

   Users that match your search criteria are displayed.

Note You can get the current number of active users via the REST API.

Additional attributes

Additional attributes in the form of *key:value* pairs are available as custom properties for the following SecureTransport objects: user accounts, service accounts, unlicensed users, account templates, transfer sites, transfer profiles, business units, site templates, route packages, route package templates, login restriction policies, applications, and certificates.

Additional attributes (except for certificates) can be added, edited and deleted through the SecureTransport Administration Tool. The Administrator REST API can also be used to manage attributes for all objects, including certificates.

There are three different types of additional attributes, depending on the object they are added for:

- *attributes that can be used in EL expressions*: for the accounts, account templates, and subscriptions;
- *attributes that can evaluate EL expressions*: for the account templates and subscriptions;
- *attributes that cannot be used in and cannot evaluate EL expressions*: for the transfer sites, transfer profiles, business units, site templates, route package templates, login restriction policies, and applications.

In the SecureTransport Administration Tool, all attributes are defined using the prefix `userVars` (user variables).

Additional attributes for Accounts and Account templates

You can include custom attributes in your email templates. The attributes for accounts and account templates can be used in EL expressions. For more information on how to create, edit and delete them, see User Accounts and Account templates.

After you have defined the additional attributes, you can use them with email notifications. Any *attribute:value* pair can be assigned, for example:

Attribute	Value
userVars.1	internalEmail@axway.com
userVars.2	ReportsMonitor

User variables with email notifications

To use `userVars` for email notifications, you must create your own mail template and specify the desired value with the following expression:

```
$DXAGENT_ACCOUNT_ATTR_USERVARS_{KEY}
```

Where `KEY` is the additional attribute key.

For example, if you want a *user name* defined in the additional attributes, type:

```
$DXAGENT_ACCOUNT_ATTR_USERVARS_NAME
```

Using '.' in key names

If the key of an additional attribute contains period characters (.), you must replace them with underscores (_) when using this attribute in an email template.

For example, to use `userVars.name.first` in an email template, type:

```
$DXAGENT_ACCOUNT_ATTR_USERVARS_NAME_FIRST
```

If there is a collision between additional attribute keys used for email template notifications, the value that they will be evaluated to in an email template is not determined.

For example:

`userVars.name_first` and `userVars.name.first` are both used as:

```
$DXAGENT_ACCOUNT_ATTR_USERVARS_NAME_FIRST
```

It is not definitive which value this expression will be evaluated to.

Usage specifics with other additional attributes

You can also set additional attributes for the following SecureTransport objects:

- *Transfer sites* on page 540
- *Transfer profiles* on page 640
- *Business units* on page 746
- *Manage site templates* on page 737
- Route Package Templates
- Login Restriction Policies
- *Applications* on page 817 (applicable to all application types)

These attributes cannot be used in email templates. They also cannot be used in and cannot evaluate EL expressions, so they cannot be accessed via `userVars`.

Note Additional attributes are set per object and are not inherited from parent objects.

Additional attributes for certificates

You can set additional attributes for the certificates through the SecureTransport Administrator REST API. See the Swagger REST API documentation for more information. These attributes are available for:

- *Certificate Management page* on page 45
- *Manage login certificates* on page 527
- *Manage partner certificates* on page 530
- *Manage private certificates* on page 534

Client-initiated and server-initiated transfers 12

You can use SecureTransport to set up and execute server-initiated transfers. There are four types of transfers:

- **Client-initiated downloads** – A client application "pulls" a file from the SecureTransport Server.
- **Client-initiated uploads** – A client application "pushes" a file to the SecureTransport Server.
- **Server-initiated downloads** – The SecureTransport Server "pulls" a file from a remote server.
- **Server-initiated uploads** – The SecureTransport Server "pushes" a file to a remote server.

Client- and server-initiated transfers can be performed using any supported protocol. The protocol servers that handle client-initiated transfers run on the SecureTransport Server or on the SecureTransport Edge in the perimeter network (DMZ). The protocol clients that perform the server-initiated transfers run on the SecureTransport Server in the Transaction Manager server JVM and can connect out through a SOCKS5 Proxy on a SecureTransport Edge or through an HTTP proxy to a remote system. This allows the protocol clients to have direct access to the file system.

Note If the remote server certificate has a X509v3 Extended Key Usage extension configured, make sure it is set to 'TLS Web Server Authentication'. Server-initiated transfers will always fail if the remote server certificate is configured with a X509v3 Extended Key Usage extension of 'TLS Web Client Authentication'.

Server-initiated transfers can be triggered by any of the following events, depending on the configuration of the transfer:

- A Folder Monitor
- A scheduler
- The arrival of a file

In addition to the protocols mentioned above, the Folder Monitor can be used for inbound and outbound file transfers.

- For outbound transfers, SecureTransport can copy the files to a specified folder.
- For inbound transfers, SecureTransport can monitor the folder for newly arrived files and use the event to trigger an application executing specific tasks.

Any server-initiated transfer requires an account to be subscribed to an application based on one of the application types: Standard Router, Site Mailbox, Shared Folder, Basic Application, File Transfer via File Service Interface, Human to System, or Advanced Routing. For detailed information, see *Applications on page 817*.

Note When setting up a server-initiated outbound transfer, make sure that the target folder exists. SecureTransport does not create the target folder on the remote system automatically.

The following topics describe managing client-initiated and server-initiated transfers:

Client-initiated transfer authentication

The following authentication methods are available to clients connecting to SecureTransport.

Protocol	Authentication method
AS2	• Basic authentication • Client certificate
FTP	• Basic authentication • Basic authentication + client certificate • Username + client certificate
HTTP	• Basic authentication • Basic authentication + client certificate • Username + client certificate • Client certificate • SAML SSO • OAuth2 (via plug-in)
PeSIT	• Basic authentication • Basic authentication + certificate • Username + certificate

Protocol	Authentication method
SSH	• Basic authentication • Basic authentication + SSH key • Username + SSH key

Client certificate and password requirements can be defined:

- globally or per selected user classes, see *Login settings* on page 465.
- on protocol server level, see Protocol servers.

Server-initiated transfer authentication

The following authentication methods are available for connecting to a remote site for the different protocols.

Protocol	Authentication method
AS2	• No authentication • Basic authentication • Client certificate
FTP	• Basic authentication • Basic authentication + client certificate • Username + client certificate
HTTP	• Basic authentication • Basic authentication + client certificate • Username + client certificate • Client certificate
PeSIT	• No authentication • Basic authentication • Basic authentication + certificate • Username + certificate
SSH	• Basic authentication • Basic authentication + SSH key • Username + SSH key

To manage the list of trusted certificates for server authentication, navigate to **Setup > Certificates** in the Administration Tool. For details, see *Certificates* on page 44.

Transfer mode for server-initiated transfers

Transfer mode for server-initiated transfers is determined by protocol and file content type.

For AS2, transfer mode is always binary. For all other protocols, including FTP(S), HTTP(S), and SFTP, SecureTransport uses the content-type of the file name to determine whether a transfer is text (ASCII) or binary (IMAGE).

If the content type is `text`, the file is transferred as text. If the content-type is not text, the file is transferred as binary.

Content-type is determined based on the file's extension. The default mapping for file name extension to content-type is stored in the file, `<FILEDRIVEHOME>/conf/mime.types`. Use the *Server Configuration* page to edit this file to change or add entries.

Server-initiated transfers of multiple files

SecureTransport allows a server-initiated transfer job to process a single file, a set of multiple files specified using wildcards in the file name, or a directory.

Note Except for Folder Monitor type applications, wildcards in directory names and recursive directory traversal are not supported.

A single transfer request for transferring multiple files is defined at the beginning of the transfer process only. If the number of files happens to change during the transfer process, these changes are not reflected and the transfer of these additional/missing files or directories fails.

The SecureTransport system keeps track of the status of all the individual transfers. If a transfer fails, it is rescheduled for a later point in time until the retry limit value is exceeded.

Retry server-initiated transfers

When a server-initiated transfer fails, SecureTransport can automatically retry the transfer. By default, SecureTransport is configured to retry such a transfer five times at two-minute intervals. You can configure the retry count and interval by editing parameters on the *Server Configuration* page:

- `EventQueue.maxRetryCount` – The number of times SecureTransport retries a SIT transfer. It applies to all server-Initiated transfers (inbound and outbound). This option does not apply to Advanced Routing except for the Pull From Partner step. The default value is `5`.
- `EventQueue.retryDelayInterval` – The time in seconds that SecureTransport waits after a transfer fails before retrying it. The default value is `120`.

- `EventQueue.internalRetryDelayInterval` – The time in seconds that SecureTransport waits when a transfer cannot be started (for example, because all outbound connections to an FTP server are in use) before retrying it. The default value is `120`.

Note The CycleId of the original transfer is preserved when retrying or resubmitting failed server-initiated transfers. For more information, see *CycleId* on page 162.

Server-initiated transfers can also be resubmitted manually, see *Resubmitted and retried transfers* on page 306.

Proxy server-initiated connections

You can proxy server-initiated connections using a SOCKS5 proxy running on a SecureTransport Edge or an HTTP proxy. You configure the use of a proxy by defining one or more network zones and making a selection in the transfer site. See *Manage the communication across Transaction Manager, protocol and proxy servers* on page 230 and the procedures for defining AS2, FTP, HTTP, PeSIT, and SSH transfer sites.

Note By default, when a proxy is configured, direct connections from the SecureTransport Server are not permitted even when the proxy is unreachable. To change the default behavior, set the `Direct.Connection.When.Proxy.Down` server configuration option to **true**. For more information, see *View and change server configuration parameters* on page 334.

If server-initiated transfers performed using FTP(S) are passing through the SOCKS5 proxy, increase the value of the `Socks.Idle.Timeout` server configuration option on the SecureTransport Edge from `600000` to `7200000` milliseconds. This prevents the FTP control connection from timing out during the transfer. You must restart the SOCKS5 proxy server for this change to take effect.

Repository encryption and server-initiated transfers

When SecureTransport performs server-initiated transfers, it defines user classes by UID and GID only – no user name is specified. As a result, if you use `EncryptClass` for encrypting or decrypting transferred files and `EncryptClass` is defined only by user name, the following is true:

- Encrypted files transferred using a server-initiated upload are decrypted before the start of the transfer.
- Files transferred using a server-initiated download are transferred with encryption.

Server-initiated transfer limitations

SecureTransport has the following limitations for server-initiated transfers:

- SecureTransport Server supports server-initiated transfers over HTTP only to remote sites running on another SecureTransport Server.

- Server-initiated transfers over FTPS from a streaming configuration with SecureTransport Edge to remote sites support only passive connection mode.
- When performing server-initiated uploads using the SSH protocol SecureTransport cannot always identify the remote operating system when the remote SSH server has version 3 or less.

  For ASCII mode SSH transfers, if the remote SSH server supports the newline (newline@vandyke.com) extension, SecureTransport correctly converts the end-of-line characters of the file.

  If the remote SSH server does not support the newline extension, SecureTransport can be configured to convert the end-of-line characters during server-initiated uploads based on the value of server configuration parameters. If the value of the `Ssh.EndOfLineConversion.enabled` server configuration parameter is true, SecureTransport uses the value of the `Ssh.EndOfLineConversion.type` server configuration parameter as the end-of-line sequence. Valid values are: `0x0A` (LF), `0x0D` (CR), and `0x0D0A` (CRLF). By default, the value of `Ssh.EndOfLineConversion.enabled` is `false` and the value of `Ssh.EndOfLineConversion.type` is `0x0D0A`.

  Whether or not the remote SSH server supports the newline extension, the remote server sees these transfers as BINARY mode.

  In all other cases of ASCII mode SSH server-initiated uploads and downloads, SecureTransport cannot identify the correct end-of-line characters to use. SecureTransport performs these transfers in BINARY mode and indicates this on the *File Tracking* page.
- The name of a file processed by a Folder Monitor transfer site cannot contain two or more of the following characters in sequence: `< > | : ? " * / \ % [ ] ~` (at the beginning of the file only).

Access 13

Use the Axway SecureTransport **Access** menu to configure how SecureTransport performs access control. Access control defines and restricts the rights of individuals to obtain data from, or place data onto, a storage device. Also, access control defines and restricts the rights of individuals to login. Access control changes are copied to all SecureTransport Servers in your Enterprise Cluster (EC).

Note If your SecureTransport deployment includes SecureTransport Edge servers in a peripheral network (DMZ), you must configure all the settings under the **Access** menu that are applicable to SecureTransport Edge exactly the same on both SecureTransport Edge and SecureTransport Servers.

The following topics describe the various methods to control access to SecureTransport:

- *User classes* on page 771 - Describes user classes.
- *Secure Socket Layer access* on page 780 - Describes Secure Socket Layer access.
- *Virtual groups* on page 783 - Describes virtual groups.
- *Filesystem restrictions* on page 785 - Lists the filesystem restrictions.
- *Upload restrictions* on page 789 - Describes the upload restrictions.
- *Download restrictions* on page 795 - Describes the download restrictions.
- *FTP command restrictions* on page 799 - Describes the FTP command restrictions.
- *Control access to Administration Tool and protocol servers* on page 801 - Describes the protocol server access control.
- *User limits* on page 806 - Describes user limits and how to use user limits.
- *User and group access* on page 809 - Describes user and group access.
- *Login restrictions* on page 810 - Describes user login restrictions.

Pluggable authorization

SecureTransport Pluggable Authorization feature provides the option to add custom authorization logic by plugging it to the system. Existing SecureTransport Access Restrictions will be executed after any custom authorization logic. The FTP protocol is an exception, where the internal restrictions will be applied before the custom logic. Custom authorization will be applied for all protocols on client-initiated transfers.

SecureTransport will be executing any custom authorization on the following operations:

- Upload a file
- Download a file
- List content of a directory

- Change permissions (file or directory)
- Rename a file
- Delete a file
- Create a directory
- Delete a directory

The custom authorization attempt can be either successful or unsuccessful.

In case of success, SecureTransport will continue executing the set of applied Access Restrictions (if any). In case of authorization failure, the operation will not be executed.

Pluggable authorization also supports file filtering capabilities. All plug-in implementations are able to use SecureTransport specific environment data described in the Developer's Guide.

Note A collection of authorization plug-ins maintained by Axway is available in the Amplify Repository.

Before your custom plug-in can be configured and used, it must be deployed, registered, and then enabled in the Server Configuration.

Plug-in deployment

The custom authorization logic (plug-in) is packaged as .jar file that follows the set of conventions described in the Developer's Guide.

To deploy an authorization plug-in, place its JAR file in the `/<st_dir>/plugins/authorization/` directory, and restart the Admin and the TM daemons.

In a cluster environment, the plug-in should be deployed on all nodes before they are added to the cluster, and the Admin and Transaction Manager services should be restarted on all nodes.

Note The plug-in is applicable only on SecureTransport Server installation.

Plug-in registration

SecureTransport identifies the plug-in by the name of its JAR file. Plug-ins are discovered and registered at the Admin daemon start. Each authorization plug-in is added to the following configuration registry in the *Server configuration* page:

```
Plugins.Authorization.Registry
```

If the plug-in has a custom configuration, it is also added to the server configuration for the end users in the following format:

```
Plugins.Authorization.<plugin_name>.<config_option>
```

Note The plug-in configuration options are exported upon server configuration export. Before importing a server configuration with custom plug-in configuration options, the relative plug-ins must be deployed. Otherwise, their configuration options will not be imported.

Plug-in activation

After being registered, the authorization plug-ins are added to the Server Configuration, but they are disabled (have a hash symbol in front of their names). SecureTransport will not automatically activate a newly registered plug-in. To activate a plug-in, remove the # symbol from its name.

Only one authorization plug-in can be enabled at a time.

Note Plug-in activation does not require service restart.

Plug-in management

Undeploy a plug-in

1. Delete the JAR file from the `/<st_dir>/plugins/authorization/` directory.

2. Restart the Admin and TM daemons.
The plug-in name is then removed from the registry along with its configuration options.

When you uninstall SecureTransport, the plug-ins JAR files are also removed.

Redeploy or update a plug-in

1. Undeploy the existing plug-in.

2. After the Admin and TM daemon restart, go to the Server Configuration registry and make sure the plug-in is not present.

3. Deploy the new plug-in (version).

After the restart, the new plug-in is added to the authorization plug-in list.

Plug-in configuration

Successful plug-in usage depends on the plug-in implementation (check the Developer's guide for more details).

If you set up a Standard Cluster, and the steps in the Plug-in deployment section are not accomplished, this will not be considered as a correct configuration.

For example, in a Standard Cluster, if the jar file is not uploaded to the secondary node, the configuration will not be considered correct, and an error message will be displayed in the Server Log at startup.

Plug-in authorization notifications

On each of the operations authorized by a plug-in, the following messages are displayed in the Server log:

- On an INFO level: "Custom authorization plug-in <plugin_name> execution for <type_of_ operation> operation finished and it took: <estimated_execution_time> ms.”
- Custom authorization result:
    - On success, on a DEBUG level: "<plugin_name>'s with class <plugin_authorizer_impl> authorization successful. Result is <result_exitcode>, '<result_message>'"
    - On failure, on an INFO level: "<plugin_name>'s with class <plugin_authorizer_impl> authorization failed. Result is <result_exitcode>, '<result_message>'"
- On an Error level, when exception is thrown from the executing of any of the plug-in.

Note All data sent/ received to/from a plug-in will be available on a DEBUG log level.

Plug-in authorization considerations and special cases

The following considerations must be taken into account:

- Custom authorization plug-ins will be executed for client-initiated transfers only.
- Authorization will be executed for all protocols on any supported and applicable operation. Therefore, some protocols may not support particular operations (For example, directory listing) and the plug-in’s implementation will not be executed.
- For pluggable authorization, server-initiated transfers are out of the scope. However, if performing such transfers on one host only with a deployed plug-in (or in two hosts which both have plug-ins deployed), half of the transfer will be authorized against the plug-in. For one of the parties such transfer always appears (and it is) as a true client-initiated transfer since the SecureTransport protocol daemons receive a remote connections and perform user operations as it is in an ordinary client upload/download.
- Custom authorization will be executed before any internal SecureTransport upload/download/filesystem restrictions. An exception is the FTP protocol, where the internal SecureTransport restrictions are evaluated before the custom authorization.
- Advanced Routing transformation steps will not trigger custom authorization.
- Publish To Account routing step is excluded from custom authorization.
- Send To Partner routing step will be authorized in the receiving party if any custom plug-ins are deployed.

Plug-in file filtering capabilities

In addition, custom authorization supports plugging an implementation for filtering of directory content. Use this feature to restrict the view of some files for a particular user. Refer to the Developer's guide for more information regarding the file filtering extension.

Note Filtering is not applied for the directory content in **Mailbox** section in SecureTransport Web Client.

User classes

User classes define sets of SecureTransport users who share characteristics and privileges.

Use user classes to define the following access restrictions:

- SSL encryption
- Filesystem, upload and download
- FTP command
- Server
- User limits

You might also define user classes to support the following SecureTransport functions:

- The *Server Usage Monitor* page reports usage information by user class.
- On the *Setting* page, you define a FTP passive mode address rules for a user class.
- On the *Command Logging* page, you enable logging for a user class.
- On the *Transfer Logging* page, you enable logging for a user class.
- When you create an LDAP domain, you can define a DN filter for a user class.
- On the LDAP *Home Folders* page, you define the home folder prefix for a user class.
- When you create an account template, you specify the user class SecureTransport applies it to.
- You define a user class named `EncryptClass` to enable repository encryption for users.

To determine the user class for a user, SecureTransport evaluates the criteria for each user class in the sequence and puts the user in the first class that matches.

User classes are defined by the following values:

- **User type** – The user type can be *real*, *virtual*, or either
- **User name** – The user's login name
- **User group** – The account group for the user
- **From address** – The IP address or host name from which the user connects
- **Custom expression** – An expression comprised of values of SecureTransport user attributes and LDAP attributes as well as SSO attributes, constant values and patterns using arithmetic, comparison, string matching, logical, and conditional operations

You can use wildcard characters to define patterns in the **User name**, **User group**, and **From address** fields. Question mark (`?`) matches one character and asterisk (`*`) matches any string of characters.

Configure user classes on SecureTransport Server only.

The following topics describe and list the default user classes and custom expressions. The also provide how-to instructions for managing user classes.

- *Default user classes* on page 772 - Describes and lists the default user classes.
- *Custom expressions* on page 772 - Lists the user class custom expressions.
- *Manage user classes* on page 776 - Provides how-to instructions for managing user classes.

Default user classes

SecureTransport provides the following default user classes:

- **RealClass** – all users of type real, connecting from any host address (`*`)
- **VirtClass** – all users of type virtual, connecting from any host address (`*`)

If not other user classes are defined, all users are in one of the default user classes. You can use these user classes to create access and security settings. For example, you can prevent virtual users from uploading documents to the server. To define more specific access and security settings more specific sets of users, create custom user classes.

Note Classes that match Single Sign-On (SSO) accounts cannot be used with users with type Real.

Custom expressions

You can use the **Custom expression** field to define a user class based on the values of any SecureTransport user attributes and LDAP attributes, including custom ones. This subsection describes the variables, constants and functions for user class specific usage of Expression Language.

User attributes in a user class

The following user attributes are supported:

- `fdxUid` – User ID (UNIX-based systems only)
- `fdxGid` – Group ID
- `fdxHomeDir` – Home folder
- `fdxUserType` – User type
- `fdxShell` – User shell (UNIX-based systems only)
- `fdxSysUser` – Name of a local or domain user of the Windows server whose credentials SecureTransport uses to access the Windows files in the session (Windows only)

- Any custom SecureTransport user attribute defined in the LDAP domain. See *Define attribute mappings for a domain on page 490*.

LDAP specific attributes

The following variables represent values from the SecureTransport LDAP domain that are supported:

- `LDAP_DOMAIN_ID` – Internal ID
- `LDAP_DOMAIN_NAME` – Value of the **Domain Name** field
- `LDAP_DN` – Value of the **Base DN** field
- `LDAP_AUTH_BY_EMAIL` – Value of the **Login by Email** field, `0` for `Disabled`, `1` for `Enabled`

SSO specific attributes

The following variables represent SSO values for SecureTransport that are supported:

- `SSO.idpId` – Identity provider Identification
- `SSO.email` – SSO user email
- `SSO.uid` – User ID of the SSO user
- `SSO.gid` – Group ID of the SSO user
- `SSO.tenant` – SSO tenant
- `SSO.homeDir` – Home directory of the SSO user
- `SSO.userName` – SSO user username

Note `UID`, `GID`, `Email` and `homeDir` SSO attributes should be mapped to SecureTransport as `fdxUid`, `fdxGid`, `fdxEmail`, `fdxHomeDir` attributes respectively.

Authenticated user specific attributes

The following variables that represent values from an already authenticated user in SecureTransport are supported:

- `DXAGENT_USERGID` - Group ID of the user
- `DXAGENT_USERUID` - User ID of the user
- `DXAGENT_USEREMAIL` - user email
- `DXAGENT_BUSINESS_UNIT_NAME` - Business unit name
- `DXAGENT_ACCOUNT_NAME` - Account name
- `DXAGENT_ACCOUNT_ID` - Account ID
- `DXAGENT_USERLOGINTYPE` - Account login type

- DXAGENT_ACCOUNT_PHONE - Account telephone number
- DXAGENT_ACCOUNT_NOTES - Account notes
- DXAGENT_ACCOUNT_UNLICENSED - Account licensed property
- DXAGENT_ACCOUNT_TYPE - Account type: Unspecified, Internal, Partner
- DXAGENT_ACCOUNT_DELIVERY_METHOD - Adhoc delivery method
- DXAGENT_ACCOUNT_IMPLICIT_ENROLLMENT - Account implicit enrollment type
- DXAGENT_HOMEDIR - Account home directory
- DXAGENT_ACCOUNT_ATTR_USERVARS_*keyname* - Account's additional attributes; the value should be surrounded by double quotes " "

Note The attributes are not supported for users instantiated from account templates.

Note DXAGENT variables will not be evaluated for PeSIT, AS2 and Publish to Account transfers, as actual user login is not performed.

Map a user based on Login type

SecureTransport allows you to map a user based on their login type:

- To map a real user, use DXAGENT_USERLOGINTYPE="REAL"
- To map a virtual user, use DXAGENT_USERLOGINTYPE="VIRTUAL"
- To map a Siteminder user, use DXAGENT_USERLOGINTYPE="SITEMINDER"
- To map an SSO user, use DXAGENT_USERLOGINTYPE="SSO"
- To map an LDAP user, use DXAGENT_USERLOGINTYPE="LDAP"

Supported constants

The following constants are supported:

- Numeric constants: -5, 100, .5, 1.05, 3.14159D, 6.0221415e23, 214748364, 0xFFECDE5E
- Character constants: 'a','\u0061','\t','\u0009','\n','\b','\r','\f', '\\','\"'
- String constants: "Finance", "US", "^.*@finance\.example\.com$"
- Logical constants: true, false
- Null constant: null (represents no value, so fdxShell = null is true if fdxShell is not defined)

Supported functions

The following functions are supported:

- `isSet("A")` – true if there is a session variable named `A`
- `toInt(A)` – converts `A` to an integer
- `toString(A)` – converts `A` to a string
- `memberOf("A", B$collection)` – true if `A` is a member of the multivalued session variable `B`. Note that the value in `A` is case-sensitive. Using `$collection` is optional:
  - Without `$collection`, the value in `A` is matched only against the last value of the exported LDAP attribute.
  - With `$collection`, the value in `A` is matched against all values of the exported LDAP attribute.

  Example:

  `memberOf("person", LDAP_DIR_objectClass$collection)` – returns `true` if `person` is among the values of the LDAP attribute `objectClass`

  For more information and examples, see the "Additional User Class custom expressions" section in KB 180877 (login required).

Supported operators

SecureTransport evaluates the expression based on the following operator precedence from highest to lowest:

- Logical unary `not`
- Arithmetic unary + and −
- Arithmetic binary `*`, `/`, and `%` (integer remainder)
- Arithmetic binary + and −
- String concatenation +
- Numeric, date and string comparison >, >=, <, <=, and `like`
- Logical, numeric, date, and string comparison = and <>
- Logical `and`
- Logical `or`
- Conditional expression `A ? B : C` (which has the value `B` if `A` is `true` or `C` if `A` is not `true`)

Use parentheses to group expressions and override the operator precedence.

SecureTransport dynamically converts numeric expressions to long integers, single-precision real numbers, or double-precision real numbers when it is necessary to evaluate an operator. When an operator requires a logical value, SecureTransport converts any value of a type other than logical to `false`.

The `like` operator matches its string left operand against a string right operand that is a Java regular expression. The result is `true` if the regular expression matches all of the left operand. The backslash (`\`) is the escape character Java regular expressions, so, in a regular expression, use two backslashes (`\\`) to match a backslash. See the examples.

Example usage

The following expression checks for virtual users who are in one of three groups:

```
fdxUserType = "virtual" and (fdxGid = 1200 or fxdGid = 1400 or
fdxGid = 1500)
```

The following expression tests the prefix of the user home directory on a Windows system:

```
fdxHomeDir like "C:\\home\\users\\finance\\.*"
```

The following two expressions return the same result, checking the email address against different regular expressions depending on the UID:

```
fdxUid > 100 and fdxUid <= 200 and fdxEmail like
".*@finance\.example\.com" or fdxEmail like ".*@hr\.example\.com"
```

```
fdxEmail like (fdxUid > 100 and fdxUid <= 200 ?
".*@finance\.example\.com" : ".*@hr\.example\.com")
```

Manage user classes

Use the *User Classes* page to add, enable, disable, reorder, and delete user classes.

The following topics provide user class examples and how-to instructions for managing user classes:

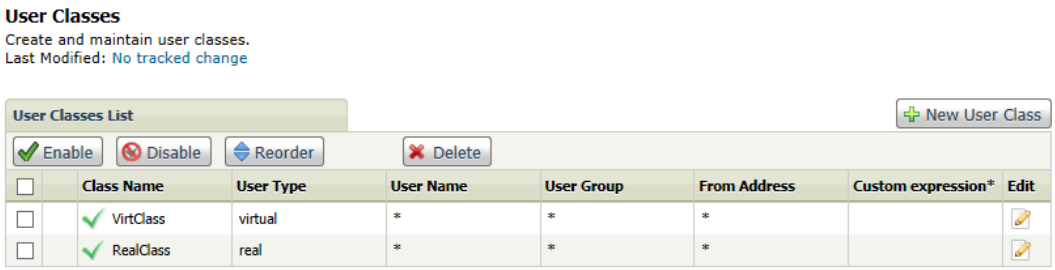
- *Add a user class* on page 776
- *Enable or disable a user class* on page 778
- *Edit a user class* on page 778
- *Reorder user classes* on page 778
- *Delete a user class* on page 778
- *User class examples* on page 779

Add a user class

Use the following procedure to add a user class.

1. Select **Access > User Classes**.

   The *User Classes* page is displayed.



2. Click **New User Class**. A new line is displayed in the *User Classes List*.
3. In the **Class Name** field, enter the name for the user class to create.

   If the name is not unique, SecureTransport uses only the first user class with that name in the *User Class List*.
4. In the **User Type** field, select the predefined user type for the user class.

   Note Because of the different ways SecureTransport treats the path name specification of the download or upload directory for real and virtual users when download or upload restrictions are defined, you should avoid selecting `*` to match all users.
5. In the **User Name** field, enter one of the following:
   - The user name, such as the UNIX-based system login name, the Windows user name, virtual user name, LDAP user name, SiteMinder, or Single Sign-On (SSO) user name.

     On Windows, type either a `username`, *`COMPUTERNAME`*`\username`, or *`DOMAIN`*`\username`.
   - A pattern using `*` and `?` to include matching users. For example, `*` includes all users.

     Only one pattern is allowed.
6. In the **User Group** field, enter one of the following:
   - The name or numerical GID of the group assigned to the user. If all characters are numeric, the value is a GID. Otherwise, it is group name. On Windows, the value can be either the Windows security identifier (SID) of the group or the GID from the `group` file.
   - An asterisk (`*`) to include users in all groups.
7. In the **From Address** field, enter a host name, a host name pattern, an IP address, or subnet specification. For valid values, see *IP addresses and host names* on page 1101.

   Only one host name, an IP address, or subnet specification is allowed.
8. To define the user class using other user attributes or LDAP attributes, enter a **Custom expression**. See *Custom expressions* on page 772.
9. Click the Save icon () in the **Edit** column.

   The status of a new user class is set to Disabled.

Note To cancel an add operation, select **Access > User Classes** again.

Enable or disable a user class

Use the following procedure to enable or disable a user class.

1. Select **Access > User Classes**.

   The *User Classes* page is displayed.
2. In the *User Classes List*, select the checkbox for each user class to modify.
3. Click **Enable** or **Disable**.

   The icons in the **Class Name** column change to indicate the status of the classes.

Edit a user class

Use the following procedure to edit a user class.

1. Select **Access > User Classes**.

   The *User Classes* page is displayed.
2. In the *User Classes List*, click the Edit icon ( ) in the **Edit** column for the user class entry to edit.
3. Make the required changes to the fields in the row.
4. Click the Save icon ( ) in the **Edit** column.

Note To cancel an edit operation, select **Access > User Classes** again.

Reorder user classes

If a user belongs to multiple classes, SecureTransport categorizes the user as belonging to the first matching class in the *User Classes List*.

If two or more user classes have the same name, SecureTransport processes only the first of those classes in the *User Classes List*.

1. Select **Access > User Classes**.

   The *User Classes* page is displayed.
2. In the *User Classes List*, click **Reorder**.

   Up and down arrows are displayed in a column before the **Class Name** column in the *User Classes List*.
3. Drag the rows of the *User Classes List* to the required order.
4. Click **Save Order**.

Note To cancel a reorder operation, select **Access > User Classes** again.

Delete a user class

Use the following procedure to delete a user class.

1. Select **Access > User Classes**.

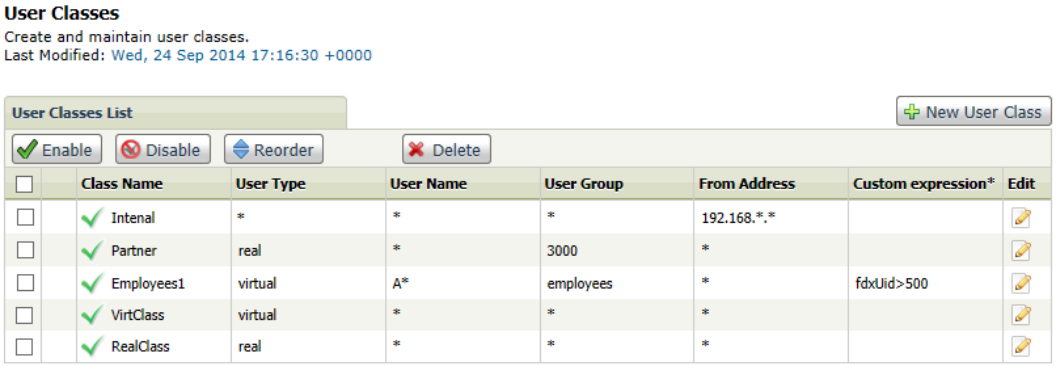
   The *User Classes* page is displayed.

2. In the *User Classes List*, select the checkbox for each user class to delete.
3. Click **Delete**.
4. Click **OK** in the confirmation dialog box.

Note If you delete a user class, it is best to remove all references to that user class from all access rules. SecureTransport ignores access rules that reference an undefined user class.

User class examples

The following example illustrates some sample user class entries.



The following table summarizes the user classes and describes their functions.

User class	Definition
Internal	Includes users of any type, name, or group, who connect from IP address that start with `192.168`.
Partner	Includes users of real type with GID 3000 who do not connect from IP address that start with `192.168`.
Employees1	Includes users of virtual type whose user name begins with `A`, are in the `employees` user group, have user ID greater than or equal to 500, and do not fall into the Internal class.
VirtClass	Includes all virtual users who do not fall into the Internal or Employee1 classes.
RealClass	Includes all real users who do not fall into the Partner class.

Because the default RealClass and VirtClass include all users, all SecureTransport users are in one of the four classes.

Secure Socket Layer access

The Secure Socket Layer (SSL) is the security protocol used by SecureTransport to encrypt communication between the server and its clients. SSL requires the server to have a certificate, which is exchanged with the client during the SSL handshake. SSL allows the client to have a certificate that is presented to the server and can be used to authenticate the SecureTransport user as an alternative to authenticating the user through a password. SSL is also used by SecureTransport to transfer files securely.

Based on user class, encryption (SSL) can be set as optional or mandatory.

- If SSL is mandatory, the SecureTransport Server only accepts SSL connections. If SSL is not enabled at the client end, SecureTransport refuses the connection.
- If SSL is optional, then both SSL and non-SSL connections are enabled. If the client requests an SSL connection, then it is negotiated. Otherwise, SecureTransport accepts the connection to proceed without encryption. If the client certificate verification is enabled, SecureTransport checks the validity and authenticity of the certificate presented by the client. If SSL is optional and the client requests SSL, but the client certificate verification fails, the client is allowed to log in with a user name and password.

If your SecureTransport deployment includes SecureTransport Edge servers in a peripheral network (DMZ), you should configure SSL access exactly the same on both SecureTransport Edge and SecureTransport Server.

The following topics describe SSL and SSH authentication and provide how-to instructions for managing SSL access:

- *SSL and SSH* on page 780 - Describe SSL and SSH authentication.
- *Manage SSL access* on page 781 - Provides how-to instructions for managing SSL access.

SSL and SSH

SSH provides mutual authentication. The client authenticates the server and the server authenticates the client. The data transferred between the client and server is encrypted.

For SSH server authentication, a key is assigned to the SSH server. As part of the connection handshake, the SSH client verifies the server key by checking whether the user has successfully connected to the server in the past. If a user is connecting to the server for the first time, the SSH client asks the user to confirm that the SSH server key and accept it before connecting to the server.

Generally, the SSH protocol provides three methods of authenticating clients: keyboard-interactive authentication, password authentication, and public key authentication. All types of client authentication are supported by SecureTransport.

Note SSH authentication is based on the public key, while SSL authentication is based on certificates. A certificate includes a public key, but it also includes information about the entity to which the key belongs.

In SecureTransport, keys are always managed in the form of certificates. Server keys are associated with Local Certificates. For details, see *Manage local certificates and certificate signing requests* on page 48.

You can assign a local server certificate to the SSH server. The key contained in the certificate is used to establish the SSH connection. Similarly SSH client keys are associated with login certificates. For details, see *Certificate types* on page 44.

The Secure Socket Layer configuration includes an option to control the use of client certificates in SSL. This option also applies to the use of SSH client keys as described in *Manage SSL access* on page 781.

Note If SecureTransport is deployed with SecureTransport Edge Server installed in a peripheral network (DMZ), you must configure the SSL access control settings on both the SecureTransport Edge and the SecureTransport Server. The settings can be the same to require the same secure connection for both types of installation or they might differ. For example, HTTP and FTP servers on the SecureTransport Server might be intended for internal use only and be protected by the firewall. In this case, the SecureTransport Server can be set up to not require SSL, depending on the organization's policy.

Manage SSL access

Use the *Secure Socket Layer* page to add, enable, disable, reorder, or delete SSL encryption entries.

Note To enable or disable SSL for HTTP transfers, you must modify the **HTTP Server** settings on the *Server Control* page. If you select **Enable HTTPS**, SecureTransport uses SSL with the HTTP connections. If you do not select **Enable HTTPS**, SecureTransport does not use SSL with HTTP connections. For more information, see Manage the HTTP server.

The following topics provide how-to instructions for managing SSL access:

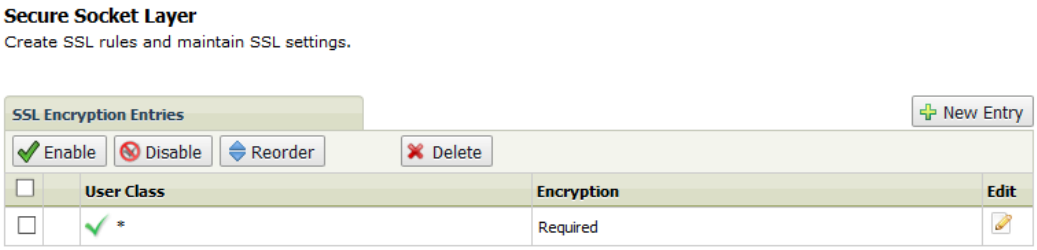
- *Add an SSL users encryption entry* on page 781
- *Enable or disable an encryption entry* on page 782
- *Edit an encryption entry* on page 782
- *Reorder encryption entries* on page 783
- *Delete an encryption entry* on page 783

Add an SSL users encryption entry

You can define SSL encryption settings based on user classes.

1. Select **Access > Secure Socket Layer**.

   The *Secure Socket Layer* page is displayed.



2. At *SSL Encryption Entries*, click **New Entry**. A new line is displayed in the *User Classes List*.
3. Select a **User Class**. The user class must already be defined in the *User Classes* page. Asterisk (`*`) means all users.
4. Select an **Encryption** option for the user class. The two option types are: `Required` and `Optional`.
5. Click the Save icon ( ) in the **Edit** column.

   Note When using certificate-based authentication on Windows with real users, add the real user to the password vault to log in without being prompted for a password. For more information, see *Real users* on page 742.

Note To cancel an add operation, select **Access > Secure Socket Layer** again.

Enable or disable an encryption entry

Use the following procedure to enable or disable an encryption entry.

1. Select **Access > Secure Socket Layer**.

   The *Secure Socket Layer* page is displayed.
2. Under *SSL Encryption Entries*, select the checkbox for each entry to modify.
3. Click **Enable** or **Disable**.

   The icons in the **User Class** column change to indicate the status of the entries.

Edit an encryption entry

Use the following procedure to edit an encryption entry.

1. Select **Access > Secure Socket Layer**.

   The *Secure Socket Layer* page is displayed.
2. Under *SSL Encryption Entries*, click the Edit icon ( ) in the **Edit** column for the entry to edit.
3. Make the required changes to the fields in the entry.
4. Click the Save icon ( ) in the **Edit** column.

Note To cancel an edit operation, select **Access > Secure Socket Layer** again.

Reorder encryption entries

SecureTransport applies the first entry in the *SSL Encryption Entries* that matched the user class of the user. So, for SecureTransport to use them, you want the more specific entries before the more general entries in the list.

1. Select **Access > Secure Socket Layer**.

   The *Secure Socket Layer* page is displayed.

2. Under *SSL Encryption Entries*, click **Reorder**.

   Up and down arrows are displayed in a column before the **User Class** column in the *SSL Encryption Entries*.

3. Drag the rows of the *SSL Encryption Entries* to the required order.
4. Click **Save Order**.

Note To cancel a reorder operation, select **Access > Secure Socket Layer** again.

Delete an encryption entry

Use the following procedure to delete an encryption entry.

1. Select **Access > Secure Socket Layer**.

   The *Secure Socket Layer* page is displayed.

2. Under *SSL Encryption Entries*, select the checkbox for each entry to delete.
3. Click **Delete**.
4. Click **OK** in the confirmation dialog box.

Virtual groups

Every user on a UNIX-based system has a user ID and a group ID. UNIX-based systems use these IDs to set process and file permissions. You can define certain groups as virtual users. Any user who is a member of a virtual group becomes a virtual user whether or not they are included in the virtual password file.

Configure virtual groups on SecureTransport Server only.

The following topic describes how to manage virtual groups:

- *Manage virtual groups* on page 783 - Provides how-to instructions for managing virtual groups.

Manage virtual groups

Use the *Virtual Groups* page to add, enable, disable, or delete SSL virtual group entries.

The following topics provide how-to instructions for managing virtual groups:

Add a virtual group

Use the following procedure to add a virtual group.

1. Select **Access > Virtual Groups**.

   The *Virtual Groups* page is displayed.

2. At *Virtual Groups List*, click **New Entry**. A new line is displayed in the *Virtual Groups List*.

3. In the **Virtual Group Name** field, enter the name or the group ID (GID) of the group.
4. Click the Save icon ( ) in the **Edit** column.

   The status of a new virtual group is set to Disabled.

Note To cancel an add operation, select **Access > Virtual Groups** again.

Enable or disable a virtual group

Use the following procedure to enable or disable a virtual group.

1. Select **Access > Virtual Groups**.

   The *Virtual Groups* page is displayed.

2. In the *Virtual Groups List*, select the checkbox for each user class to modify.
3. Click **Enable** or **Disable**.

   The icons in the **Virtual Group Name** column change to indicate the status of the classes.

Edit a virtual group

Use the following procedure to edit a virtual group.

1. Select **Access > Virtual Groups**.

   The *Virtual Groups* page is displayed.

2. In the *Virtual Groups List*, click the Edit icon ( ) in the **Edit** column for the virtual group entry to edit.

3. Make the required changes to the **Virtual Group Name** field.
4. Click the Save icon () in the **Edit** column.

Note To cancel an edit operation, select **Access > Virtual Groups** again.

Delete a virtual group

Use the following procedure to delete a virtual group.

1. Select **Access > Virtual Groups**.

   The *Virtual Groups* page is displayed.
2. In the *Virtual Groups List*, select the checkbox for each user class to delete.
3. Click **Delete**.
4. Click **OK** in the confirmation dialog box.

Filesystem restrictions

Use the *Filesystem* pane of the *Restrictions* page to control rights for specific user classes to modify files and directories on the SecureTransport server.

The order of the entries in the list in the *Filesystem* pane of the *Restrictions* page is important because SecureTransport applies the filesystem restrictions starting with the last in the list and proceeding to the first. Once the user class for the user filesystem is established, SecureTransport applies the last entry for that user class or for all user classes (*) that has a file pattern that matches the filesystem. If no entry matches, SecureTransport allows access to the filesystem.

For each user class, put the entries with more general paths before those with more specific paths. For example, to allow users to download from the `/outgoing` directory only, put an entry that allows downloads from that path after an entry that denies download from all locations (*).

Operations that can be allowed or denied are:

- **Delete a File** – Specifies whether or not users from the specified user class may delete files on the SecureTransport Server.
- **Rename a File** – Specifies whether or not users from the specified user class may rename files on the SecureTransport Server.
- **Overwrite a File** – Specifies whether or not users from the specified user class may overwrite existing files on the SecureTransport Server.
- **Make a Directory** – Specifies whether or not users from the specified user class may create directories on the SecureTransport Server.
- **Remove a Directory** – Specifies whether or not users from the specified user class may remove directories from the SecureTransport Server.
- **Change File Mode** – Specifies whether or not users from the specified user class may change file access permissions on the SecureTransport Server. This option is applicable only for UNIX.

- **Change Umask** – Specifies whether or not users from the specified user class may change the access permissions mask for new files being uploaded to the SecureTransport Server. This option is applicable only for UNIX.

  Note The SecureTransport Server is compliant with the SFTP protocol specification and does not apply a umask to the mode bits if they are specified by the client. If the mode bits are not specified by the client, the umask is applied based on the configuration of the `Users.DefaultUmask` option.

  Note If the SecureTransport `Users.DefaultUmask` configuration option is set to some value, that value is used for the umask. If the `Users.DefaultUmask` configuration option is empty, the operating system's umask is used.

- **Access a File/Directory** – Specifies whether or not users from the specified user class have access to files or directories on the SecureTransport Server.

Note When an ST Web Client user moves a file using cut and paste, SecureTransport checks the filesystem Rename a File permission. SecureTransport does not check the upload restrictions in this case.

The `Restrictions.OrderOfApplication` server configuration option defines the order of application for filesystem and upload and download restrictions. There are two available values for the option:

- **legacy** (default) - rules are applied from bottom to top
- **new** - rules are applied from top to bottom

Configure filesystem restrictions on SecureTransport Server only.

The following topic describes how to manage filesystem restrictions:

- *Manage filesystem restrictions on page 786* - Provides how-to instructions for managing filesystem restrictions.

Manage filesystem restrictions

Use the *Filesystem* pane of the *Restrictions* page to add, enable, disable, or delete filesystem restriction entries.

Note SecureTransport applies filesystem restrictions starting with the last in the list and proceeding to the first. When you create two or more restrictions with the same action and that apply to the same users and the same path, make sure to put an entry with a more general path above one with a more specific path.

The following topics provide how-to instructions for managing filesystem restrictions:

- *Add a filesystem restriction on page 787*
- *Enable or disable a filesystem restriction on page 788*
- *Edit a filesystem restriction on page 788*

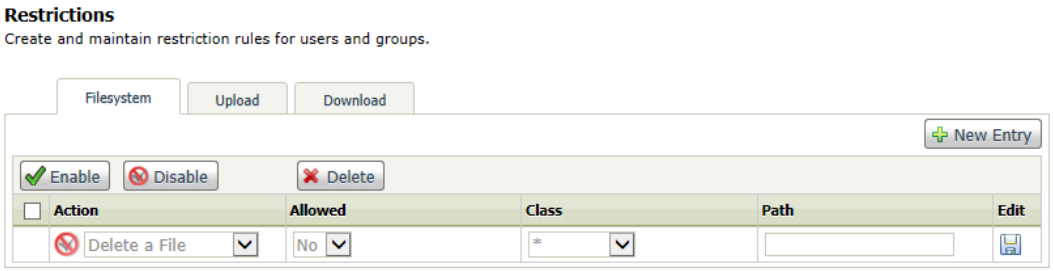
- *Delete a filesystem restriction on page 788*
- *Filesystem restriction examples on page 789*

Add a filesystem restriction

Use the following procedure to add a filesystem restrictions.

1. Select **Access > Restrictions**.
2. Click the **Filesystem** tab.

   The *Filesystem Restrictions* pane is displayed.
3. Click **New Entry**. A new line is displayed in the list.



4. Select an **Action** from the list. For description of the options, see *Filesystem restrictions* on page 785.
5. In the **Allowed** field, select **No** to deny the action or **Yes** to allow it.
6. In the **Class** field, select a user class. Asterisk (`*`) means all users.
7. In the **Path** field, type the path of the directory for which the restriction applies.

   Specify the path relative to the filesystem root for the user. For a real user, the file system root is the operating system root. For a virtual user, the file system root is the user's home directory.

   On Windows, you must use a POSIX path. Specify drives as `/drives/c` and `/drives/d` instead of `C:\` and `D:\`.

   You can use UNIX-style wildcard characters to apply restrictions for an entire directory. Path entries must contain both a forward slash and the asterisk wildcard (`/*`) to allow or deny everything. For example, on Windows, to prevent deletion of the contents of the `C:\temp` directory, specify `/drives/C/temp/*` as the path. In this example, specifying `/drives/C/temp` only prevents the directory itself from being deleted, not its contents. With SecureTransport version 5.4, two new parameters are introduced with Filesystem restrictions: `{s}` and `{i}`. These two options are used as prefixes to regular expressions and their purpose is to match the Path in case sensitive (`{s}`)or case insensitive (`{i}`) manner.

- `{i}` matches Path in an expression in a case-insensitive fashion.
  Example use: Delete a File action of files that match the expression `{i}/*.xml` will delete any xml file, regardless of filename extension case: whether it is XML, xml or XmL.

- `{s}` matches all files in an expression in a case-sensitive fashion.
  Example use: Delete a File action of files that match the expression `{s}/*.TXT` will only delete files with TXT extension (uppercase, as defined in the expression) and will not delete files with .txt extension (lowercase, not defined in the expression).

Note Along with the two regular expression prefixes, a dedicated configuration option is introduced: `Restrictions.pathIgnoreCases`. When it is set to `false`, all expressions that do not use the `{i}` or `{s}` prefixes will match the path in case-sensitive manner. Respectively, when set to `true`, all expressions without the `{i}` or `{s}` prefixes will match in case-insensitive manner.

8. Click the Save icon () in the **Edit** column.

   SecureTransport adds the new entry at the top of the list. The status of a new entry is set to Disabled.

Note To cancel an add operation, select **Access > Restrictions** again.

Enable or disable a filesystem restriction

Use the following procedure to enable or disable a filesystem restriction.

1. Select **Access > Restrictions**.
2. Click the **Filesystem** tab.

   The *Filesystem Restrictions* pane is displayed.
3. Select the checkbox for each entry to modify.
4. Click **Enable** or **Disable**.

   The icons in the **Action** column change to indicate the status of the classes.

Edit a filesystem restriction

Use the following procedure to edit a filesystem restriction.

1. Select **Access > Restrictions**.
2. Click the **Filesystem** tab.

   The *Filesystem Restrictions* pane is displayed.
3. Click the Edit icon () in the **Edit** column for the entry to edit.
4. Make the required changes to the fields.
5. Click the Save icon () in the **Edit** column.

Note To cancel an edit operation, select **Access > Restrictions** again.

Delete a filesystem restriction

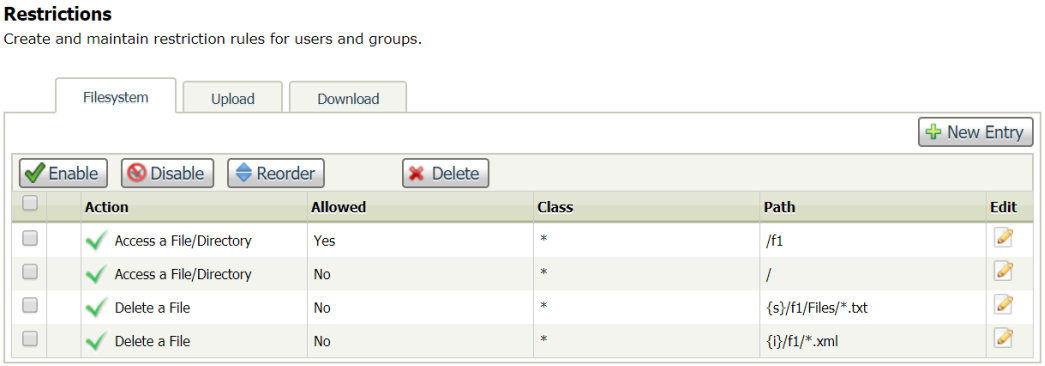
Use the following procedure to delete a filesystem restriction.

1. Select **Access > Restrictions**.
2. Click the **Filesystem** tab.

   The *Filesystem Restrictions* pane is displayed.
3. Select the checkbox for each entry to delete.
4. Click **Delete**.
5. Click **OK** in the confirmation dialog box.

Filesystem restriction examples

Here are some examples of filesystem restrictions:



- The first condition defines access to the `/f1` folder only.
- The second condition restricts access to the root folder.
- The third condition deletes all entirely lowercase TXT files in the `/f1/Files` folder and omits all other case variations (for example,.TxT and .TXT files will not be deleted with this expression). Also, in case there is a `/f1/files` path, its content will remain untouched by this expression since it exactly matches the `/f1/Files` path.
- The fourth condition deletes all XML files in the `/f1` folder (.XML, .xml, .XmL, etc.).

Note Paths are relative to the filesystem root for the user. See *Path specifications*.

Upload restrictions

Use upload restrictions to allow or deny permission for users to upload files based on user class. For each user class, you can specify upload permissions and, for UNIX-based systems, the value of the owner, group and access permissions of the uploaded file.

The order of the entries in the list in the *Upload* pane of the *Restrictions* page is important because SecureTransport applies the upload restrictions starting with the last in the list and proceeding to the first. Once the user class for the user uploading the file is established, SecureTransport applies the last entry for that user class or for all user classes (*) that has a file pattern that matches the file to be uploaded. If no entry matches, SecureTransport allows the upload.

For each user class, put the entries with more general paths before those with more specific paths. For example, to allow users to upload to the `/incoming` directory only, put an entry that allows uploads to that path after an entry that denies upload to all locations (*). See *Path specifications*.

Note For metadata file upload for a protocol implemented using the file services interface, the IP address is not used to choose the user class. The upload restrictions defined for the first user class that matches the user type, name, and group control these transfers.

Configure upload restrictions on SecureTransport Server only.

The following topic describes how-to manage upload restrictions:

- *Manage upload restrictions* on page 790 - Provides how-to instructions for managing upload restrictions.

Manage upload restrictions

Use the *Upload* pane of the *Restrictions* page to add, enable, disable, reorder, or delete upload restriction entries. Using upload restrictions, for each user class, you can allow or deny permission to upload files based on the destination and set the owner, group, or file system permission (mode) of the uploaded file on a UNIX-based system.

The following topics provide how-to instructions for managing upload restrictions:

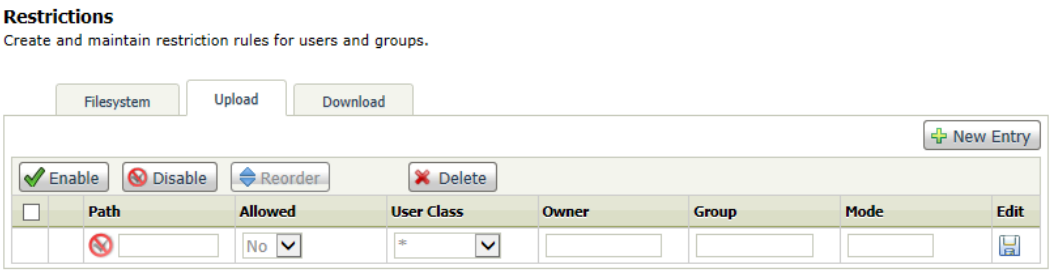
- *Add an upload restriction* on page 790
- *Enable or disable an upload restriction* on page 794
- *Edit an upload restriction* on page 794
- *Reorder upload restrictions* on page 794
- *Reorder upload restrictions* on page 794

Add an upload restriction

Use the following procedure to add an upload restriction.

1. Select **Access > Restrictions**.
2. Click the **Upload** tab.

   The *Upload Restrictions* pane is displayed.
3. Click **New Entry**. A new line is displayed in the list.



4. Complete the fields in the following table.

Field	Description
Path	The file name or directory you want to apply a restriction to. Specify the path relative to the file system root for the user. For a real user, the file system root is the operating system root. For a virtual user, the file system root is the user's home directory. On Windows, you must use a POSIX path. Specify drives as `/drives/c` and `/drives/d` instead of `C:\` and `D:\`. You can use UNIX-style wildcard characters to apply restrictions for an entire directory and its subdirectories. Path entries must contain both a forward slash and the asterisk wildcard (`/*`) to deny or allow everything. Specifying `/temp` applies the restriction only to the directory itself, not its contents. To apply the restriction to the directory and its contents, you must specify `/temp/*`. For example, if you specify `/drives/C/temp` as the path and allow uploads, uploading will be allowed but Owner, Group, and Mode will not be applied to the uploaded file. To apply Owner, Group, and Mode or to allow uploads to subdirectories of `/temp`, you must specify `/drives/C/temp/*` as the path. With SecureTransport version 5.4, two new parameters are introduced with Filesystem restrictions: `{s}` and `{i}`. These two options are used as prefixes to regular expressions and their purpose is to match the Path in case sensitive (`{s}`)or case insensitive (`{i}`) manner. • `{i}` matches Path in an expression in a case-insensitive fashion. Example use: Access to path that matches the expression `{i}/*.xml` will allow the user to upload any xml file, regardless of filename extension case: whether it is XML, xml or XmL. • `{s}` matches all files in an expression in a case-sensitive fashion. Example use: Access to path that matches the expression `{s}/*.TXT` will allow the user to upload only files with TXT extension (uppercase, as defined in the expression) and will not be able to upload files with .txt extension (lowercase, not defined in the expression). Along with the two regular expression prefixes, a dedicated configuration option is introduced: `Restrictions.pathIgnoreCases`. When it is set to `false`, all expressions that do not use the `{i}` or `{s}` prefixes will match the path in case-sensitive manner. Respectively, when set to `true`, all expressions without the `{i}` or `{s}` prefixes will match in case-insensitive manner.
Allowed	Select **Yes** or **No** based on whether you want to restrict uploading.

Field	Description
User Class	Select a user class. Asterisk (`*`) means all users.
Owner	(UNIX-based systems only) User ID to be set for the uploaded file when **Allowed** is **Yes**. When the `Users.Uploads.RestrictionsApplication` server configuration parameter is set to `limited`, the value of this field is ignored if the file mode is set by the client. The `Users.Uploads.RestrictionsApplication` server configuration parameter default value is: `limited` When the `Users.Uploads.RestrictionsApplication` server configuration parameter is set to `full`, the value of this field is always applied.
Group	(UNIX-based systems only) Group name or ID to be set for the uploaded file when **Allowed** is **Yes**. When the `Users.Uploads.RestrictionsApplication` server configuration parameter is set to `limited`, the value of this field is ignored if the file mode is set by the client. The `Users.Uploads.RestrictionsApplication` server configuration parameter default value is: `limited` When the `Users.Uploads.RestrictionsApplication` server configuration parameter is set to `full`, the value of this field is always applied.
Mode	(UNIX-based systems only) File access permissions to be applied to the uploaded file when **Allowed** is **Yes**. When the `Users.Uploads.RestrictionsApplication` server configuration parameter is set to `limited`, the value of this field is ignored if the file mode is set by the client. The `Users.Uploads.RestrictionsApplication` server configuration parameter default value is: `limited` When the `Users.Uploads.RestrictionsApplication` server configuration parameter is set to `full`, the value of this field is always applied. If you leave this field empty, the file mode set by the client is applied regardless of the value of `Users.Uploads.RestrictionsApplication` server configuration parameter.

5. Click the Save icon () in the **Edit** column.

   The status of a new entry is set to Disabled.

Note To cancel an add operation, select **Access > Restrictions** again.

Enable or disable an upload restriction

Use the following procedure enable or disable an upload restriction.

1. Select **Access > Restrictions**.
2. Click the **Upload** tab.

   The *Upload Restrictions* pane is displayed.
3. Select the checkbox for each entry to modify.
4. Click **Enable** or **Disable**.

   The icons in the **Path** column change to indicate the status of the classes.

Edit an upload restriction

Use the following procedure to edit an upload restriction.

1. Select **Access > Restrictions**.
2. Click the **Upload** tab.

   The *Upload Restrictions* pane is displayed.
3. Click the Edit icon ( ) in the **Edit** column for the entry to edit.
4. Make the required changes in the fields.
5. Click the Save icon ( ) in the **Edit** column.

Note To cancel an edit operation, select **Access > Restrictions** again.

Reorder upload restrictions

Use the following procedure to reorder upload restrictions.

1. Select **Access > Restrictions**.
2. Click the **Upload** tab.

   The *Upload Restrictions* pane is displayed.
3. Click **Reorder**.

   Up and down arrows are displayed in a column before the **Path** column.
4. Drag the rows of the entries to the required order.
5. Click **Save Order**.

Note To cancel a reorder operation, select **Access > Restrictions** again.

Delete an upload restriction

Use the following procedure to delete an upload restriction.

1. Select **Access > Restrictions**.
2. Click the **Upload** tab.

   The *Upload Restrictions* pane is displayed.
3. Select the checkbox for each entry to delete.
4. Click **Delete**.
5. Click **OK** in the confirmation dialog box.

Download restrictions

Use download restrictions to allow or deny permission for users to download files based on user class. For each user class, you can specify upload permissions that allow or deny the user the ability to view, create, or modify files and directories.

The order of the entries in the list in the *Download* pane of the *Restrictions* page is important because SecureTransport applies the download restrictions starting with the last in the list and proceeding to the first. Once the user class for the user downloading the file is established, SecureTransport applies the last entry for that user class or for all user classes (*) that has a file pattern that matches the file to be downloaded. If no entry matches, SecureTransport allows the download.

For each user class, put the entries with more general paths before those with more specific paths. For example, to allow users to download from the `/outgoing` directory only, put an entry that allows downloads from that path after an entry that denies download from all locations (*).

Configure download restrictions on SecureTransport Server only.

The following topic describes how to manage download restrictions:

- *Manage download restrictions* on page 795 - Provides how-to instructions for managing download restrictions.

Manage download restrictions

Use the *Download* pane of the *Restrictions* page to add, enable, disable, reorder, or delete download restriction entries.

The following topics provide how-to instructions for managing download restrictions:

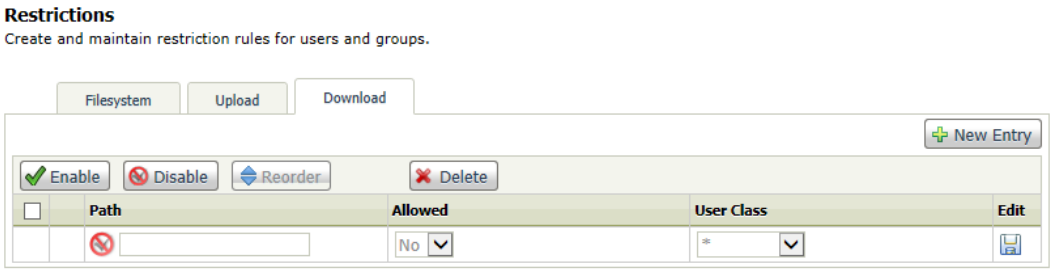
- *Add a download restriction* on page 796
- *Enable or disable a download restriction* on page 798
- *Edit a download restriction* on page 798
- *Reorder download restrictions* on page 798
- *Delete a download restriction* on page 799

Add a download restriction

Use the following procedure to add a download restriction.

1. Select **Access > Restrictions**.
2. Click the **Download** tab.

   The *Download Restrictions* pane is displayed.
3. Click **New Entry**. A new line is displayed in the list.



4. Complete the fields in the following table.

Field	Description
Path	The file name or directory you want to apply a restriction to. Specify the path relative to the file system root for the user. For a real user, the file system root is the operating system root. For a virtual user, the file system root is the user's home directory. On Windows, you must use a POSIX path. Specify drives as `/drives/c` and `/drives/d` instead of `C:\` and `D:\`. You can use UNIX-style wildcard characters to apply restrictions for an entire directory and its subdirectories. Path entries must contain both a forward slash and the asterisk wildcard (`/*`) to deny or allow everything. For example, on Windows, to prevent deletion of the contents of the `C:\temp` directory, specify `/drives/C/temp/*` as the path. In this example, specifying `/drives/C/temp` applies the restriction only to the directory itself, not its contents. With SecureTransport version 5.4, two new parameters are introduced with Filesystem restrictions: `{s}` and `{i}`. These two options are used as prefixes to regular expressions and their purpose is to match the Path in case sensitive (`{s}`) or case insensitive (`{i}`) manner. • `{i}` matches Path in an expression in a case-insensitive fashion. Example use: Access to path that matches the expression `{i}/*.xml` will allow the user to download any xml file, regardless of filename extension case: whether it is XML, xml or XmL. • `{s}` matches all files in an expression in a case-sensitive fashion. Example use: Access to path that matches the expression `{s}/*.TXT` will allow the user to download only files with TXT extension (uppercase, as defined in the expression) and will not be able to download files with .txt extension (lowercase, not defined in the expression). Along with the two regular expression prefixes, a dedicated configuration option is introduced: `Restrictions.pathIgnoreCases`. When it is set to `false`, all expressions that do not use the `{i}` or `{s}` prefixes will match the path in case-sensitive manner. Respectively, when set to `true`, all expressions without the `{i}` or `{s}` prefixes will match in case-insensitive manner.
Allowed	Select **Yes** or **No** based on whether you want to restrict downloading.
User Class	Select a user class. Asterisk (`*`) means all users.

5. Click the Save icon () in the **Edit** column.

   The status of a new entry is set to Disabled.

Note To cancel an add operation, select **Access > Restrictions** again.

Enable or disable a download restriction

Use the following procedure to enable or disable a download restriction.

1. Select **Access > Restrictions**.
2. Click the **Download** tab.

   The *Download Restrictions* pane is displayed.
3. Select the checkbox for each entry to modify.
4. Click **Enable** or **Disable**.

   The icons in the **Path** column change to indicate the status of the classes.

Edit a download restriction

Use the following procedure to edit a download restriction.

1. Select **Access > Restrictions**.
2. Click the **Download** tab.

   The *Download Restrictions* pane is displayed.
3. Click the Edit icon () in the **Edit** column for the entry to edit.
4. Make the required changes in the fields.
5. Click the Save icon () in the **Edit** column.

Note To cancel an edit operation, select **Access > Restrictions** again.

Reorder download restrictions

Use the following procedure to reorder download restrictions.

1. Select **Access > Restrictions**.
2. Click the **Download** tab.

   The *Download Restrictions* pane is displayed.
3. Click **Reorder**.

   Up and down arrows are displayed in a column before the **Path** column.
4. Drag the rows of the entries to the required order.
5. Click **Save Order**.

Note To cancel a reorder operation, select **Access > Restrictions** again.

Delete a download restriction

Use the following procedure to delete a download restriction.

1. Select **Access > Restrictions**.
2. Click the **Download** tab.

   The *Download Restrictions* pane is displayed.
3. Select the checkbox for each entry to delete.
4. Click **Delete**.
5. Click **OK** in the confirmation dialog box.

FTP command restrictions

The SecureTransport FTP server uses a number of standard and extended FTP commands. You can restrict individual FTP commands for specified user classes. By default, the page includes entries that allow all listed commands for all users.

To allow an FTP command for some users and restrict it for others, create two or more entries and reorder them so that the more restrictive rule comes before the less restrictive rules.

If your SecureTransport deployment includes SecureTransport Edge servers in a peripheral network (DMZ) and if you need different restrictions for users who connect using SecureTransport Edge and using SecureTransport Server, you can configure FTP command restrictions differently.

The following topics describe the FTP SITE command and the how-to instructions for managing FTP command restrictions:

- *FTP SITE command* on page 799 - Describes the FTP SITE command.
- *Manage FTP command restrictions* on page 799 - Provides how-to instructions for managing FTP command restrictions.

FTP SITE command

Some FTP `SITE` commands that SecureTransport accepts are handled differently than the other FTP commands. The `SITE VERS`, `SITE AUTH` and `SITE FEAT` commands are not listed even though SecureTransport accepts those commands. Those commands must always be allowed, because Axway Secure Client depends on the responses of those commands to determine the capabilities of SecureTransport Server. SecureTransport restricts the `SITE HELP` command based on the setting for the `HELP` command.

Manage FTP command restrictions

Use the *FTP Commands* page to allow or restrict individual FTP commands for specified user classes.

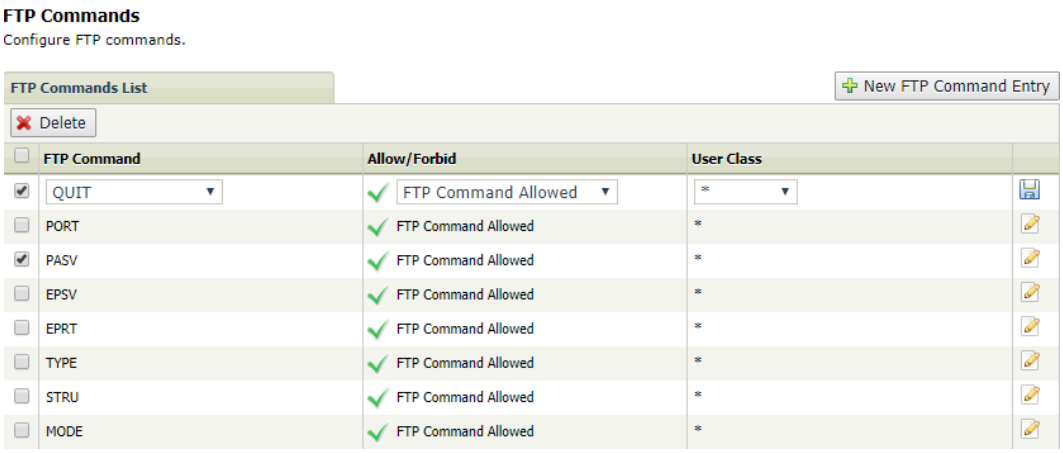
The following topics provide how-to instructions for managing FTP command restrictions:

- *Add an FTP command entry* on page 800
- *Edit an FTP command entry* on page 800
- *Delete an FTP command entry* on page 801

Add an FTP command entry

Use the following procedure to add an FTP command entry.

1. Select **Access > FTP Commands**.
2. The *FTP Commands* page is displayed.
3. Click **New FTP Command Entry**. A new line is displayed in the list.



4. Select an **FTP Command** to allow and restrict.
5. In the **Allow/Forbid** field, select **FTP Command Allowed** or **FTP Command Forbidden**.
6. Select the **User Class**. Asterisk (*) means all users.
7. Click the Save icon ( ) in the **Edit** column.

   The new entry is added before the existing entries for that FTP command.

Note To cancel an add operation, select **Access > FTP Commands** again.

Edit an FTP command entry

Use the following procedure to edit an FTP command entry.

1. Select **Access > FTP Commands**.

   The *FTP Commands* page is displayed.
2. Click the Edit icon ( ) in the **Edit** column for the entry to edit.
3. Make the required changes in the fields.
4. Click the Save icon ( ) in the **Edit** column.

Note To cancel an edit operation, select **Access > FTP Commands** again.

Delete an FTP command entry

Use the following procedure to delete an FTP command entry.

1. Select **Access > FTP Commands**.

   The *FTP Commands* page is displayed.
2. Select the checkbox for each entry to delete.
3. Click **Delete**.
4. Click **OK** in the confirmation dialog box.

Note You cannot delete the last entry for each FTP command.

Control access to Administration Tool and protocol servers

You can control access to the SecureTransport Administration Tool and the protocol FTP, HTTP, and SSH servers by creating access rules. You can allow or deny access, or use a combination of allow and deny, as well as define any number of rules and set their order.

Control access to Administration Tool

The access rules for the Administration Tool are configured on the *Admin Access Control* page. An access rule permits or denies access based on an IP address or a range of IP addresses. When a client attempts to connect to the SecureTransport Administration Tool, the server first checks if the client matches any deny rule. DNS lookup is supported, but you must first enable it by setting the `Admin.ReverseDNSLookup` configuration option to `true`.

In a streaming deployment, you can configure access control differently on the backend and the EDGE servers.

Control access to protocol servers

The access rules for the protocol servers are configured on the *Server Access Control* page, and only on SecureTransport Server. You can limit access to the SecureTransport FTP, HTTP and SSH servers for specific authenticated users based on user class and client host address. DNS lookup is supported, but you must first enable it. See *Enable host names for access control* on page 803 Protocol server rules are applied after successful authentication. When a client attempts to connect to a protocol server and authenticates successfully, the server first checks the configured user class rules, and then the IP address.

Note Protocol server access restrictions do not work for SiteMinder logins.

The following sections describe access rule ordering and provide how-to instructions on how to create, delete, and order access rules:

- *Manage server access* on page 803
- *Access rule order* on page 802

Access rule order

You set the order that the server applies the access rules. Using rule order and multiple rules, you can implement detail access control.

If you select *Allow then Deny* rule order, the server denies access to a computer if:

- It is not explicitly specified in an allow rule

  or

- It is explicitly specified in a deny rule.

With *Allow then Deny*, if the IP address or host name of a computer is not specified in either an allow rule or a deny rule, the server denies access. So, the default is no access.

A more general deny rule overrides a more specific allow rule, so to allow access from an entire subnet or range of IP addresses and deny access from specific hosts, select *Allow then Deny* and define an allow rule for the subnet or range of IP addresses and deny rules for each host. For protocol servers (FTP, HTTP, SSH), the evaluation of the IP address is based on the user class. If the client's user group matches a restricted one, the server proceeds to the IP address check. If it doesn't, the client can connect.

Note Be careful not to deny access to the Administration Tool from all computers. If you select *Allow then Deny* rule order and delete all admin access control rules, no computer can access the Administration Tool. If this happens, contact Axway Global Support.

If you select *Deny then Allow* rule order, the Administration Tool server allows access to a computer if:

- It is not explicitly specified in a deny rule

  or

- It is explicitly specified in an allow rule.

With *Deny the Allow*, if the IP address or host name of a computer is not specified in either an allow rule or a deny rule, the server allows access. So, the default is access.

A more general allow rule overrides a more specific deny rule. So, to deny access from an entire subnet or range of IP addresses and allow access from specific hosts, select *Deny the Allow* and define a deny rule for the subnet or range of IP addresses and allow rules for each host. For protocol servers (FTP, HTTP, SSH), the address is evaluated based on the user class. If a restricted user class matches user's, proceeding with evaluation of the address. Otherwise false is returned for both allow/deny access rules.

Enable host names for access control

To use host names in the **Address** field of rules that control access to the Administration Tool, FTP, HTTP, and SSH servers, enable reverse DNS lookup for those severs.

Note Enabling reverse DNS lookup might reduce the server's performance because DNS lookups involve a series of requests through the DNS name server tree.

Enable reverse DNS lookups for the Administration Tool server

Use the following procedure to enable reverse DNS lookups for the Administration Tool server.

1. Select **Operations > Server Configuration**.

   The *Server Configuration* page is displayed.

2. To enable reverse DNS lookups for the Administration Tool server, search for the `Admin.ReverseDNSLookup` parameter and set it to `true`.

3. Bounce the server.

Enable reverse DNS lookups for the FTP, HTTP, or SSH server

Use the following procedure to enable reverse DNS lookups for the FTP, HTTP, or SSH servers.

1. Select **Setup > Miscellaneous**.

   The *Miscellaneous Options* page is displayed.

2. In the **Reverse DNS Lookups** list, select `Reverse DNS lookups enabled`.

3. Click **Apply**.

Manage server access

You can define any number of rules to control access to the SecureTransport Administration Tool or the FTP, HTTP, and SSH servers.

The following topics provide how-to instructions for managing server access:

- *Add an access rule* on page 804
- *Enable or disable an access rule* on page 804
- *Edit an access rule* on page 805
- *Delete an access rule* on page 805
- *Change the order that rules are applied* on page 805
- *Server access rules example* on page 805

Add an access rule

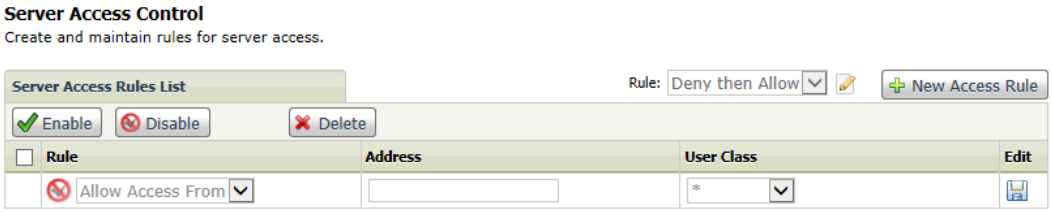
Use the following procedure to add an access rule.

1. Select **Access > Admin Access Control** or **Access > Server Access Control**.

   The *Admin Access Control* or *Server Access Control* page is displayed.

   The initial configuration allows access to the Administration Tool from all computers.

2. Click **New Access Rule**. A new line is displayed in the list.



3. In the **Rule** field, select an access permission. The types of access permissions available are:
   - Allow Access From
   - Deny Access From
4. In the Address field, enter a host name, an IP address, or a value that represents a range of IP addresses to apply the rule. For valid IP addresses and values for IP address ranges, see *IP addresses and host names* on page 1101.

   Only one host name, IP address, or IP address range value is allowed. A host name pattern is not valid.
5. For a server access rule, select a **User Class**. Asterisk (*) means all users.
6. Click the Save icon ( ) in the **Edit** column.

   The status of a new entry is set to Disabled.

Note To cancel an add operation, select **Access > Admin Access Control** or **Access > Server Access Control** again.

Enable or disable an access rule

Use the following procedure to enable or disable an access rule.

1. Select **Access > Admin Access Control** or **Access > Server Access Control**.

   The *Admin Access Control* or *Server Access Control* page is displayed.
2. Select the checkbox for each rule to modify.
3. Click **Enable** or **Disable**.

   The icons in the **Rule** column change to indicate the status of the classes.

Edit an access rule

Use the following procedure to edit an access rule.

1. Select **Access > Admin Access Control** or **Access > Server Access Control**.

   The *Admin Access Control* or *Server Access Control* page is displayed.
2. Click the Edit icon ( ) in the **Edit** column for the rule to edit.
3. Make the required changes in the fields.
4. Click the Save icon ( ) in the **Edit** column.

Note To cancel an edit operation, select **Access > Admin Access Control** or **Access > Server Access Control** again.

Delete an access rule

Use the following procedure to delete an access rule.

1. Select **Access > Admin Access Control** or **Access > Server Access Control**.

   The *Admin Access Control* or *Server Access Control* page is displayed.
2. Select the checkbox for each rule to delete.
3. Click **Delete**.
4. Click **OK** in the confirmation dialog box.

Change the order that rules are applied

Use the following procedure to change the order that access rules are applied.

1. Select **Access > Admin Access Control** or **Access > Server Access Control**.

   The *Admin Access Control* or *Server Access Control* page is displayed.
2. Click the Edit icon ( ) to the right of the **Rule** list.
3. Select a rule order. The available rule orders are:
   - Deny then Allow
   - Allow then Deny
4. Click the Save icon ( ) to the right of the **Rule** list.

Server access rules example

The following example uses an IP address pattern to specify a range of IP addresses as described in *IP addresses and host names* on page 1101. It denies access to all computers except those in the 198.160.123 subnet.

User limits

Use user limit entries to limit the number of concurrent users who can connect to the SecureTransport FTP and HTTP servers. The limit you define applies to each protocol server, if you limit the users from a user class to 10, 10 users can connect concurrently to the FTP server and 10 can connect to the HTTP server. You can also specify a time range and days of the week to apply the limit.

If your SecureTransport deployment includes SecureTransport Edge servers in a peripheral network (DMZ) and if you need different user limits for users who connect using SecureTransport Edge and using SecureTransport Server, you can configure user limits differently. For example, you can allow access to users who connect to SecureTransport Server from your internal network during a time you restrict access to users who connect using SecureTransport Edge.

SecureTransport applies all user limits defined for all users or for the user class that the user is in. If no user limit applies to the user, the user can connect to the FTP and HTTP servers. However, connections might be limited by licenses or hardware or network capacity.

User limits do not apply to protocol servers other than FTP and HTTP.

The following topic describes how to manage user limits:

- *Manage user limits* on page 806 - Provide how-to instructions for managing user limits.

Manage user limits

Use the *User Limits* pane of the *Access Rules* page to add, enable, disable, reorder, or delete user limits. Using user limits, you can set the maximum number of users who can log in to SecureTransport for each user class. You can also specify the days and time the limit is in effect.

The following topics provide how-to instructions for managing user limits:

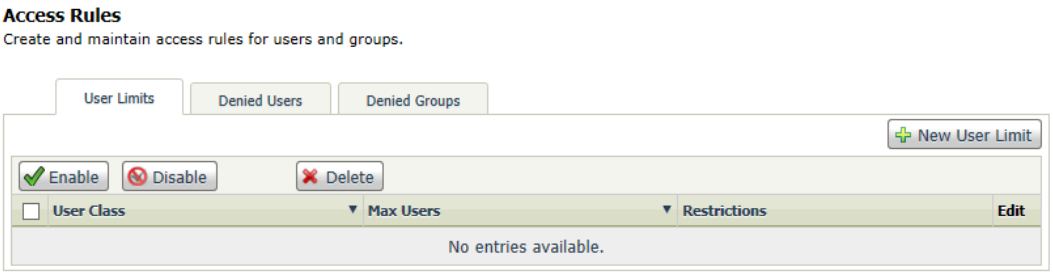
- *Add a user limit* on page 806
- *Enable or disable a user limit* on page 808
- *Edit a user limit* on page 808
- *Delete a user limit* on page 808

Add a user limit

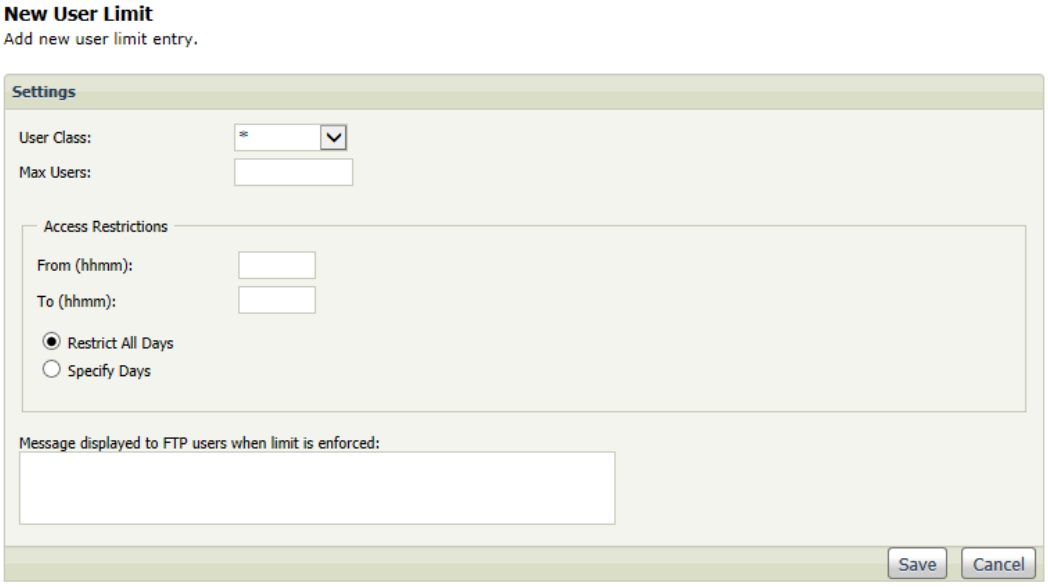
Use the following procedure to add a user limit.

1. Select **Access > Access Rules**.
2. Click the **User Limits** tab.

   The *User Limits* pane is displayed.



3. Click **New User Limit**. The *New User Limit* page is displayed.



4. Select a **User Class**. Asterisk (*) means all users.
5. Type the maximum number of concurrent users for that class.

   Note SecureTransport applies the user limit separately for each protocol. For example, if the maximum users for a user class is 50, SecureTransport allows a maximum of 50 concurrent connections to the FTP server and 50 connections to the HTTP server.

6. Under *Access Restrictions*, to specify the start and end times for SecureTransport to apply the restriction, enter the time in 24-hour format in the **From** and **To** fields. To specify that SecureTransport apply the restriction all the time, leave the **From** and **To** fields empty.
7. Under *Access Restrictions*, to specify the days of the week for SecureTransport to apply the restriction, click **Specify Days** and select the days. To specify that SecureTransport apply the restriction on all days, click **Restrict All Days**.
8. In the field provided, enter a message to be displayed when a user tries to connect to the FTP server and is denied access due to this user limits restriction.
9. Click **Save.**

The *User Limits* pane of the *Access Rules* page is displayed with the new user limit listed. The status of a new user limit is set to Disabled.

Enable or disable a user limit

Use the following procedure to enable or disable a user limit.

1. Select **Access > Access Rules**.
2. Click the **User Limits** tab.

   The *User Limits* pane is displayed.
3. Select the checkbox for each entry to modify.
4. Click **Enable** or **Disable**.

   The icons in the **User Class** column change to indicate the status of the classes.

   The *User Access* page is displayed.

Edit a user limit

Use the following procedure to edit a user limit.

1. Select **Access > Access Rules**.
2. Click the **User Limits** tab.

   The *User Limits* pane is displayed.
3. Click the Edit icon ( ) in the **Edit** column for the entry to edit. The *User Limit* page is displayed.
4. Make the required changes in the fields.
5. Click **Save.**

   The *User Limits* pane of the *Access Rules* page is displayed with the user limit updated.

Delete a user limit

Use the following procedure to delete a user limit.

1. Select **Access > Access Rules**.
2. Click the **User Limits** tab.

   The *User Limits* pane is displayed.
3. Select the checkbox for each user limit to delete.
4. Click **Delete**.
5. Click **OK** in the confirmation dialog box.

User and group access

Use this configuration to deny access to SecureTransport to individual users and to users who are members of a group.

By default, the rules deny access to the user `root` and the group `daemon`.

User and group access restrictions apply only to the FTP and HTTP servers.

If your SecureTransport deployment includes SecureTransport Edge servers in a peripheral network (DMZ) and if you need different user and group access for users who connect using SecureTransport Edge and using SecureTransport Server, you can configure user and group access rules differently.

The following topic describes how-to manage user and group access:

- *Manage user and group access* on page 809 - Provides how-to instructions for managing user and group access.

Manage user and group access

Use the *Denied Users* pane and the *Denied Groups* pane of the *Access Rules* page to add and remove user and groups that SecureTransport prevents from connecting to the FTP and HTTP servers.

Every group listed as a denied group must be defined at the operating system level so that SecureTransport can determine the group name from the GID in the user account.

The following topics provide how-to instructions for managing user and group access:

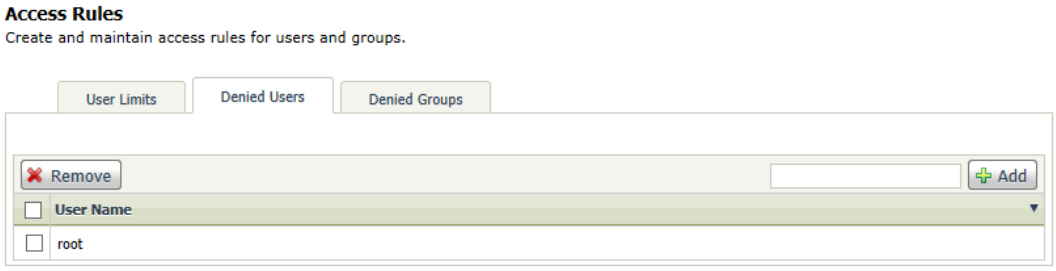
- *Add a user or group to the denied list* on page 809
- *Remove users or groups from the denied list* on page 810

Add a user or group to the denied list

Use the following procedure to add a user or group to the denied list.

1. Select **Access > Access Rules**.
2. Click the **Denied Users** tab or the **Denied Groups** tab.

   The *Denied Users* or *Denied Groups* pane is displayed.



3. Enter in the field to the left of the **Add** button a user name for denied users or a group name or group ID (GID) for denied groups.
4. Click **Add**.
5. The user or group is added to the denied list.

Remove users or groups from the denied list

Use the following procedure to remove users or groups from the denied list.

1. Select **Access > Access Rules**.
2. Click the **Denied Users** tab or the **Denied Groups** tab.

   The *Denied Users* or *Denied Groups* pane is displayed.
3. Select the checkbox for each user or group to remove.
4. Click **Remove**.
5. Click **OK** in the confirmation dialog box.

Login restrictions

Login restrictions define and restrict the rights of individuals to log in to SecureTransport Servers or SecureTransport Edges through the configuration and use of login restriction policies. The configured login restriction policies are applicable to user accounts, account templates, and business units in a hierarchical inheritance and precedence order.

The login restriction policy inheritance order is:

1. If a policy is not defined on the account or account template level, then the policy of the associated business unit is used.
2. If a policy is not defined either on the account or account template level or on the business unit level, but a default policy is set, then the default policy applies.
3. If no policy is set on the account or template, business unit, or default levels, then access is not restricted.

The login restriction policy precedence order is:

1. If a policy is defined on the account or account template level, then it takes precedence over the policy assigned on the business unit level.
2. If no policy is defined on the account or account template level, then the policy assigned on the business unit level takes precedence.

A single user restriction policy can be selected and set as the default policy. The default policy is the suggested user restriction policy when creating a new business unit, account, or account template but it also applies to users that do not have a corresponding user account and do not match an account template. This enables login restrictions to be defined even for external users who do not have any account definition inside SecureTransport.

Note Login restriction policies do not apply to connections over the PeSIT and AS2 protocols.

Refer to the following topics to manage user accounts, account templates, and business units:

- To manage user accounts, refer to *User accounts* on page 501.
- To manage account templates, refer to *Manage account templates* on page 719.
- To mange business units, refer to *Business units* on page 746.

Refer to the following topics to manage login restrictions and create login restriction policies:

- *Manage Login Restriction Policies* on page 811
- *Manage Login Restriction Policy rules* on page 813

Manage Login Restriction Policies

Login restrictions limit access to SecureTransport Servers and SecureTransport Edges through the evaluation of `Allow then Deny` and `Deny then Allow` Login Restriction Policies for end users. The following table describes the evaluation result for the two types of policies.

Match	Allow then Deny	Deny then Allow
Match Allow only	Access is allowed	Access is allowed
Match Deny only	Access is denied	Access is denied
No match	Access is denied	Access is allowed
Match both Allow and Deny	Access is denied	Access is allowed

The following topics provide how-to instructions for managing login restriction policies:

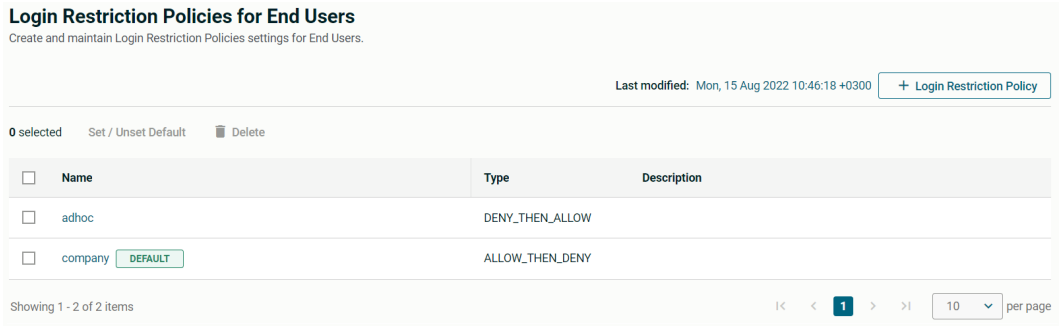
- *Add or edit Login Restriction Policies* on page 811
- *Change the default Login Restriction Policy* on page 813
- *Delete Login Restriction Policies* on page 813

Add or edit Login Restriction Policies

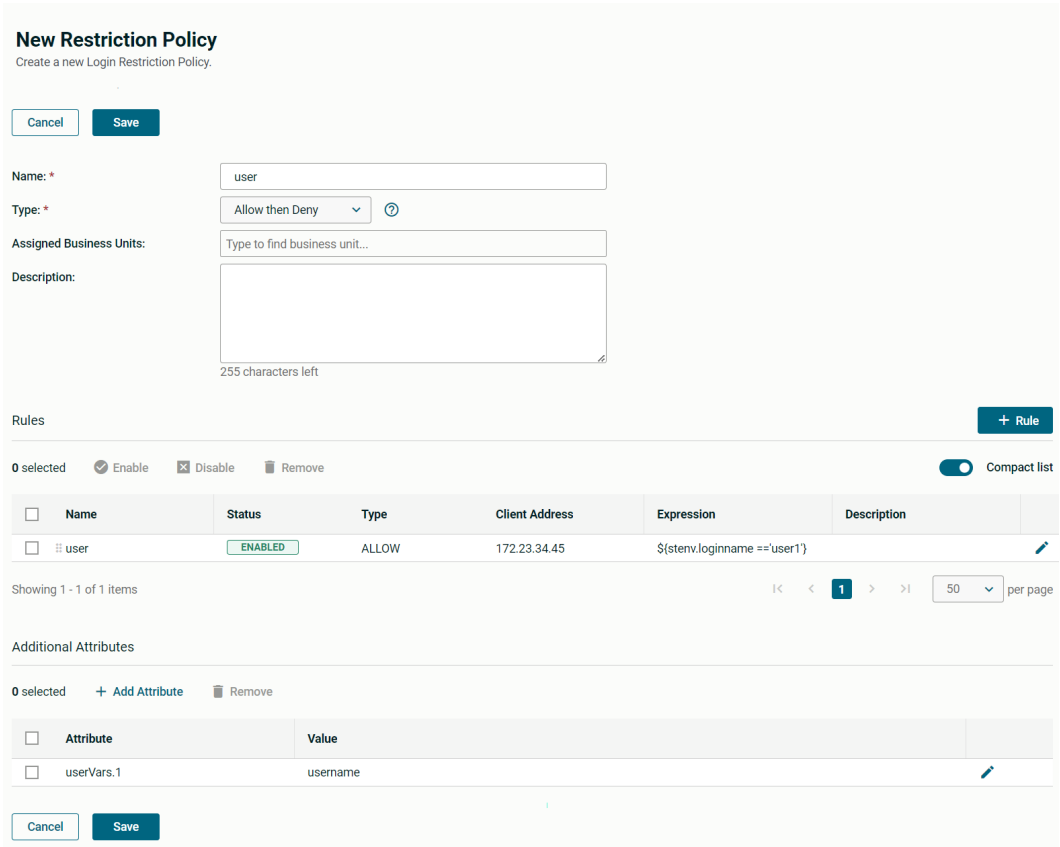
Note Login restriction policies do not apply to connections over the PeSIT and AS2 protocols.

Use the following procedure to add or edit a login restriction policy:

1. Navigate to **Access > Login Restrictions**.



2. Click **+ Login Restriction Policy** to add a new policy, or click on the name of an already existing policy to edit it.



3. Enter a **Name** for the policy.
4. Select the policy **Type**:
   - `Allow then Deny` - login access is denied unless there is a matching Allow rule. If both a Deny and an Allow rule are matched, access is denied.
   - `Deny then Allow` - login access is allowed unless there is a matching Deny rule. If both a Deny and an Allow rule are matched, access is allowed.
5. (Optional) **Assign Business Units** to the login restriction policy.
6. (Optional) Enter a **Description**.

7. Add policy rules. Note that new rules are enabled by default. For details, see *Add or edit a policy rule* on page 813.
8. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

   Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
9. Click **Save**.

Change the default Login Restriction Policy

To change the default login restriction policy:

1. Select the checkbox for the login restriction policy you want to set as default.
2. Click the **Set / Unset Default** button.

   The selected policy is set as default. The default policy is indicated by the addition of a green DEFAULT label.

To unselect the default login restriction policy:

1. Select the checkbox for the current default policy.
2. Click the **Set / Unset Default** button.

   The selected policy is unset as default. No default login restriction policy is defined.

Delete Login Restriction Policies

Use the following procedure to delete a login restriction policy:

1. Select the checkbox for the policy you want to delete.
2. Click the **Delete** button.
3. Confirm the deletion.

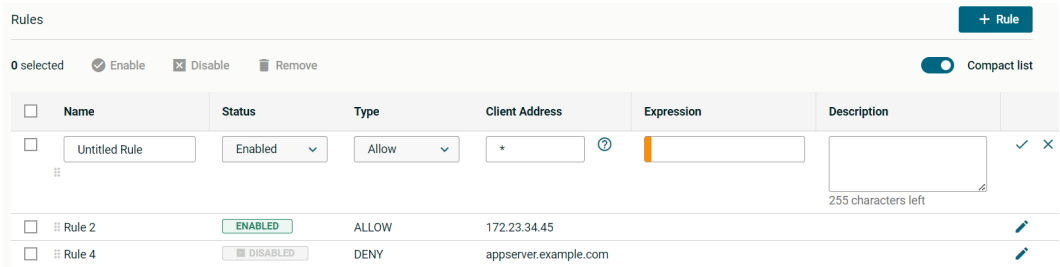
Manage Login Restriction Policy rules

The following topics provide instructions for managing policy rules.

Add or edit a policy rule

Use the following procedure to add or edit a Login Restriction Policy rule.

1. Click on the name of an existing policy, or **+ Login Restriction Policy** to create a new one. See *Add or edit Login Restriction Policies* on page 811.
2. Click **+ Rule** to add a new rule, or click  next to an existing rule to edit it.



3. Enter a **Name** for the policy rule.
4. Select the **Status**.
5. Select the **Type**:
   - `Allow` - the policy rule allows access for the client address.
   - `Deny` - the policy rule denies access to the client address.
6. Enter a **Client Address**:
   - **IPv4 address**: Enter an exact IPv4 to specify a single host.

     Examples: `172.23.34.45`; `127.0.0.1`
   - **IPv6 address**: Enter an exact IPv6 to specify a single host (two colons (`::`) can represent one sequence of zero bits).

     Examples: `FC00:1234:56:0:0:0:AB:EF`; `FC00:1234:56::AB:EF`; `::1`
   - **IPv4 CIDR**: Classless Inter-Domain Routing (CIDR) notation specifies an IPv4 address and a number of significant bits separated by a slash (`/`). Use CIDR notation to represent a range of IP addresses.

     Example: `172.23.34.0/24` represents `172.23.34.0` through `172.23.34.255`
   - **IPv6 CIDR**: Classless Inter-Domain Routing (CIDR) notation specifies an IPv6 address and a number of significant bits separated by a slash (`/`). Use CIDR notation to represent a range of IP addresses.

     Example: `FC00:1234:56::/120` represents `FC00:1234:56::` through `FC00:1234:56::FF`
   - **Host**: Enter a valid host name that resolves to a valid IPv4 or IPv6 address.

     Example: `appserver`
   - **FQDN**: Enter an FQDN that has a successful DNS resolution.

     Example: `appserver.example.com`
   - **Wildcard pattern**: Enter a host name pattern using an asterisk (`*`) to represent one or

more characters.

Examples: `*.example.com`; `example.*`

- **All addresses**: Use an asterisk (`*`) to allow all client addresses.

7. (Optional) Enter an **Expression**.

   Specify an expression using SecureTransport expression language. Use the following named variable sets:

   - `${sess['variable']}`
   - `${env['variable']}, ${stenv['variable']}, or ${stenv.variable}`

   If an expression is specified, a rule will be applied only if both the client address matches and the expression evaluates to true.

   If no expression is specified, only the client address is considered.

   Example:

   ```
   ${stenv.loginname =='user1'}
   ```

   You can create a rule that limits the possible concurrent open sessions by a user. To do this, you must use the `currentSessions` variable and evaluate it against the threshold value set in your rule.
   Example:

   `${currentSessions <= 3}` - this example sets a session limit of up to 3 concurrent sessions per user

   To restrict user logins to a specific Edge server or a network zone, you can use two variables:

   - The `DXAGENT_CLIENTADDR` variable effectively represents an Edge server hostname (or an IP address, depending on the network setup) and is always present when connecting through an Edge server. It can be used in Login Restriction Policy expressions, such as `{stenv.clientaddr}` or `${sess.clientaddr}`.
   - When connecting through an Edge server, the value of the `DXAGENT_EDGEID` variable is taken from the configuration option `EdgeId`. This configuration option is defined by a SecureTransport administrator and is valid for FTP and HTTP daemons only. It can be used in Login Restriction Policy expressions, such as `{stenv.edgeid}` or `${sess.edgeid}`.

   When a user logs in through the Private zone (i.e., through Backend protocol daemons), these variables are not available in the environment, and in this case the only valid expressions are `${empty sess.clientaddr}` or `${empty sess.edgeid}`.

   You may use custom HTTP headers in Login Restriction Policy expressions, where the name of the HTTP header must be capitalized and all dashes must be replaced by underscores (_).
   Example:

   ```
   ${env['DXAGENT_HTTP_X_FORWARDED_FOR'] == '10.10.10.10'}
   ```

8. (Optional) Enter a **Description** for the policy rule.
9. Click ✓ to save the rule, or ✕ to cancel.
10. Click **Save** to save the Login Restriction Policy and its rule(s).

To reorder rules, click ⋮⋮ (*reorder* icon) in front of a rule and use the drag and drop method. Alternatively, right-click on a rule to open the reorder menu.

Enable, disable or remove a policy rule

Select one or more policy rules using the checkboxes, and use one of the following buttons:

- Click **Enable** to enable the selected rule(s).
- Click **Disable** to disable the selected rule(s).
- Click **Remove** and confirm to delete the selected rule(s).

Applications 14

This section introduces Axway SecureTransport applications and describes how to use the Applications menu features of the SecureTransport Administration Tool.

Application overview

In SecureTransport, *applications* are sets of workflow you can create to perform file processing, including the following:

- Transform data
- Schedule file transfers
- Route files
- Monitor shared folder monitoring across user accounts
- Trigger sequential sets of actions

An application is defined as an instance of a set of workflow called an *application type*. Once you have defined an application, you create a connection between an application and one or more *accounts*. Such a connection is defined through a *subscription*.

SecureTransport ships with the following built-in application types:

- **Account Maintenance** – Automatically deletes, disables, or purges accounts based on their inactivity or age. You can configure account maintenance schedule and emails notifications, as well as email alerts for password and certificate expiration.
- **Advanced Routing** – Provides options to create multiple automated flows for file transformations, routing and transfers between different participants, partner systems, and applications.
- **Archive Maintenance** – Automatically deletes archived files based on a schedule you define.
- **Audit Log Maintenance** – Automatically deletes Audit log records that are older than a specified number of days or months (6 months by default). You can schedule how often to run this application and configure it to optionally export these records prior to deletion.
- **Axway Sentinel Link Data Maintenance** – Removes all SentinelLinkData entries to files that do not exist anymore, based on a schedule you define.
- **Axway Transfer CFT** – Enables Axway Transfer CFT to push files to SecureTransport.
- **Basic Application** – Processes server-initiated transfers and performs data transformations.
- **File Maintenance** – Automatically deletes files from account home folders based on a specified retention or expiration period. You can schedule how often to run this application and configure it to optionally send notification before or/and after file deletion.

- **File Transfer via File Services Interface** – Processes metadata files sent from another system for a protocol implemented using the file services interface.
- **Human to System** – Provides a way to route H2S file transfers.
- **Log Entry Maintenance** – Automatically deletes Server log records that are older than a specified number of days, hours or minutes (1 day by default). You can schedule how often to run this application and configure it to optionally export these records prior to deletion.
- **Login Threshold Maintenance** – Unlocks accounts locked according to the selected "Lock account after N successful logins" option in the *Account settings* and sends a report to specified email contacts.

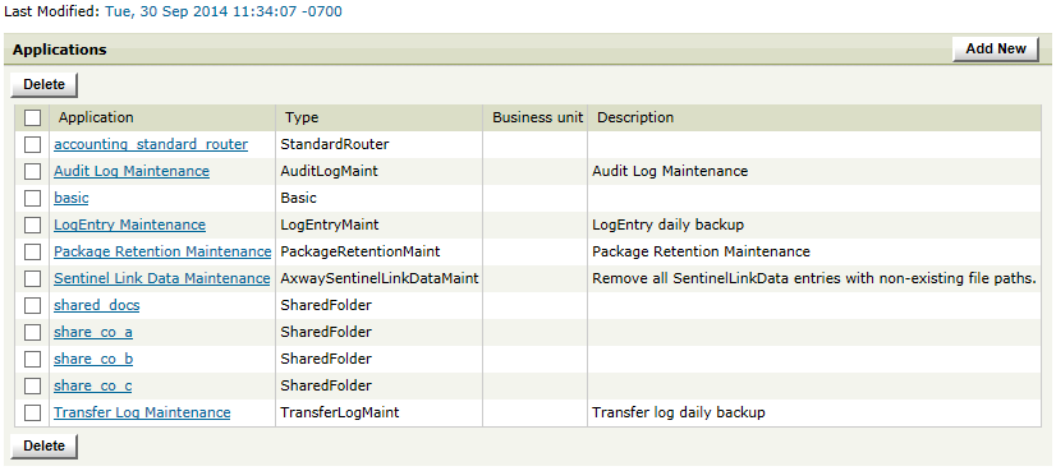
  See *Login Threshold Maintenance application* on page 845.
- **Package Retention Maintenance** – Deletes expired file packages from ad hoc file transfers.
- **Shared Folder** – Provides shared data storage between accounts.
- **Site Mailbox** – Similar to Basic application, however with dedicated outbox and inbox folders for files transfers using a transfer site. This application is recommended for AS2 transfer sites.
- **Standard Router** – Provides basic options to automate flows for file transformations, routing and transfers between an account and internal systems.
- **Transfer Log Maintenance** – Automatically deletes Transfer log records that are older than a specified number of days (30 days by default). You can schedule how often to run this application and configure it to optionally export these records prior to deletion.
- **Unlicensed Accounts Maintenance** – Deletes inactive unlicensed user accounts older than a specified number of days (60 days by default). You can schedule how often to run this application.

Manage applications 14

This subtopic provides instructions on how to access, edit, and delete applications.

Access applications

Select **Application**.
The *Applications* page is displayed. It lists all the applications available currently on the system. Use it to add, delete, view, and edit applications.


Last Modified: Tue, 30 Sep 2014 11:34:07 -0700

Applications Add New

Delete

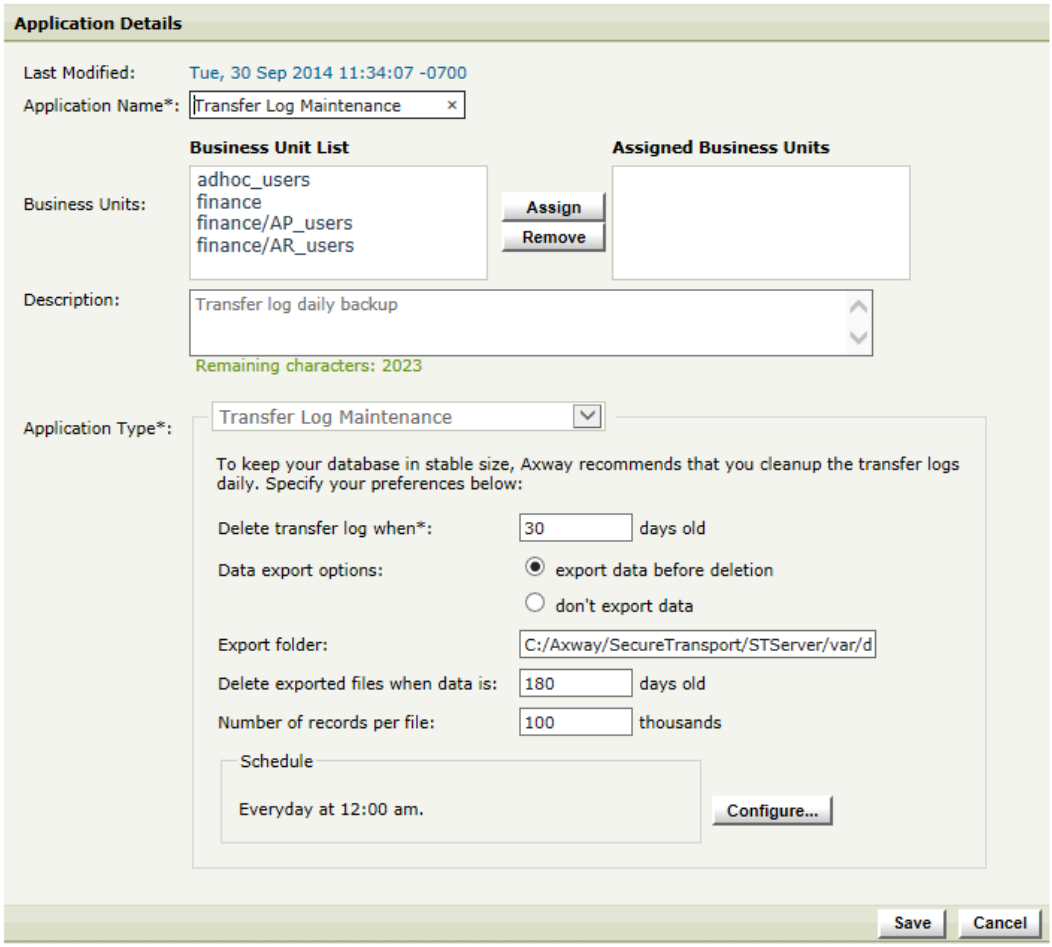
Application	Type	Business unit	Description
accounting_standard_router	StandardRouter		
Audit Log Maintenance	AuditLogMaint		Audit Log Maintenance
basic	Basic		
LogEntry Maintenance	LogEntryMaint		LogEntry daily backup
Package Retention Maintenance	PackageRetentionMaint		Package Retention Maintenance
Sentinel Link Data Maintenance	AxwaySentinelLinkDataMaint		Remove all SentinelLinkData entries with non-existing file paths.
shared_docs	SharedFolder		
share_co_a	SharedFolder		
share_co_b	SharedFolder		
share_co_c	SharedFolder		
Transfer Log Maintenance	TransferLogMaint		Transfer log daily backup

Delete

View or edit an application

Use the following procedure to view or edit an application.

1. Select **Application**.
2. On the *Applications* page, click the name of the application you want to view or edit. The *Application Details* page of the application you selected is displayed.



3. View or edit the information displayed.

   The *Application Details* page varies for the different application types. It dynamically displays the attributes for the respective application type. You can edit all the standard settings of an application, except application type.

4. If you edit the information, click **Save** to preserve the changes.

Delete an application

Use the following procedure to delete an application.

1. Select **Application**. The *Applications* page is displayed.
2. Select the checkbox for the application you want to delete.
3. Click **Delete**.
4. When prompted, confirm that any subscription to this application will be lost.

Note When you delete an application, all subscriptions to this application are removed.

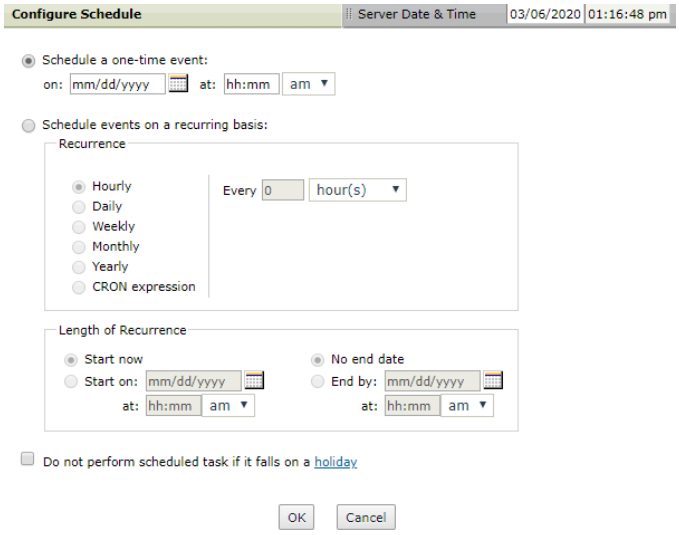
Configure a schedule for a maintenance application

SecureTransport allows you to schedule maintenance events that will be executed once at a specified time in the future or at periodic intervals. You can access the scheduler page by creating or editing any maintenance application.

To schedule a maintenance event:

1. Select **Application** and click **Add New** or select a maintenance application from the list.
2. Scroll down to the *Schedule* pane and click **Configure**.

   The *Configure Schedule* dialog box is displayed.



3. Select the event's timing:
   - One-time event – the event is executed once at the specified date and time.
   - Recurrent – the event is executed at the specified periodic intervals until it is deleted.
     a. Specify *Recurrence* for the event by selecting one of the supported intervals: **Hourly**, **Daily**, **Weekly**, **Monthly**, or **Yearly**. If you select **CRON Expression**, enter your quartz cron expressions in the text box, each on a new line.
     b. In the *Length of Recurrence* pane, select a specific start day and time, end day and time, both, or neither. By default, a recurring event's schedule begins as soon as it is created, and continues indefinitely, until it is disabled.
4. Choose whether the task should be performed if it falls on a day specified as a holiday in the Holiday Schedule. Note that the Holiday Schedule functionality does not allow for executing a scheduled task on the next working day if the specified date happens to be a holiday – when this occurs, the tasks are not executed.
5. Click **OK** when finished setting the schedule.

Note If you configure a schedule and save it after the scheduled start time, the task will not be executed. You must save your configured schedule before the scheduled start time.

Before queuing a new task, the server checks if a previous instance of same periodic task is still pending. If there is a pending instance of the same periodic scheduled task , the new task is not scheduled.

If the server goes down for some time and restarts, the scheduler does not execute any scheduled tasks missed during the server down time.

For information on scheduling file downloads, see *Scheduled downloads and tasks* on page 674.

Create applications

Every application in SecureTransport must be based on an application type. Currently, SecureTransport supports multiple application types. Learn how to create applications of each application type:

- *Archive Maintenance application* on page 826
- *Archive Maintenance application* on page 826
- *Audit Log Maintenance application* on page 828
- *Axway Sentinel Link Data Maintenance application* on page 830
- *Axway Transfer CFT application* on page 831
- *Basic Application* on page 832
- *File Transfer via File Services Interface application* on page 837
- *Human to System application* on page 838
- *Log Entry Maintenance application* on page 839
- *Login Threshold Maintenance application* on page 845
- *Package Retention Maintenance application* on page 846
- *Shared Folder application* on page 847
- *Site Mailbox application* on page 849
- *Standard Router application* on page 850
- *Transfer Log Maintenance application* on page 856
- *Unlicensed Accounts Maintenance application* on page 861
- *File Maintenance application* on page 834

For instructions on creating an Advanced Routing application, see *Create Advanced Routing application* on page 874.

Account Maintenance application

In SecureTransport 5.5, you can define an Account Maintenance policy to automatically delete, disable, or purge accounts based on account inactivity or age. You can configure account maintenance schedule and emails notifications, as well as email alerts for password and certificate expiration.

An Account maintenance policy can be set at three levels:

- Global – by creating an Account Maintenance application without assigning it to a specific business unit. Only one instance of the Account Maintenance application can be created.
- Business Unit – by modifying or disabling the global policy for a specific business unit (**Accounts > Business Unit > Account Maintenance**)
- Account – by modifying or disabling the global or a business unit-level policy for a specific user account (**Accounts > User Account** section **> Account Maintenance**)

An account or a business unit policy takes precedence over the global one. The account-level policy overrides any value defined for the same policy in the Business Unit settings.

Note The Account Maintenance policy does not apply to unlicensed user accounts. For more information, see *Unlicensed Accounts Maintenance application* on page 861

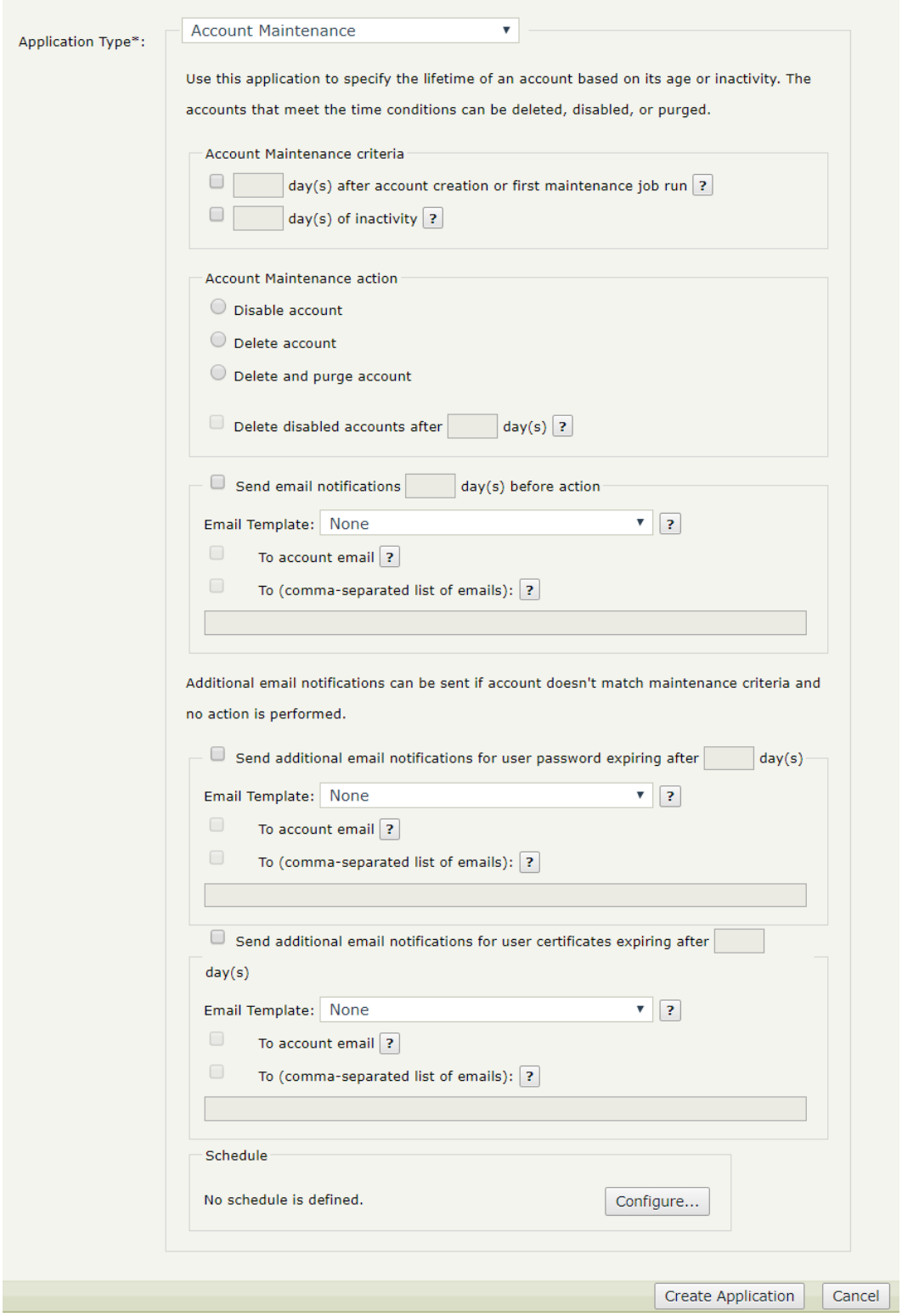
Before you create an Account Maintenance application, make sure that the `AccountMaintenanceApp` rule package is enabled in the Transaction Manager. For more information, see *Manage rule packages* on page 223.

Use the following procedure to create an Account Maintenance application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Accounts Maintenance** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.

4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have *Business units* on page 746. To create a global policy, do not assign a business unit.

   Note If a business unit is assigned to the application, the Account Maintenance policy will be applied ONLY to the accounts in this business unit and will NOT be applied to accounts that don't belong to any business unit.

5. (Optional) Enter an application **Description**.
6. Define the *Account Maintenance Criteria*:
   - account age
     Select the **X day(s) after account creation or first maintenance job run** checkbox, and specify the period, in days, from the account creation date before a maintenance action is performed on that account. If an account does not have a creation date set, the first maintenance job run will be used instead.
   - account inactivity
     Select the **X day(s) of inactivity** checkbox and you specify the time period, in days, from the last login time of an account before a maintenance action is performed on that account. If current account doesn't have last login date set, the first maintenance job run will be used instead.

   Note At Business Unit level, you can also select a specific date for Account Maintenance to execute for all accounts under the business unit.

7. In the *Account Maintenance action* pane, select the action to be performed on the accounts that meet the criteria. You can choose to:
   - **Delete account**
   - **Delete and purge account**: deletes the account and the account home folder
   - **Disable account**
     - When the **Disabled account** checkbox is selected, you can also specify a period, in days, after which the accounts disabled from Account Maintenance will be deleted. When a disabled account awaits deletion, a warning message will be notifying you on the account edit page.
8. Specify when an email notification about an upcoming maintenance action to be sent, its contents and recipients:
   - Select the **Send email notifications X day(s) before action** check-box and specify how many days before the expected action execution date the notification to be sent. You can also input comma-separated values for the notifications period. For example, if you input *1,2,3*, then the email will be sent to user exactly 3 days, 2 days, 1 day before action execution.
     Note The email notification will be sent to current account the first time it matches an Account maintenance criteria.
   - Select an **Email Template** from the drop-down to be used for the notification email. You can configure email template `AccountManagementNotification.xhtml` in **Setup > Mail Templates**.

- Select **To (comma-separated list of emails)** to add a list of email addresses to which the notification to be sent. This option is not available at the account level.
- Select the **To account email** to send the notification to the account's email address.

9. Configure additional email notifications to be sent when the account doesn't match maintenance criteria and no action is performed:
   - Select the **Send additional email notifications for user password that is expiring after X day(s)** checkbox and specify when a password expiration notification to be sent. The timing of your notification is defined as the number of days before the expected password expiration date. You can also input comma-separated values for the notifications period.

     The email password notification is sent to the specified account(s) once a day.

     - Use **Email Template** to select the template for user password expiration emails, and specify the notification recipients.
   - Select **Send additional email notifications for user certificates expiring after X day(s)** checkbox and specify the number of days after which additional email notifications for user certificates expiration will be sent. The timing of your notification is defined as the number of days before expected certificate expiration date. You can also input comma-separated values for the notifications period.

     The email certificate notification will be sent to the specified account (s) once a day.

     - Use **Email Template** to select the template for user certificate expiration emails, and specify the notification recipients.
10. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.
11. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

    Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
12. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Archive Maintenance application

The Archive Maintenance application automatically deletes archived files based on a schedule you define.

You must enable the `ArchiveMaintApp` rule package from the Transaction Manager before you can use an Archive Maintenance application. For more information, see *Manage rule packages* on page 223.

The configuration for an Archive Maintenance application is stored in the database.

Note If the database partition feature is not available because the export database feature is not installed, a warning message will be displayed. For additional information, refer to the *SecureTransport Installation Guide*.

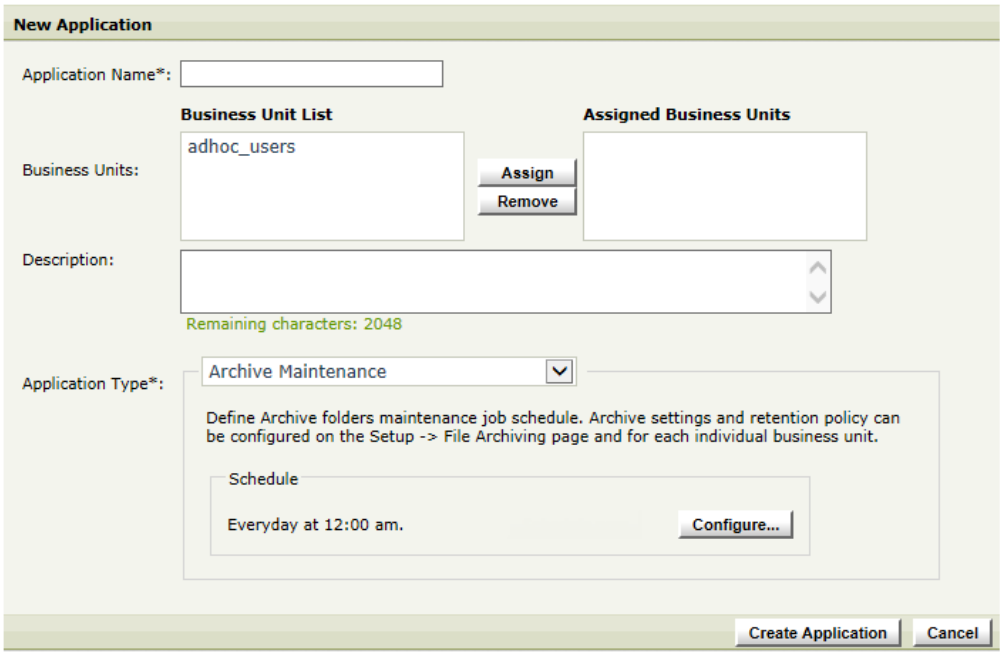
The Archive Maintenance application defines the file archiving maintenance job schedule. Archive settings and retention policy can be configured on the **Setup > File Archiving** page and for each individual business unit.

For information on configuring the file archiving global configuration, see *File archiving global configuration* on page 226.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Archive Maintenance** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.
5. (Optional) Enter an application **Description**.
6. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.
7. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

   Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
8. Click **Create Application**.

To use the Archive Maintenance application, make sure that the `ArchiveMaintApp` and the `ArchiveAgent` rule packages are enabled on the *Transaction Manager Settings on page 217* page.

Enable Multithreading

When the Archive maintenance application is to process a large number of files, it can be executed multi-threaded. To enable multithreading, set the number of threads to execute file deletion in the `FileArchiving.DeleteFiles.ProcessingThreads` configuration option.

Increasing the number of threads increases the load on the storage on which the application operates. The number of threads should not exceed 16.

Set maximum run time

Occasionally, if the Archive Maintenance application is processing a large number of files, it may not be able to finish until the next scheduled occurrence. In this case, it may be advisable to specify the maximum time (in minutes) that you expect the application to run in the `FileArchiving.DeleteFiles.MaximumProcessingTime` configuration option. If you set it to `0`, the application continues to run until it completes.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Audit Log Maintenance application

The Audit Log Maintenance application automatically deletes Audit log records that are older than a specified number of days or months (6 months by default). You can schedule how often to run this application and configure it to optionally export these records prior to deletion.

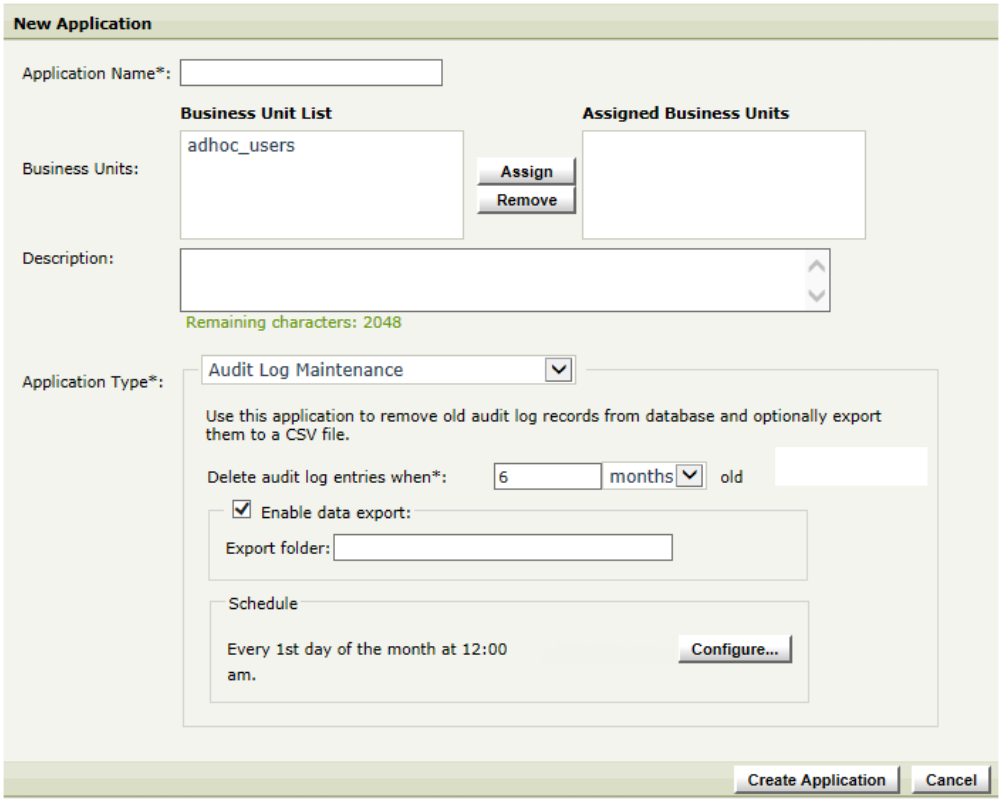
Make sure that the `AuditLogMaintApp` rule package is enabled in the Transaction Manager before you enable an Audit Log Maintenance application. For more information, see *Manage rule packages on page 223*.

Use the following procedure to create an Audit Log Maintenance application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Audit Log Maintenance** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.

   Note As with all applications, assigning business units to the application controls which delegated administrators can manage the application. It does not control which log entries Transfer Log Maintenance processes.

5. (Optional) Enter an application **Description**.
6. In the **Delete audit log entries when** field, specify in days or months how old transfer log entries will be when they are deleted. This field is mandatory.

   You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
7. To export the deleted audit log entries to a file before they are deleted, select **Enable data export**, and, in the **Export folder** field, specify where the export files are stored. You must enter a full directory path.
8. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.

   For information on working with the SecureTransport scheduler , see *Scheduled downloads and tasks* on page 674. Click **OK** when finished setting the schedule.

9. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

   Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.

10. Click **Create Application**.

Configure chunk size

The Audit Log Maintenance application deletes and exports log entries in chunks with a default size of 1000 entries. Processing a large amount of data from the database can be a resource-intensive task, especially if the data set is voluminous. To avoid out of memory errors, you can decrease the chunk size using the `AuditLog.ChunkSize` server configuration option. The option accepts values between 1 and 1000. Values outside of this range are considered invalid and result in server log warnings upon application execution. If the configuration option has an invalid value, the default will be used.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Axway Sentinel Link Data Maintenance application

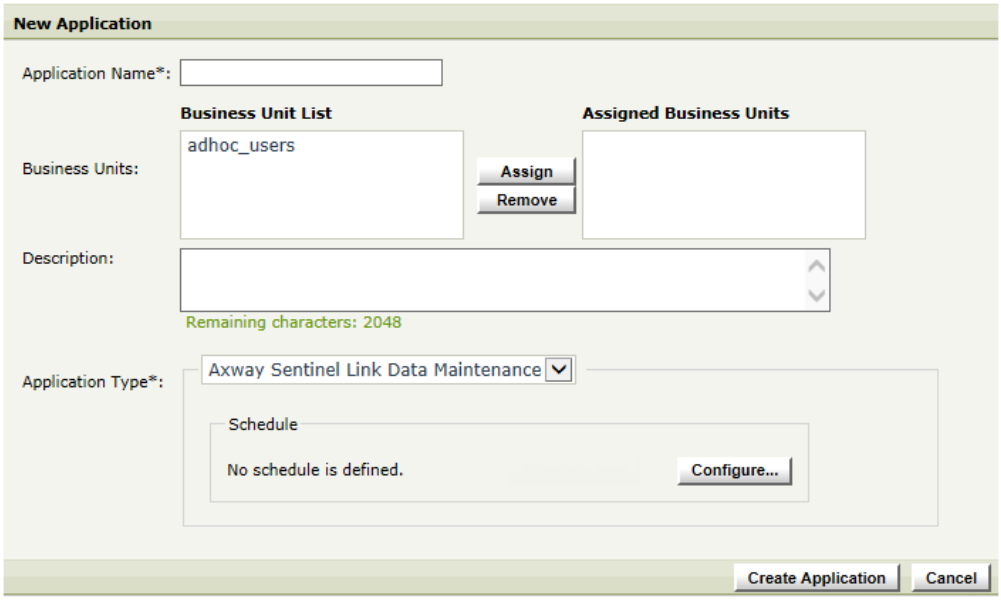
The Axway Sentinel Link Data Maintenance application removes all SentinelLinkData entries to files that do not exist anymore, based on a schedule you define.

Use the following procedure to create an Axway Sentinel Link Data Maintenance type application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Axway Sentinel Link Data Maintenance** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.
5. (Optional) Enter an application **Description**.
6. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.
7. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs. Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
8. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Axway Transfer CFT application

The Axway Transfer CFT application enables Axway Transfer CFT to push files to SecureTransport.

You must enable the `AxwayTransferCFT` rule package in the Transaction Manager before you can use an Axway Transfer CFT application. For more information, see *Manage rule packages on page 223*.

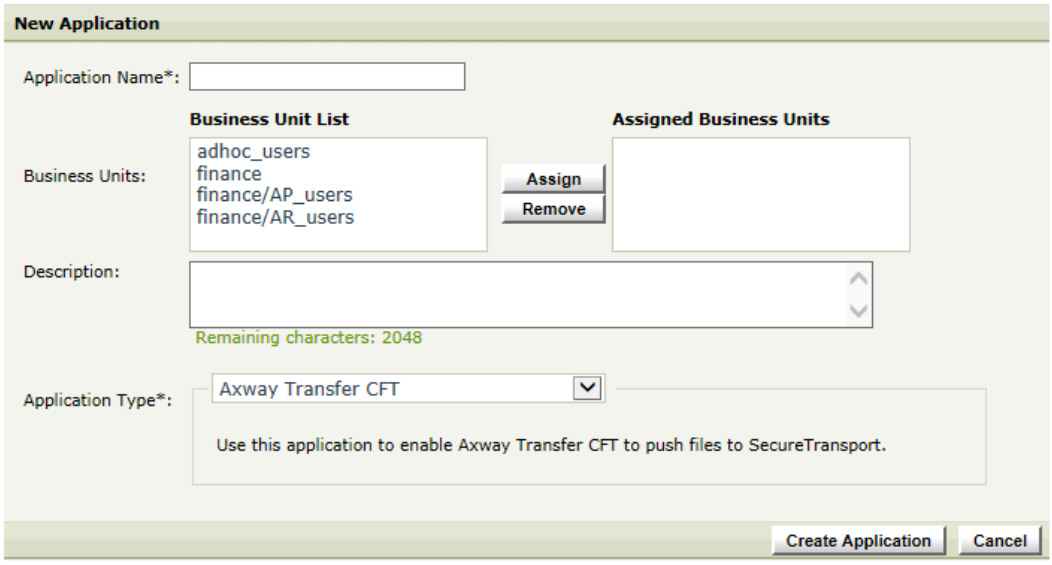
Note This application is not needed for Transfer CFT communication and is used only for legacy configurations.

Use the following procedure to create an Axway Transfer CFT application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Axway Transfer CFT** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.
5. (Optional) Enter an application **Description**.
6. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

   Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
7. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Basic Application

The Basic Application performs server-initiated transfers and data transformations without file routing.

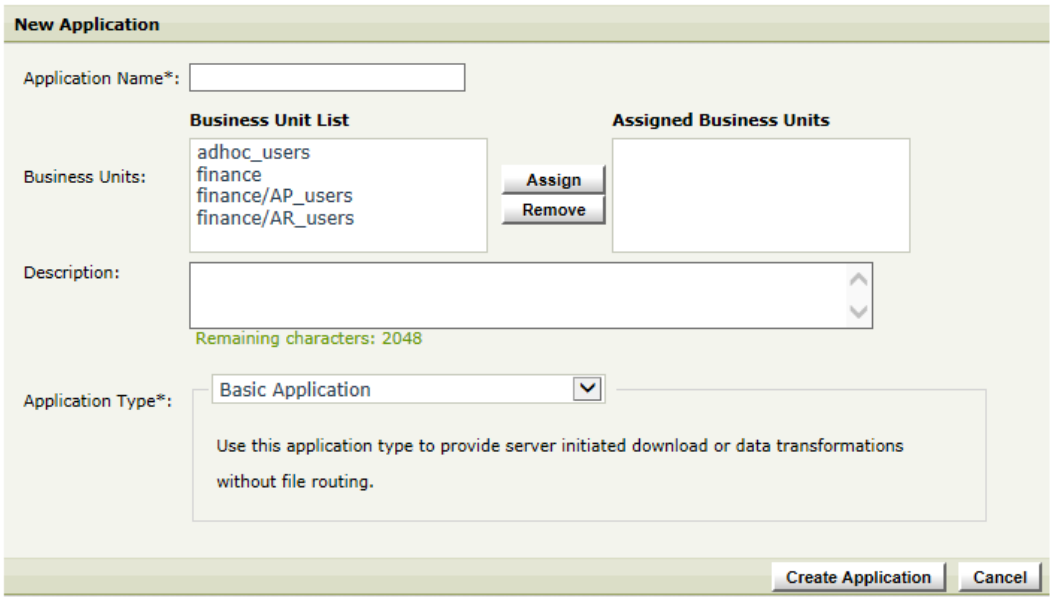
You must enable the `BasicApp` rule package from the Transaction Manager before you can use a Basic Application. See *Manage rule packages* on page 223.

Use the following procedure to create a Basic Application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Basic Application** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.
5. (Optional) Enter an application **Description**.
6. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs. Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
7. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

File Maintenance application

In SecureTransport 5.5, you can create a file maintenance policy that deletes files from the account home folder based on a specified retention or expiration period. You can schedule the maintenance and configure notifications to be sent to specific recipients before or/and after the deletion of files.

A File Maintenance policy can be set at four levels:

- Global – by creating a File Maintenance application without assigning it to a specific business unit. Only one instance of the File Maintenance application can be created.
- Business Unit – by modifying or disabling the global policy for a specific business unit (**Accounts > Business Unit > File Maintenance policy settings**)
- Account – by modifying or disabling the global or a business unit-level policy for a specific account (**Accounts > User Account** section **> File Maintenance policy**)
- Account template – by modifying or disabling the global policy for accounts assigned to a specific account template (**Accounts > Account templates** settings **> File Maintenance policy**)

The global policy is in effect for all accounts that are not assigned to a business unit and all top-level business units. It can be overridden at these two lower levels. When you set a policy on a specific business unit, accounts that are directly under that BU or any child BU inherit that policy. In this case, the inherited policy settings can be overridden at the lower levels only if **Allow File Maintenance policy modifying** is enabled at the parent BU level.

File Maintenance is not performed on the following content:

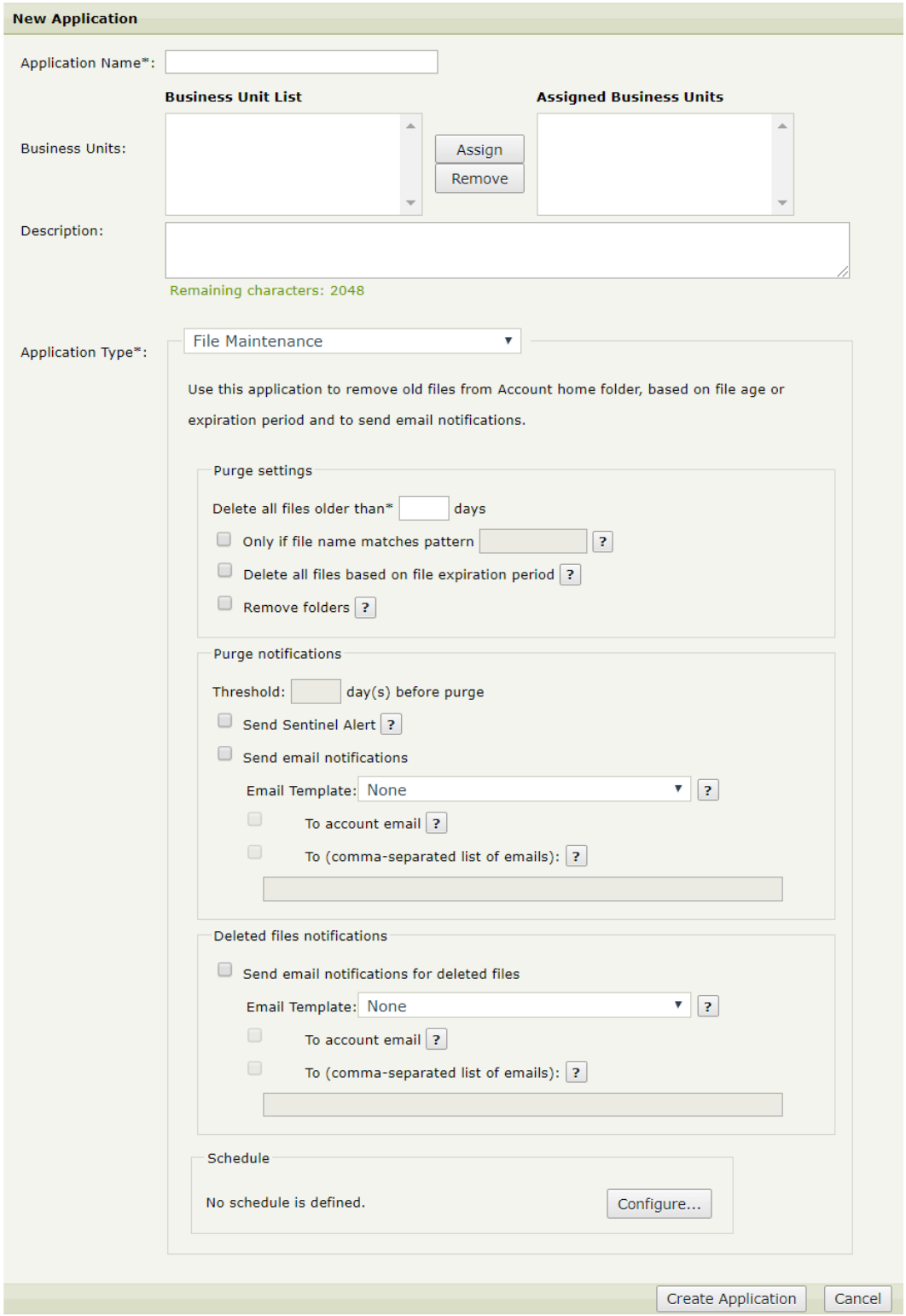
- Anonymous AdHoc accounts configured in **Setup > AdHoc Settings**
- Service accounts
- The account`s mailbox and STFS folders
- Shared folders not owned by the user
- Files in a Shared Folder application subscription folder

Before you create a File Maintenance application, make sure that the `FileMaintenanceApp` rule package is enabled in the Transaction Manager. For more information, see *Manage rule packages* on page 223.

Use the following procedure to create a File Maintenance application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.
2. Select **File Maintenance** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.

4. (Optional) Use the **Assign** and **Remove** buttons to assign business units to the application. The **Business Unit List** contains the names of the business units you have *Business units* on page 746. To create a global policy, do not assign a business unit.

   Note If a business unit is assigned to the application, the File Maintenance policy will be applied ONLY to the accounts in this business unit and will NOT be applied to the accounts that don't belong to any business unit.

5. (Optional) Enter an application **Description**.
6. In the *Purge settings* pane:
   a. In the **Delete all files older than** field, specify the number of days files should be retained in the account home folder. This field is mandatory.

      You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.

   b. Select the **Only if file name matches pattern** checkbox and specify a file name pattern to identify files to be deleted.
   c. Select the **Delete all files based on file expiration period** checkbox to enable the deletion of files based on their expiration period. The expiration period is set in epoch time (in milliseconds) as a flow attribute named `EXPIRE.ON`. If an expiration period is not set for a file, it will be deleted based on the retention period set in the File Maintenance application.
   d. Select the **Remove folders** checkbox to enable the deletion of any subfolder of the account home folder that has been left empty after file maintenance.

      Note File Maintenance will not delete empty Subscription folders, except in the case that the users are assigned to an account template and their home folders are constructed using Expression Language.

7. (Optional) In the *Purge notifications* pane, select the notification method and threshold:
   a. Select the **Send to Sentinel Alert** checkbox to enable sending of a *TO_BE_DELETED* state to Sentinel. To avoid event redundancy, even if the application is configured to run several times a day, only one *TO_BE_DELETED* state will be sent to Sentinel.
   b. Select **Send email notifications** to enable the sending of email notifications. Then, specify the contents and the recipients of the notification:
      - Select the **Email Template** from the drop-down to be used for the pending file deletion email. For details on configuring email templates, see *Mail templates* on page 195.
      - Select the **To account email** to send a pending file deletion notification to the account's email address. The user will receive one email per day containing all files pending for deletion.
      - Select **To (comma-separated list of emails)** to add a list of email address to which the notification to be sent. The notification will contain all files pending for deletion files per each account. It will be sent upon each execution of the application. This option is not available at the account level.

c. In the **Threshold** field, specify the file age, after which a notification to be send. This field is active only after a notification method is selected. The threshold value should be either a positive integer less than the one specified in **Delete all files older than** or a comma-separated list of positive integers.

8. In the *Deleted files notifications*, select **Send email notifications for deleted files** to enable the sending of an email report on deleted files. Then, specify the contents and the recipients of the notification:
   a. Select the **Email Template** from the menu to be used for file deletion reports. For details on configuring email templates, see *Mail templates* on page 195.
   b. Select **To account email** to send a list of the deleted files to the account email. The notification is sent once per day, even if the application is configured to run several times a day. This setting can be overridden at a business unit level only.
   c. Select **To (comma-separated list of emails)** to add a list of email addresses to which the notification to be sent. A notification is sent upon each execution of the application and contains all the files deleted files per account. This option is not available at the account level.
9. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.
10. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.
    Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
11. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

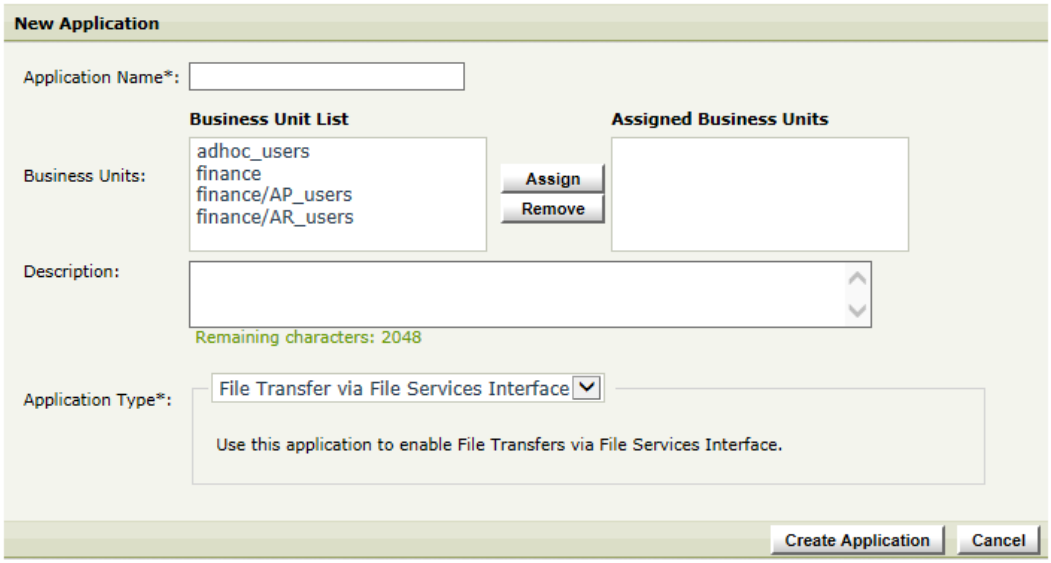
File Transfer via File Services Interface application

The File Transfer via File Services Interface application processes metadata files sent from another system for a protocol implemented using the file services interface.

You must enable the `FileServicesInterface` rule package from the Transaction Manager before you can use a File Transfer via File Services Interface application. For more information, see *Manage rule packages* on page 223.

Use the following procedure to create a File Transfer via File Services Interface application.

1. Select **Application** and click **Add New**.
   The *New Application* page is displayed.
2. Select **File Transfer via File Services Interface** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.
5. (Optional) Enter an application **Description**.
6. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs. Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
7. Click **Create Application**.

For more information about configuring transfers using a file services interface protocol, see *File services interface transfers* on page 1039.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Human to System application

The Human to System application provides a way to route H2S file transfers.

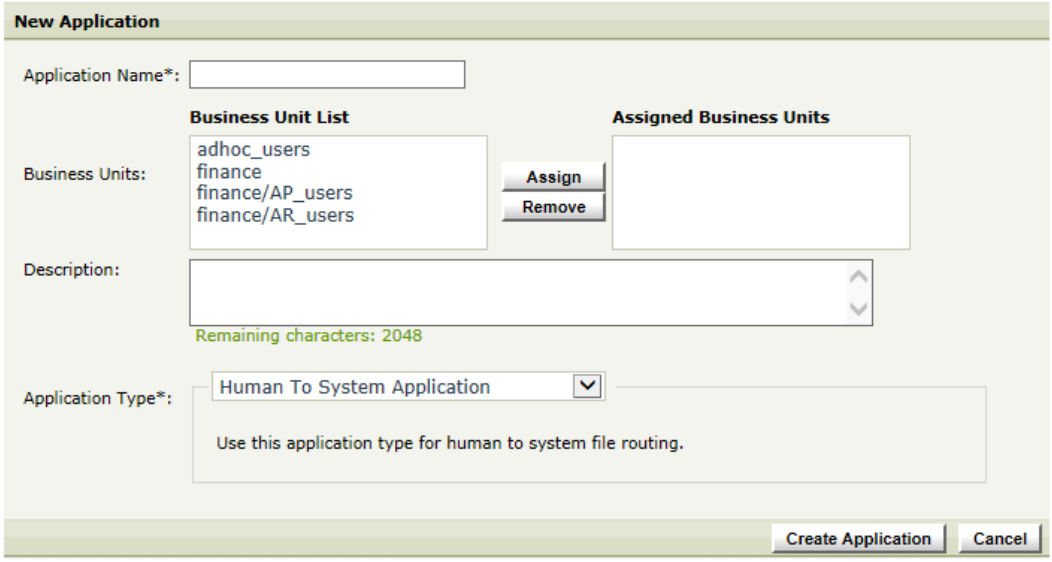
You must enable the `HumanSystem` rule package from the Transaction Manager before you can use a Human to System application. For more information, see *Manage rule packages* on page 223.

Use the following procedure to create a Human to System application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Human To System Application** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.
5. (Optional) Enter an application **Description**.
6. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs. Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
7. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Log Entry Maintenance application

The Log Entry Maintenance application automatically deletes server log records that are older than a specified number of days. You can schedule how often it should run and configure it to export records before deletion.

To create a Log Entry Maintenance application, you must first enable the `LogEntryMaintApp` rule package from the Transaction Manager. For more information, see *Manage rule packages* on page 223.

Partitioning

For all databases, the `Partition.DaysToPrebuild` server configuration option determines the number of days that partitions will be pre-created in advance. For heavily loaded environments, Axway recommends that you set a value greater than 7, to help avoid a serious deadlock in case the needed partition is not created. If this option is not set early on (right after installation), the server log may eventually become full and cause the Administration Tool to stop responding. In this case, you must manually create the partitions, configure `Partition.DaysToPrebuild`, and export the log.

On Oracle and Microsoft SQL Server

If you leave `Partition.DaysToPrebuild` empty (default), the Log Entry maintenance application will create partitions with the following considerations:

- if there is an upcoming scheduled maintenance job, partitions will be created daily for the number of days left before the scheduled job plus 1 additional day;
- if there are no upcoming scheduled jobs, partitions will be created for 1 day ahead.

The maximum value of the `Partition.DaysToPrebuild` option is `30`, even if set higher in the Administration Tool. In order to specify a higher number of days to pre-build partitions, the `partitionsMaxNumberToPrebuild` system property must be added in the `start_tm_console` script with the following format:

```
JAVA_OPTS=" -DpartitionsMaxNumberToPrebuild=<desired number of days> $JAVA_OPTS"
```

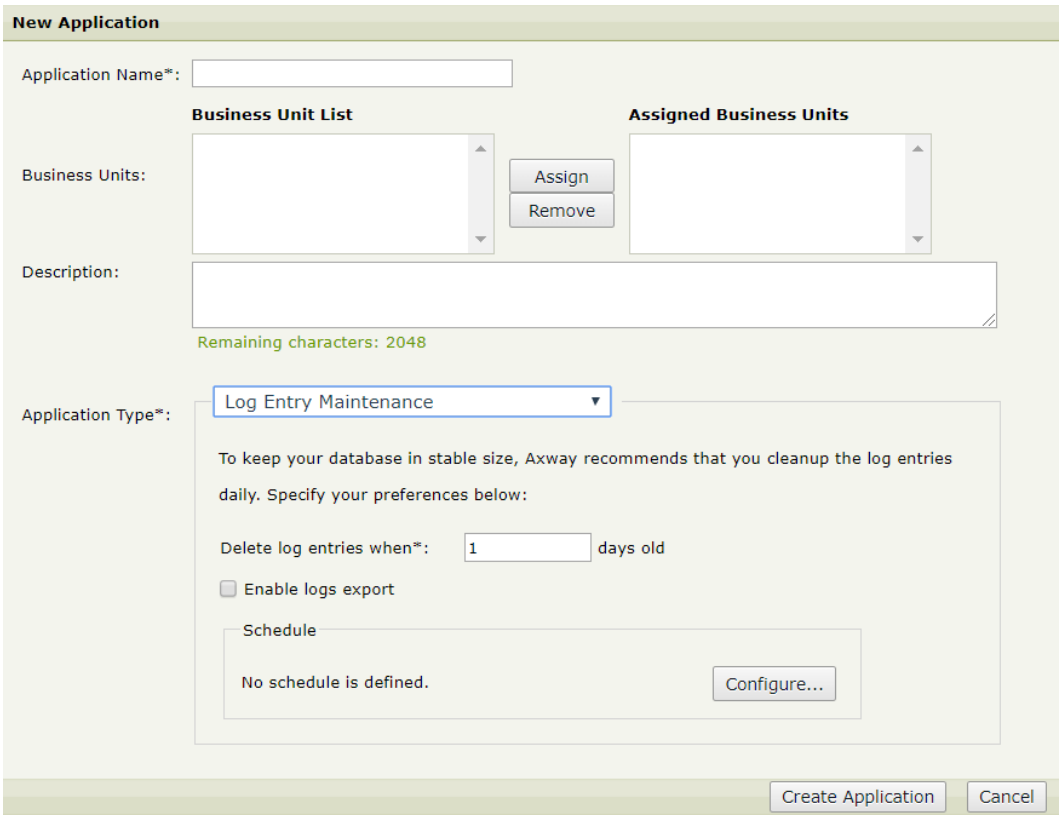
On PostgreSQL

The Log Entry maintenance application does not create new partitions; instead, they are created by a dedicated service. See *Change external PostgreSQL configuration and manage partitioning on page 104*.

Create a Log Entry Maintenance application

Use the following procedure:

1. Select **Application** and click **Add New**.
2. Select **Log Entry Maintenance** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units to the application, see *Business units* on page 746.
5. (Optional) Enter a **Description** for the application.
6. In the **Delete log entries when** field, specify the period after which log entries will be deleted from the database. For external databases, the retention period is in days. For embedded databases, you can specify a period in days, hours, or minutes.

   You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
7. (Optional) Configure the export of server log entries before deletion. The procedure differs depending on the database you use, see *Configure Server log records export before deletion* on page 842
8. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.
9. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

   Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
10. Click **Create Application**.

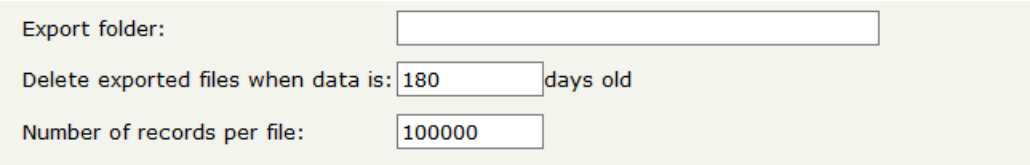
Configure Server log records export before deletion

You can configure the Log Entry Maintenance application to export old server log records before deleting them. This procedure differs depending on the database.

You can also use the `log_export` command line utility to export server log entries.

Export from an embedded MariaDB database

To configure the export of server log records from an embedded database, use the following procedure:



1. In the **Export folder** field, enter the full path to a directory where exported files will be stored.
2. In the **Delete exported files when data is _ days old** field, specify the number of days that exported files will remain in the export directory before they are deleted. If you leave this field empty or specify 0, SecureTransport will not delete the files.
3. In the **Number of records per file** field, specify the maximum number of records that can exist in an exported file. When this value is exceeded, SecureTransport will start exporting the transfer log entries in a new file.

Export from a Microsoft SQL Server database

Use the Microsoft SQL Server export functionality with caution in regard to your backup strategy.

To configure the application to export the server log records before deletion, use the following procedure:

1. Select **Enable logs export**.
2. In the **Export folder** field, enter the absolute path to a directory where the exported files will be stored.
3. Complete the following steps on the Microsoft SQL Server:

a. Create an export directory.

b. Create a new filegroup:

```
ALTER DATABASE databaseuser ADD FILEGROUP [ST_SERVERLOG_
ARCHIVE]
```

c. Create a file:

```
ALTER DATABASE databaseuser
     ADD FILE
     (NAME = N'ST_SERVERLOG _ARCHIVE_databaseuser',
     FILENAME = N'<EXPORT_DIR>\ST_SERVERLOG_ARCHIVE_
databaseuser.ndf',
     SIZE = 10MB,
     MAXSIZE = 100MB,
     FILEGROWTH = 1MB)
     TO FILEGROUP [ST_SERVERLOG_ARCHIVE]
GO
```

d. Grant the database user all permissions to the `<EXPORT_DIR>` directory created in step **a**.

e. Grant *Backup database* and *Backup log* permissions to the database user. The *Backup log* permission is only required if the database is in *Full recovery mode*.

Export from an Oracle database

When your server uses an Oracle database, SecureTransport uses a partitioned table for the log entries. Your Oracle DBA can implement data export via an Oracle functionality. If export database procedures are not deployed, the **Enable logs export** checkbox is disabled.

1. Select the **Enable logs export** checkbox.
2. In the **Export folder** field, enter the absolute path to a directory where the exported files will be stored. You must fill in the name of the directory as defined in the Oracle database (for example, `ST_DMPDIR`).
3. Complete the following steps on the database server on the Oracle server:
   a. Create the directory where logs will be exported and make sure that the Oracle user has permissions.
   b. Log in into Oracle as SYSDBA and create the `ST_DMPDIR` directory using the following syntax:

```
CREATE DIRECTORY ST_DMPDIR AS '/YOUR_DIRECTORY_HERE';
```

c. Grant all privileges on the directory to the SecureTransport user:

```
GRANT ALL PRIVILEGES ON DIRECTORY ST_DMPDIR TO ST_
DATABASE_USER
```

d. Grant create table privileges to the SecureTransport user:

```
GRANT CREATE TABLE TO ST_DATABASE_USER;
```

4. In the **Parallelism Degree** field, specify the number of processors to use during an export operation. You can specify any value from `1` to the number of processors available on the server. You can limit the effect of the export on database performance by limiting the number of used processors.

Export from a PostgreSQL database

With PostgreSQL, SecureTransport uses partitioned tables for storing log data. During installation, three tables are created for storing the server log data: *logging_event*, *logging_event_exception*, and *logging_event_property*. Each table is partitioned daily.

For exporting records from a PostgreSQL database, SecureTransport uses the *pg_dump* utility that ships with PostgreSQL.

Use the following procedure to configure the application to export old server log records before deletion. Note that the database user must be either a superuser or a member of `pg_execute_server_program`.

1. Make the local socket connections trusted or password protected (an encrypted local connection for exports is not supported).

   On the PostgreSQL server, open the *pg_hba.conf* file for editing and modify or add the following line:

   - on Unix-based platforms: `local all all trust` or `local all all password`)
   - on Windows OS: `host all all 127.0.0.1/32 trust` or `host all all 127.0.0.1/32 password`

2. On the PostgreSQL server, create the directory where the logs will be exported and make sure that the PostgreSQL user has permissions.
3. In the Log Entry Maintenance application settings, select the **Enable logs export** checkbox.
4. In the **Export folder** field, enter the absolute path to the directory you created in step 2.
5. In the **Path to pg_dump utility** field, enter the absolute path to the *pg_dump* utility including the file name (`pg_dump.exe` on Windows, `pg_dump` on Unix). Usually it is in the

PostgreSQL's `bin` directory on the filesystem of the database server.

6. Save the application settings.

Login Threshold Maintenance application

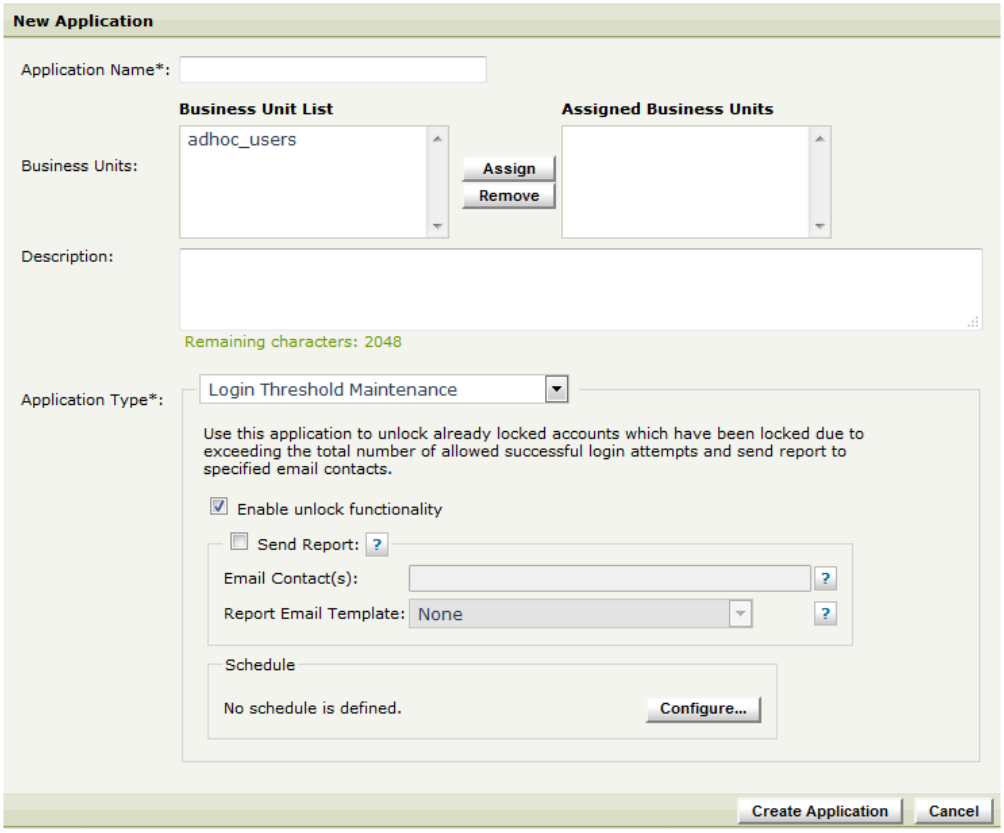
The Login Threshold Maintenance application unlocks accounts locked according to the selected "Lock account after N successful logins" option in the *Account settings* and sends a report to specified email contacts.

Use the following procedure to create a Login Threshold Maintenance application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Login Threshold Maintenance** from the mandatory **Application Type** list.



3. Enter an unique **Application Name** of up to 80 characters. You cannot enter forward slashes (`/`) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.

4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.

5. (Optional) Enter an application **Description**.

6. (Optional) Select **Enable unlock functionality**. **Enable unlock functionality** is selected by default.
7. (Optional) Select **Send Report**. If **Send Report** is selected, a report will be sent to the specified email addresses.
    - If **Enable unlock functionality** is selected, the report will contain a list of unlocked users.
    - If **Enable unlock functionality** is not selected, the report will contain a list of locked, due to login threshold functionality, users.
8. Enter the email address or addresses to deliver the report to in the **Email Contact(s)** field. Email addresses can be separated by either a comma or a semicolon.
9. Select the email template for the report from the **Report Email Template** list. The `LoginThresholdReport.xhtml` template is the default template for the Login Threshold Maintenance application.
10. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.
11. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

    Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
12. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Package Retention Maintenance application

The Package Retention Maintenance applicaiton deletes expired file packages from ad hoc file transfers.

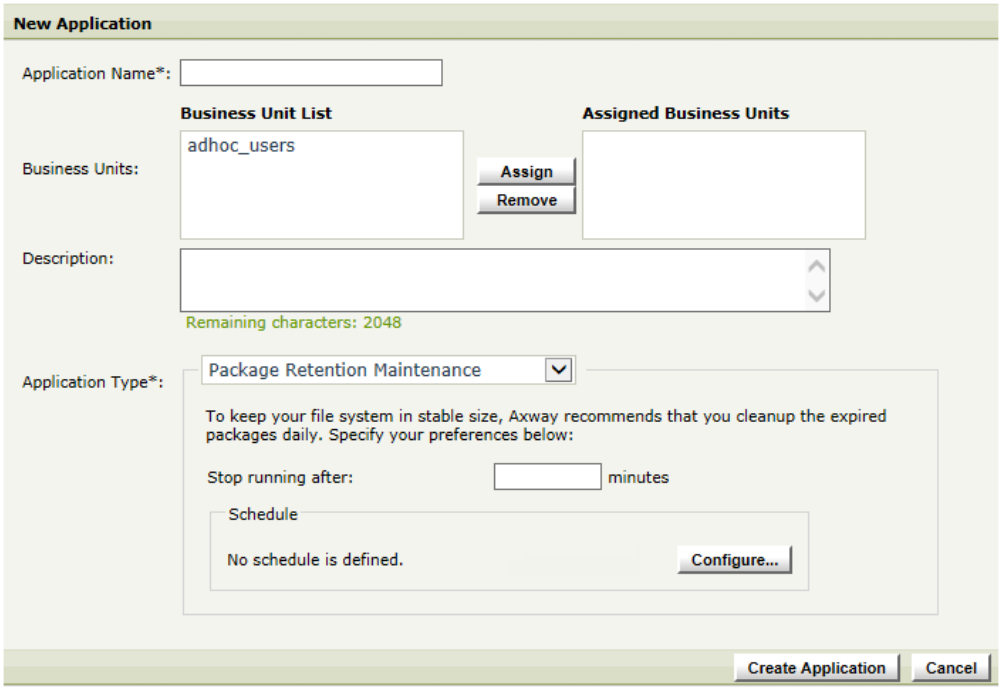
You must enable the `PackageRetentionMaintApp` rule package from the Transaction Manager before you can use a Package Retention Maintenance application. For more information, see *Manage rule packages* on page 223.

Note It is still possible to download anonymous attachments after message expiration, if they have not yet been deleted by any scheduled Package Retention Maintenance Applications.

Use the following procedure to create a Package Retention Maintenance application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.
2. Select **Package Retention Maintenance** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.
5. (Optional) Enter an application **Description**.
6. In the **Stop running after: _____ minutes** field, enter the maximum number of minutes the application runs each time it is started.
7. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.
8. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs. Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
9. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Shared Folder application

The Shared Folder application provides shared data storage between accounts.

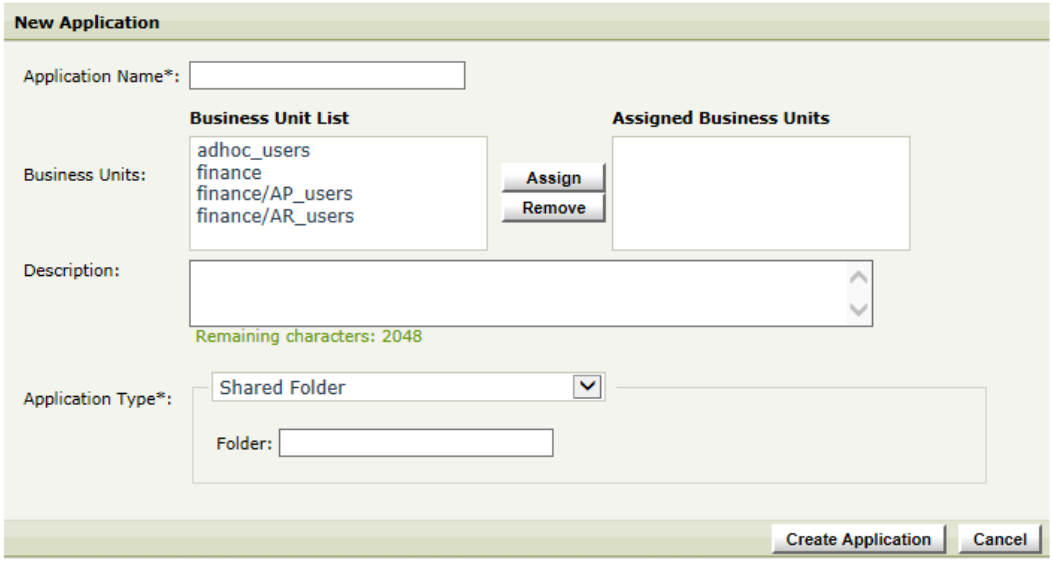
You must enable the `SharedFolder` rule package in the Transaction Manager before you can use a Shared Folder application. For more information, see *Manage rule packages on page 223*.

All users of a shared folder must be either repository encryption users or not repository encryption users.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Shared Folder** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (`/`) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.

4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.

5. (Optional) Enter an application **Description**.

6. In the **Folder** field, type the full path of the folder that you want to share for the accounts that subscribe to the new application.

   You cannot use the following characters in the folder path or name: `* < > ? " / \ | :`

   > **Note** Under Windows, you use Windows-style paths when you specify a shared folder. Denote drives as `C:\` and `D:\`.

7. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

   Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.

8. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Site Mailbox application

The Site Mailbox application is similar to the Basic application, however with dedicated outbox and inbox folders for files transfers using a transfer site. This application is recommended for AS2 transfer sites.

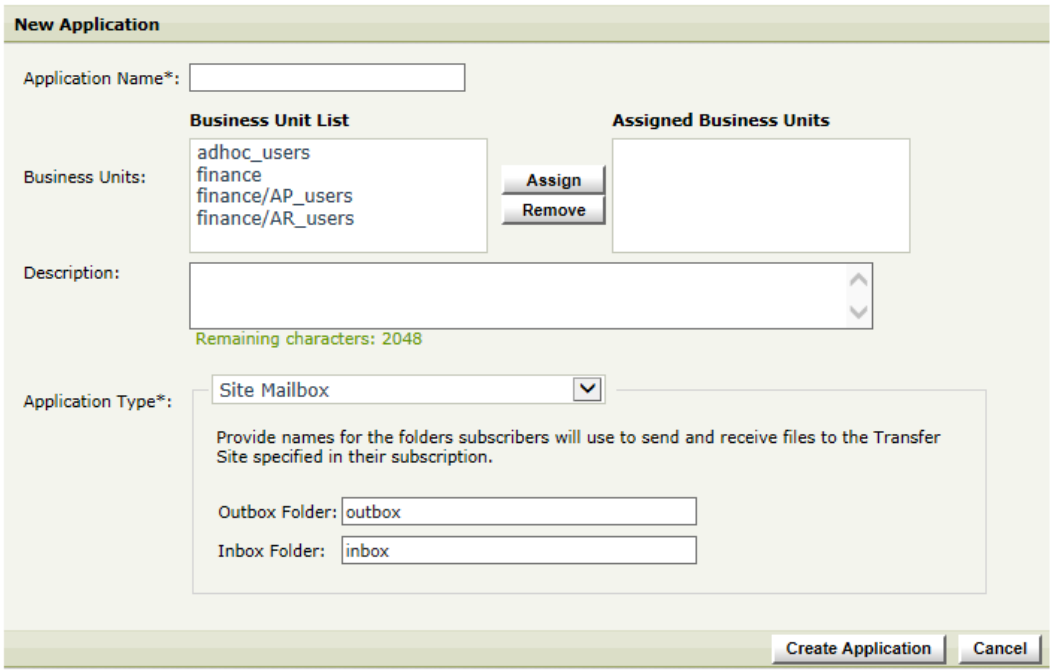
You must enable the `SiteMailbox` rule package in the Transaction Manager before you can use a Site Mailbox application. For more information, see *Manage rule packages on page 223*.

When a user account is subscribed to the Site Mailbox application, there are four subscription settings possible. Require Valid Signature and Require File Encryption are applied to the subscription folder for both incoming and outgoing transfers. Encrypt Files and Sign Files are applied only for the outgoing transfers

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Site Mailbox** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (`/`) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.

4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units on page 746*.

5. (Optional) Enter an application **Description**.
6. Under **Outbox Folder**, type the name of the folder subscribers use to send files. The default value is `outbox`.
7. Under **Inbox Folder**, type the name of the folder subscribers use to receive files. The default value is `inbox`.
8. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs. Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
9. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Standard Router application

The Standard Router application provides basic options to automate flows for file transformations, routing and transfers between an account and internal systems. It is used for file transfer only within an organization, and requires a service account.

1. Create a service account.

2. Create an instance of SR application, lets say SR-app where you select the service account that will use the application.

Put in folder and Get form folder are created under the service account home folder.

3. Create a user account and subscribe it to the app

Before you can use a Standard Router application, you must enable the `StandardRouter` rule package from the Transaction Manager. For more information, see *Manage rule packages* on page 223.

Note A Standard Router application triggers a schedule, even if it is not subscribed to a user account.

When a Standard Router application is used, a file integrity check of receipts generated generates a message that there was no successful MDN comparison in the following cases:

- When a file is uploaded in the user account outbox folder: The file is moved to the submit folder of the service account.
- When the service account pulls a file from its transfer site: The file is moved from the receive folder of the service account to the inbox folder of the user account.

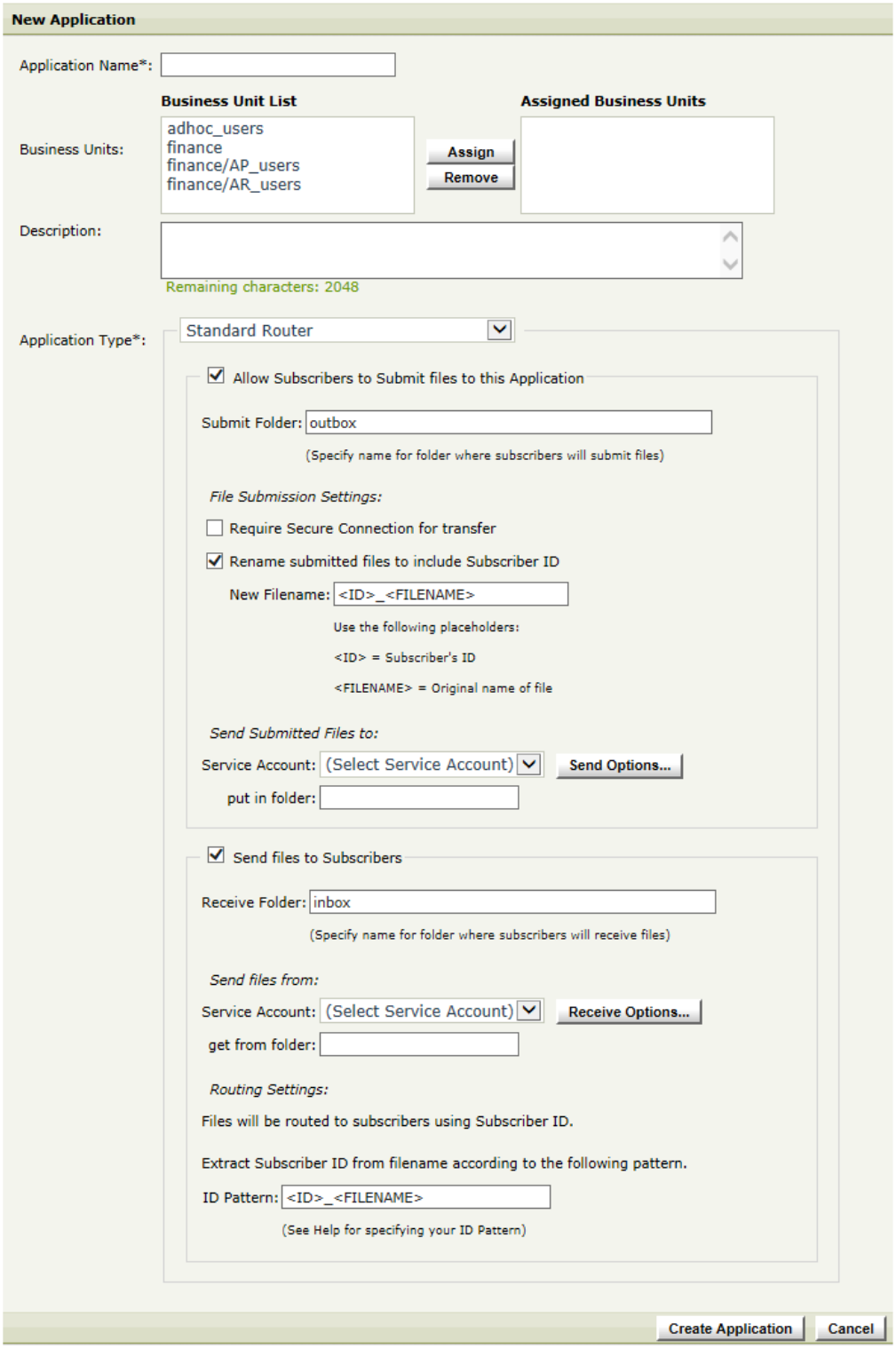
Note You can set up the Standard Router as a one-way configuration by selecting only one of the two choices: **Allow Subscribers to Submit files to this Application** or **Send files to Subscribers**. When you set up a one-way configuration, you do not need to specify a folder.

Use the following procedure to create a Standard Router application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select **Standard Router** from the mandatory **Application Type** list.

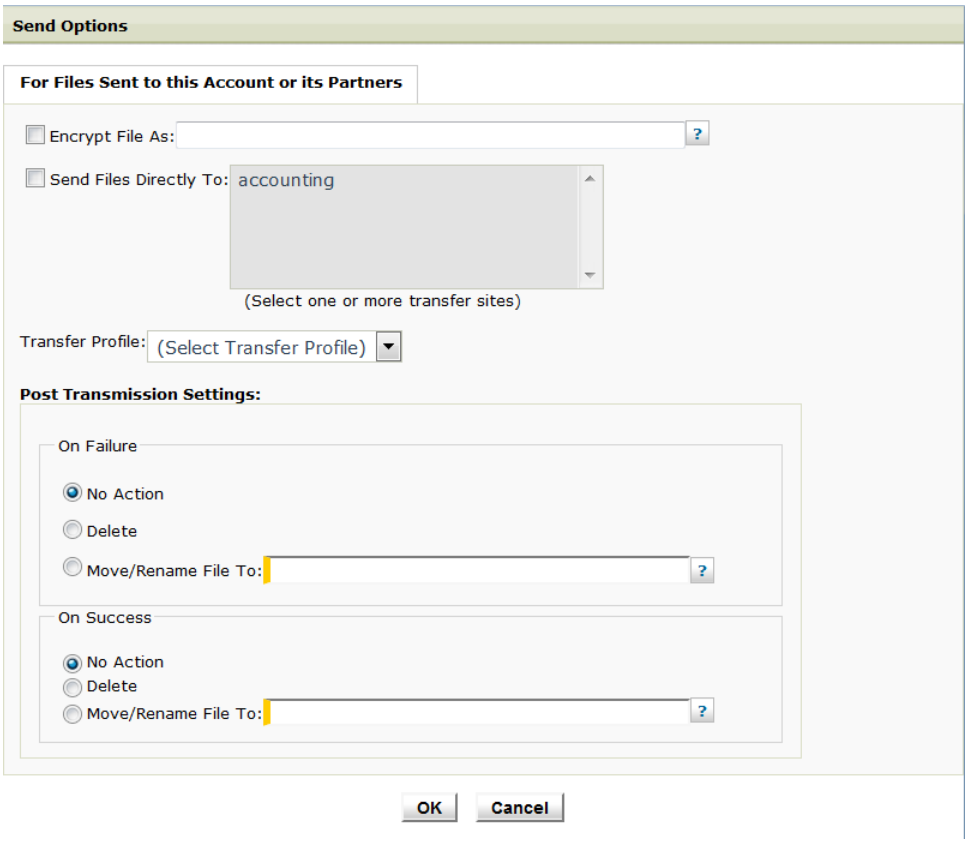


3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.

4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.
5. (Optional) Enter an application **Description**.
6. (Optional) Select **Allow Subscribers to Submit files in this Application** to permit incoming file transfers from the subscriber parties to the application. If you enable this option, continue specifying values for the parameters in the pane.
    a. In the **Submit folder** field, type the name of the folder where incoming transferred files are submitted via subscriptions. The application only processes files stored in the submit folder. Any files stored outside the submit folder are not routed for transferring. The submit folder is created as a sub-folder of the subscription folder. The subscription folder is specified during the creation of the respective subscription. For details, see *Transfer sites* on page 540.
    b. In the **File Submission Settings** group, select the **Require Secure Connection for transfer** option to enable SSL for the incoming transfers.
    c. In the **File Submission Settings** group, select the **Rename submitted files to include Subscriber ID** option to add a prefix to the file name identifying the sender before it is sent to the internal system. The subscriber ID is specified during the creation of the respective subscription.

       Then, in the **New Filename** field, define the format of the new file name. By default, the file is renamed in the format `<ID> <FILENAME>` where, `<ID>` is the Subscriber ID specified when the subscription is created and `<FILENAME>` is the original name of the transferred file.

       The use of the placeholders, `<ID>` and `<FILENAME>`, in the new file name format is mandatory. The character you use to separate `<ID>` from `<FILENAME>` must not be included in the `<ID>` string.
    d. In **Service Account**, select a service account to which you want to send submitted files.
    e. In the **put in folder** field, type the name of the folder to be used by the service account you specified.
    f. (Optional) Click **Send Options** to display the *Send Options* dialog box.



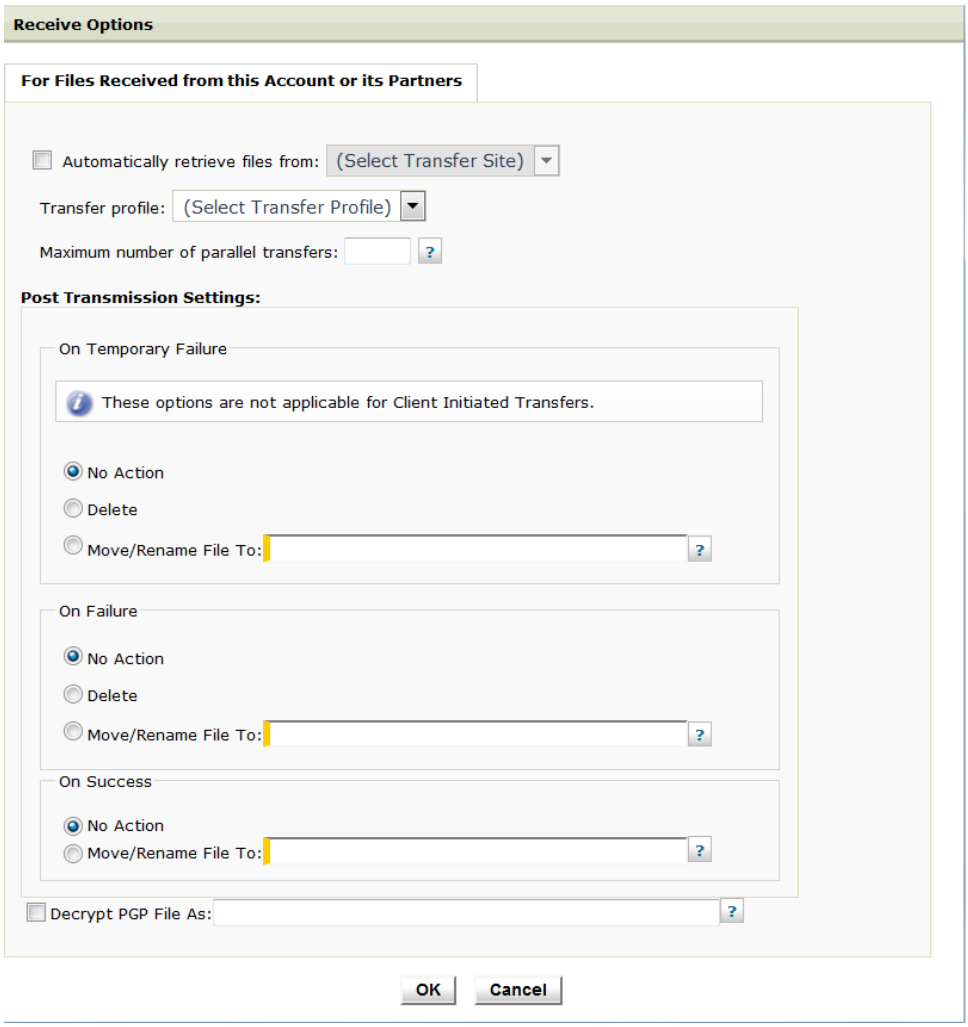
Define the settings for sending files to the service account. Choose one or more of the following options, and then click **OK** in the *Send Options* dialog box.

Encrypt File As – Select this checkbox to require that files submitted are encrypted.

Send files directly to <transfer site> – Select this checkbox to send files directly to the transfer site you select from the drop-down list.

Post Transmission Settings – Select which action you want SecureTransport to take when the transfer fails or succeeds.

7. (Optional) Select the **Send files to Subscribers** checkbox to permit outgoing file transfers from the application to the subscribed parties. If you enable this option, continue specifying values for the parameters in the pane.
    a. In the **Receive folder** field, type the name of the folder where outgoing transferred files are submitted to the subscriber. The receive folder is created as a sub-folder of the subscription folder.
    b. Select a service account to receive files from in the **Service Accounts** list.
    c. In the **get from folder** field, type the name of the folder to be used by the service account you specified.
    d. (Optional) Click **Receive Options** to display the *Receive Options* dialog box and configure the settings for receiving files from this service account. Choose one or more of the following options and enter the maximum number of parallel transfers, and then click **OK** in the *Receive Options* dialog box.



Automatically retrieve files from – Select this checkbox and select a transfer site from the drop-down list to automatically retrieve files from the transfer site when they arrive.

Maximum number of parallel transfers – Enter a number to limit the number parallel transfers. If you enter a value greater than zero, SecureTransport executes only the specified number of transfers in parallel. If the value is null or zero, the maximum number of parallel transfers is limited by system capacity.

Post Transmission Settings – Select which action you want SecureTransport to take when the transfer has a temporary failure, a permanent failure, or succeeds.

Decrypt PGP File As – Select this checkbox to require that files are decrypted after the transfer is complete.

e. In the **Routing Settings** group, specify a pattern in the **ID Pattern** box to define the ID of the subscriber to whom files are routed. By default, the pattern is `<ID>_<FILENAME>` where, `<ID>` is a regular expression corresponding to the Subscriber ID specified when the subscription is created and `<FILENAME>` is the original name of the transferred file.

The use of the placeholders, `<ID>` and `<FILENAME>`, in the new file name format is mandatory.

8. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs. Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
9. Click **Create Application**.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Transfer Log Maintenance application

The Transfer Log Maintenance application automatically deletes transfer log records that are older than a specified number of days. You can schedule how often it should run and configure it to export records before deletion.

To create a Transfer Log Maintenance application, you must first enable the `TransferLogMaintApp` rule package from the Transaction Manager. For more information, see *Manage rule packages* on page 223.

Partitioning

For all databases, the `Partition.DaysToPrebuild` server configuration option determines the number of days that partitions will be pre-created in advance. For heavily loaded environments, Axway recommends that you set a value greater than 7, to help avoid a serious deadlock in case the needed partition is not created. If this option is not set early on (right after installation), the transfer log may eventually become full and cause the Administration Tool to stop responding. In this case, you must manually create the partitions, configure `Partition.DaysToPrebuild`, and export the log.

On Oracle and Microsoft SQL Server

If you leave `Partition.DaysToPrebuild` empty (default), the Transfer Log maintenance application will create partitions with the following considerations:

- if there is an upcoming scheduled maintenance job, partitions will be created daily for the number of days left before the scheduled job plus 1 additional day;
- if there are no upcoming scheduled jobs, partitions will be created for 1 day ahead.

The maximum value of the `Partition.DaysToPrebuild` option is `30`, even if set higher in the Administration Tool. In order to specify a higher number of days to pre-build partitions, the `partitionsMaxNumberToPrebuild` system property must be added in the `start_tm_console` script with the following format:

```
JAVA_OPTS=" -DpartitionsMaxNumberToPrebuild=<desired number of
days> $JAVA_OPTS"
```

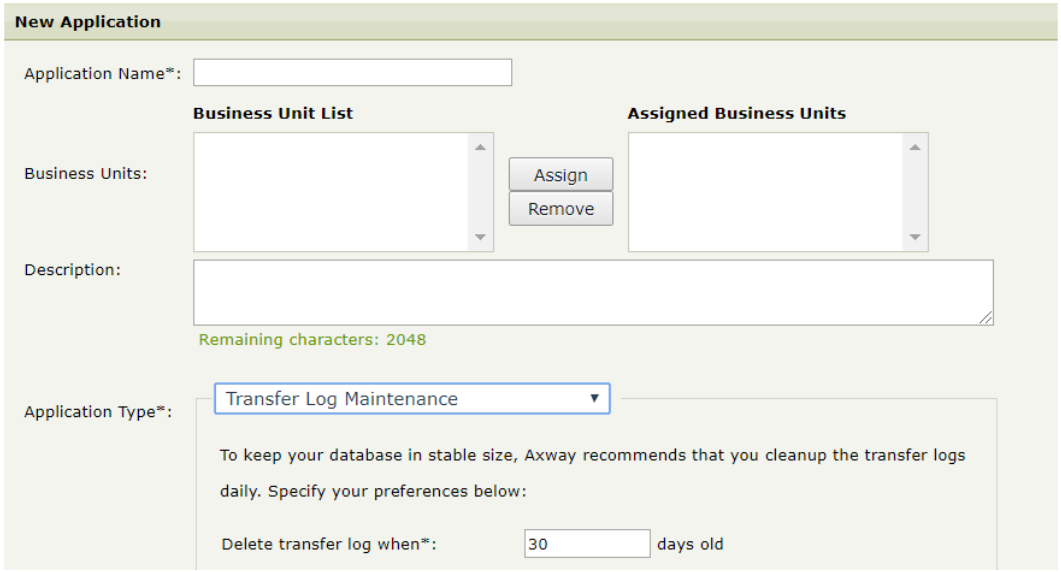
On PostgreSQL

The Transfer Log maintenance application does not create new partitions; instead, they are created by a dedicated service. See *Change external PostgreSQL configuration and manage partitioning on page 104*.

Create a Transfer Log Maintenance application

Use the following procedure:

1. Select **Application** and click **Add New**.
2. Select **Transfer Log Maintenance** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units to the application, see *Business units* on page 746. Assigning business units controls which delegated administrators can manage the application.
5. (Optional) Enter a **Description** for the application.
6. In the **Delete transfer log when _ days old** field, specify the period after which log entries will be deleted from the database. The application computes the age of transfer log entries to midnight of the day it is run. For example, if the value of this field is `1` and the application runs at 4:00 a.m., the application deletes entries created before midnight at the beginning of the

previous day. It does not delete entries created between midnight and 4:00 a.m. on the previous day. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.

7. (For MariaDB only) In the **Delete in-progress transfers that started more than _ days ago** field, specify a period after which in-progress transfers will be deleted. Consider using the same value for completed and in-progress transfers. Otherwise, the result could be a performance degradation or failure to execute the application.
8. (Optional) Configure the export of the server log entries before deletion. The procedure differs depending on the database you use, see *Configure transfer log exports* on page 858
9. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.
10. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

    Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
11. Click **Create Application**.

Configure transfer log exports

You can configure the Transfer Log Maintenance application to export old transfer log records before deleting them. This procedure differs depending on the database.

You can also use the `log_export` command-line utility to export transfer log entries.

Export from an embedded MariaDB database

To export logs from an embedded database:

1. Select **Export data before deletion**.
2. In the **Export folder** field, enter the full path to a directory where exported files will be stored.
3. In the **Delete exported files when data is _ days old** field, specify the number of days that exported files will remain in the export directory before they are deleted. If you leave this field empty or specify 0, SecureTransport will not delete the files.
4. In the **Number of records per file** field, specify the maximum number of records (in thousands) that can exist in an exported file. When this value is exceeded, SecureTransport will start exporting the transfer log entries in a new file.

Export from a Microsoft SQL Server database

Use the Microsoft SQL Server export functionality with caution in regard to your backup strategy.

To export logs from an external MSSQL database:

1. Select **Enable logs export**.
2. In the **Export folder** field, enter the absolute path to a directory where the exported files will be stored
3. Complete the following steps on the Microsoft SQL Server:
   a. Create an export directory.
   b. Create a new filegroup:

      ```
      ALTER DATABASE databaseuser ADD FILEGROUP [ST_
      FILETRACKING_ARCHIVE]
      ```

   c. Create a file:

      ```
      ALTER DATABASE databaseuser
           ADD FILE
           (NAME = N'ST_FILETRACKING_ARCHIVE_databaseuser',
           FILENAME = N'<EXPORT_DIR>\ST_FILETRACKING_ARCHIVE_
      databaseuser.ndf',
           SIZE = 10MB,
           MAXSIZE = 100MB,
           FILEGROWTH = 1MB)
           TO FILEGROUP [ST_FILETRACKING_ARCHIVE]
      GO
      ```

   d. Grant the database user all permissions to the `<EXPORT_DIR>` directory created in step **a**.
   e. Grant **Backup database** and **Backup log** permissions to the database user. The **Backup log** permission is only required if the database is in **Full recovery mode**.

Export from an Oracle database

To export logs from an external Oracle database:

1. Select **Enable logs export**.
2. In the **Export folder** field, specify where the export files will be stored. You must fill in the name of the directory defined in the Oracle database (for example, `ST_DMPDIR`).
3. Complete the following steps on the Oracle Server:
   a. Create the directory where the logs will be exported and make sure that the Oracle user has permissions.
   b. Log in into Oracle as SYSDBA and create the `ST_DMPDIR` directory using the following syntax:

      ```
      CREATE DIRECTORY ST_DMPDIR AS '/YOUR_DIRECTORY_HERE';
      ```

   c. Grant all privileges on the directory to the ST user:

      ```
      GRANT ALL PRIVILEGES ON DIRECTORY ST_DMPDIR TO ST_
      DATABASE_USER
      ```

   d. Grant create table privileges to the ST user:

      ```
      GRANT CREATE TABLE TO ST_DATABASE_USER;
      ```

4. In the **Parallelism Degree** field, specify the number of processors to use during an export operation. You can specify any value from `1` to the number of processors available on the server. You can limit the effect of the export on database performance by limiting the number of used processors.

Export from a PostgreSQL database

With PostgreSQL, SecureTransport uses partitioned tables for storing log data. During installation, five tables are created for storing transfer log data: *subtransmissionstatus*, *transferdata*, *transferdetails*, *transferresubmitdata*, and *transferprotocolcommands*. Each table is partitioned daily.

For exporting records from a PostgreSQL database, SecureTransport uses the *pg_dump* utility that ships with PostgreSQL.

Use the following procedure to configure the application to export old transfer log records before deletion. Note that the database user must be either a superuser or a member of `pg_execute_server_program`.

1. Make the local socket connections trusted or password protected (an encrypted local connection for exports is not supported).

   On the PostgreSQL Server, open the *pg_hba.conf* file for editing and modify/add the following line:

- on Unix-based platforms: `local all all trust` or `local all all password`)
- on Windows OS: `host all all 127.0.0.1/32 trust` or `host all all 127.0.0.1/32 password`

2. On the PostgreSQL Server, create the directory where the logs will be exported and make sure that the PostgreSQL user has permissions.
3. In the Transfer Log Maintenance application settings, select the **Enable logs export** checkbox.
4. In the **Export folder** field, input the absolute path to the folder you created at Step 2.
5. In the **path to pg_dump utility** field, enter the absolute path to the *pg_dump* utility including the file name (`pg_dump.exe` on Windows, `pg_dump` on Unix). Usually it's in the PostgreSQL's `bin` directory on the filesystem of the database server.
6. Save the application settings.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Unlicensed Accounts Maintenance application

This application automatically deletes unlicensed user accounts that have been inactive for a specified period of time (60 days by default). It determines inactive accounts by evaluating three properties (dates): *Last Login*, *Account Created* and *LastPasswordChange*. While the first two are displayed in the *Account Status* (in the *Settings* tab), the last one is stored in the database and is not exposed in the user interface.

Inactive accounts are:

- Users created more than the specified number of days ago that neither have logged in nor changed their password during the specified period.
- Users that have changed their passwords before the specified number of days but have not logged in either during the specified period or at any time.

Prerequisites

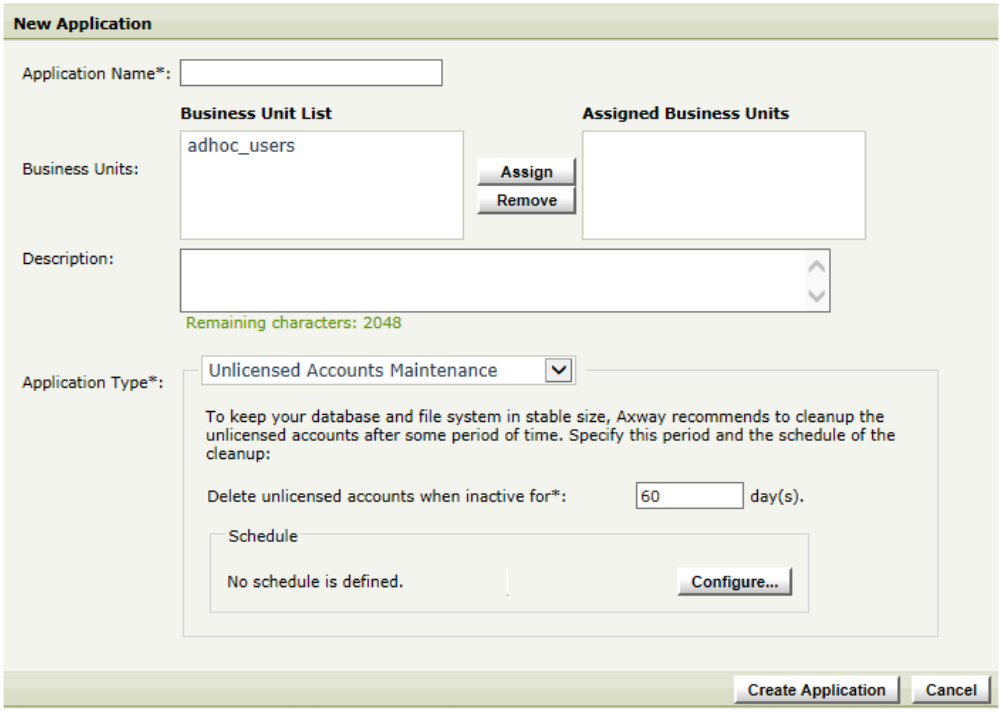
Before you create a Unlicensed Accounts Maintenance application, go to **Setup > TM settings** and make sure that the `UnlicensedAccountMaintApp` rule package is enabled. For more information, see *Manage rule packages* on page 223.

Configuration steps

Use the following procedure to create an Unlicensed Accounts Maintenance application.

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.
2. Select **Unlicensed Accounts Maintenance** from the mandatory **Application Type** list.



3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you have created. For details, see *Business units* on page 746.
5. (Optional) Enter a **Description**.
6. In the **Delete unlicensed accounts when inactive for** field, specify how long in days an unlicensed account must be inactive before it is deleted.
7. (Optional) In the *Schedule* pane, click **Configure** to configure a maintenance schedule.
8. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

   Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
9. Click **Create Application**.

Configure logging

With the default configuration, SecureTransport does not report on account deletions performed by the Unlicensed Account Maintenance application. However, the application activities can be logged in either the Server log, the Audit log, or both.

- To enable audit logging for account deletions performed by the Unlicensed Account Maintenance application, set the `AuditLog.Enabled.TM` server configuration option to `true`.

- To enable reporting to Server log, edit the *<FILEDRIVEHOME>/conf/tm-log4j.xml* file to add the following logger:

```
<Logger name="com.tumbleweed.st.server.appframework.sql.AccountManagerImpl"
level="DEBUG" additivity="false">
<AppenderRef ref="ServerLog"/>
</Logger>
```

Note This logger generates debug-level log messages for each account change, such as disablement or deletion, performed during maintenance.

See parent topic: *Applications* on page 817 and follow shortcuts to other applications you need to create or configure.

Advanced Routing 15

This chapter provides detailed information about the Advanced Routing concepts and procedures. It describes how to create routes, subscribe user accounts to Advanced Routing applications and assign route package templates to user accounts.

In addition to configuration descriptions, you will be able to observe some example basic and advanced use cases.

The final topics of this chapter also contain instructions for Advanced Routing best practices and troubleshooting.

Advanced Routing overview

Advanced Routing (AR) provides a mechanism for multiple routing and transformation iterations over one or more files. As the name suggests, it allows the SecureTransport administrator to create diverse patterns for file transformations and movement (routing) across different partner systems and internal entities. UI-wise, Advanced Routing is a standard SecureTransport *application*: in order to use it, either an account or account template must subscribe to it.

Advanced Routing (AR) is the most complex application, both in terms of configuration and possible use cases. Using it involves concepts and terminology that are entirely AR-specific. In order to take advantage of these concepts, you must first understand the aspects of AR and the great multitude of file transfer scenarios it covers.

What can Advanced Routing do?

AR allows you to configure complex sets of actions (called *route steps*) to apply on one or more files during a transfer process. Route steps are divided into two categories: *transformation* (modification) and *routing* (transferring) of files.

- *Transformation* of files – this includes several predefined file modifications: you configure in what way you want to modify (transform) a file. For example, you can apply PGP encryption/decryption, compression/decompression, character replacement or renaming of filenames, etc.
- *Routing* of files – this is a definition of the AR file transfer: you configure where you want to send your file.

Each transformation or routing is defined as a *step* in a *route*. For example, if you want to 1)encrypt a file, 2)send it to the subscription folder of a selected account and then 3)decrypt it, you must perform one step per each action, three in total. These three steps form the route.

Summary: Route steps are *definitions for actions* you apply to one or more files in a specific order. A set of route steps forms a route.

Advanced Routing allows you to add more routes if you need to. You decide how many steps you'll have in each of your routes. Theoretically, your route package can include as many steps as you want; however, it is not recommended to have too many as it might adversely affect performance.

For best UI experience, it is recommended that you understand the terminology introduced with AR.

Advanced Routing glossary of terms

The following terms and concepts are introduced with Advanced Routing:

- **Route and route steps**

  Advanced Routing provides advanced transformation and routing capabilities to file transfers. Each transformation and movement of files is performed in a *route*. A route is a set of (route) steps which are consecutively executed. The order of step execution is defined in the routes and the eventual processing of steps is triggered by predefined conditions. Only AR file transfers use steps. You can have multiple steps defined in a route. Each step performs either of the two possible actions: file *transformation* or file *routing*.

  Note that both Transformation and route execution can be based on file path/name patterns or other environment variables

- **Step conditions**

  Each step you define has a trigger condition associated with it. You can trigger step execution either by using EL or always (unconditionally).

- **Route package**

  A collection of routes defines a route package. The routes in a package process the file in parallel: this means that no order in route execution is defined. Route package is performed on the Account level and is an instance of the Route Package Template.

- **Route Package Template**

  The Route Package Template is used for adding a route package per account. You can specify the Business units a package template will be available to: only accounts part of the specified Business units will be able to use route packages as defined in the Route Package Template.

Triggering conditions and events

When specific predefined conditions are met, the route kicks off and its steps are performed in the predefined order. These conditions are triggered by specified events. As with routing steps, conditions are wrapped in routes as part of a *Route Package Template* or *Route Package*.

As an administrator, you define the conditions which will trigger transformation and routing processing or both over one or more files. Those conditions or steps are stored in a Route Package or Route Package Template. In order to reuse already defined steps, you must add them to a *Route Package Template*. In contrast, one or more steps which are specific for a particular user can be defined in a *Route Package*.

Advanced Routing processing can be triggered by the following events:

- Successful client upload
- Failed client upload
- Successful client download
- Failed client download
- Successful server pull
- Temporarily failure of a server pull
- Failed server pull - Also applies to failed wildcard and individual file pulls
- Arrival of a positive or negative PeSIT acknowledgment
- Arrival of a PeSIT message

Advanced Routing setup overview

The following list represents the setup stages that a SecureTransport administrator must go through in order to configure an AR instance.

- Create a route package template
- Create AR application
- Create a route package instance from the route package template
- Subscribe account to AR application to trigger the route

As noted above, you must have an account (or an account template) in order to create a subscription to the AR application. You can have multiple AR applications that cover different use cases for different groups of users. Additionally, you can create and AR delegated administrator to administer all AR configurations and file transfers.

As you can see, setup itself introduces some AR-specific concepts. The route package template, the route package instance and the route itself are all different forms of the same concept: a reusable configuration that is easily upgradable when necessary.

Advanced Routing features

Advanced Routing offers the following features:

- Conditioning
    - Transformation and route execution can be based on file path/name patterns or other environment variables
- Transformations
    - PGP Encryption, PGP Decryption, Compress, Decompress, Line Ending, External Script, Encoding Conversion, Characters Replace, Line Padding, Line Truncating, Line Folding, and Rename transformations

    - Multiple transformation execution (for example, Decompress > PGP Decryption > Compress)
    - Renaming
- Routing
    - File routing to transfer sites, accounts (including virtual and LDAP ones), and file system through Publish To Account and Send To Partner
    - Renaming and deleting
    - Overwrite upload folder - optional setting for the new upload folder name which overwrites the one configured in the transfer site settings
    - File routing from transfer sites through Pull From Partner
    - Overwrite download folder - optional setting for the new download folder name which overwrites the one configured in the transfer site settings
    - Overwrite download pattern – optional setting for the new download pattern which overwrites the one configured in the transfer site settings
- Tracking and notifications
    - File Tracking integration
    - Sentinel integration
    - Email notifications on routing and transformation successes, failures, and triggering
- Extensive Expression Language support
- Post routing, post transformation, and post processing actions
- Ability to specify and overwrite transformation and routing steps on an account basis
- Distributed execution of the routes in a Standard Cluster or Enterprise Cluster

The following topics provide detailed info about different aspects of Advanced Routing :

Order of configuration

This topic provides the configuration order for the Advanced Routing feature. It also provides a brief overview of each configuration item.

Business units can be created prior to configuring the Route Package Template. For configuration information, refer to *Business units* on page 746.

Optionally, an Advanced Routing administrator can be created. For configuration information, refer to *Advanced Routing delegated administrator* on page 870 and to *Advanced Routing delegated administrator* on page 870.

The Advanced Routing feature should be configured in the following order:

1. Create user account (or account template) or use an existing one. See .
2. Create Advanced Routing application. See *Create Advanced Routing application* on page 874.
3. Create Route Package Template.
4. Assign Route Package Template to the account from Step 1. See *Assign Route Package Template* on page 887.
5. Subscribe to the Advanced Routing application. See *Subscribe to Advanced Routing application* on page 888.

Create Advanced Routing administrator

In order to setup an administrator dedicated to managing the Advanced Routing application and Route Package Templates it is necessary to create an Advanced Routing administrator with file tracking, accounts, applications, mail templates, and routes privileges. For instructions on creating an Advanced Routing administrator, refer to *Advanced Routing delegated administrator* on page 870.

Create user accounts

In order to subscribe an account to an Advanced Routing application instance, start off by creating a SecureTransport user account or account template (or use an existing one). For information on creating user accounts, refer to *Create a user account* on page 503. For information on creating account templates, refer to *Manage account templates* on page 719

Create Advanced Routing application

Navigate to the **Application** tab and click **Add New**. Specify the preferred **Application Name** for the Advanced Routing application instance. Select *Advanced Routing* from the **Application Type** combo box. For additional information on creating an Advanced Routing application instance, refer to *Create Advanced Routing application* on page 874.

Create Route Package Template

Navigate to the **Routes** tab and click **New Route Package Template**. Specify the preferred **Route Package Template Name** for the Route Package Template, determine assigned business units, enter a route template description, determine execution routes, add transformation and routing steps, and determine notifications. For additional information

Assign Route Package Template

You must subscribe the selected user to the Advanced Routing application prior to assigning the user a Route Package Template. To assign the user a Route Package Template, navigate to **Accounts > User Accounts**, select the desired user account, and then select the **Routes** tab for the selected account. From the *Routes* tab, select the desired Route Package Template from the **Route Package Template** list and then click **Assign Route**. The *Create Route Package* screen is displayed with a link to the selected Route Package Template under the **Created From** label. For more details on assigning a Route Package Template, refer to *Assign Route Package Template* on page 887.

Subscribe to Advanced Routing application

To subscribe a user account to the Advanced Routing application, navigate to **Accounts > User Accounts**, select the desired user account, and then select the **Subscriptions** tab for the selected account. From the **Subscriptions** tab, select *Advanced Routing* from the **Subscribe to** list and click **Subscribe**. For additional details on subscribing to the Advanced Routing application, refer to *Subscribe to Advanced Routing application* on page 888. For details on managing subscriptions, refer to *Manage subscriptions* on page 664.

Configuration

This topic provides the step-by-step instructions for creating delegated administrator accounts, user accounts, applications, and templates for Advanced Routing. It also includes step-by-step instructions for adding an Advanced Routing application, subscribing to the Advanced Routing application, and assigning a Route Package Template.

The following topics provide how-to instructions for configuring Advanced Routing:

- *Advanced Routing delegated administrator* on page 870 - Provides how-to instructions for creating an Advanced Routing delegated administrator.
- *Create user accounts* on page 873 - Provides how-to instructions for creating user accounts.
- *Create Advanced Routing application* on page 874 - Provides how-to instructions for creating Advanced Routing application.
- *Manage Route Package Templates* on page 875 - Provides how-to instructions for managing Route Package Templates.
- *Manage Routes* on page 881 - Provides how-to instructions for managing Routes.
- *Assign Route Package Template* on page 887 - Provides how-to instructions for assigning a Route Package Template.
- *Subscribe to Advanced Routing application* on page 888 - Provides how-to instructions for subscribing to the Advanced Routing application.

Advanced Routing delegated administrator

The configuration of a delegated administrator for Advanced Routing requires creating an Advanced Routing administrator role and creating an Advanced Routing administrator with the specified administrator settings. The following topics provide the configuration details for creating the role of Advanced Routing administrator and assigning the role to the Advanced Routing administrator.

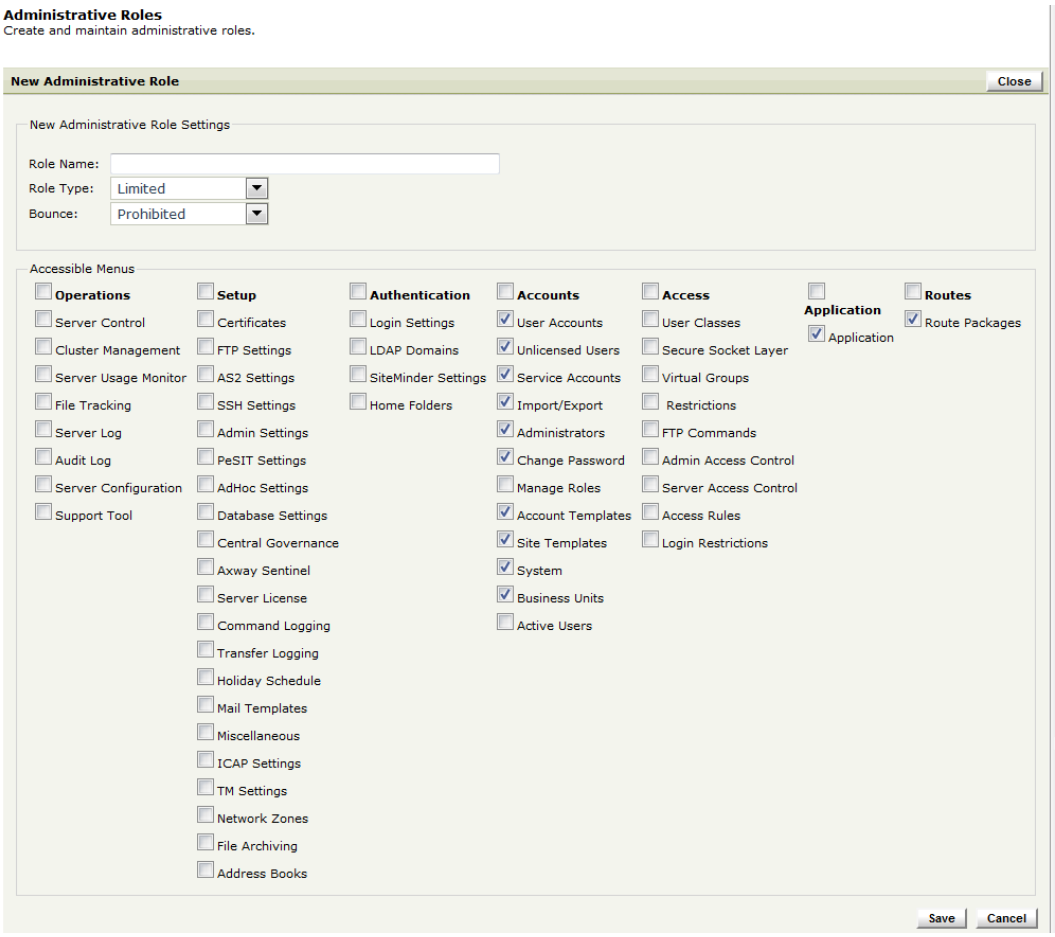
The following topics provide how-to instructions for creating an Advanced Routing administrator role and creating an Advanced Routing administrator:

- *Create Advanced Routing administrator role* on page 870
- *Create Advanced Routing administrator* on page 871

Create Advanced Routing administrator role

For details on creating administrative roles, refer to *Administrative roles* on page 711. The Advanced Routing administrator role should be created with the same settings as a delegated administrator plus the selection of *Route Packages* as shown in the following table and figure.

Function	Selections
Role Name:	Advanced Routing Administrator
Role Type:	Limited
Bounce:	Prohibited
Accessible Menus	
Operations	No selections
Setup	Mail Templates
LDAP	No selections
Accounts	User Accounts, Unlicensed Users, Service Accounts, Import/Export, Administrators, Change Password, Account Templates, Site Templates, System. Business Units
Access	No selections
Application	Application
Routes	Route Packages



Create Advanced Routing administrator

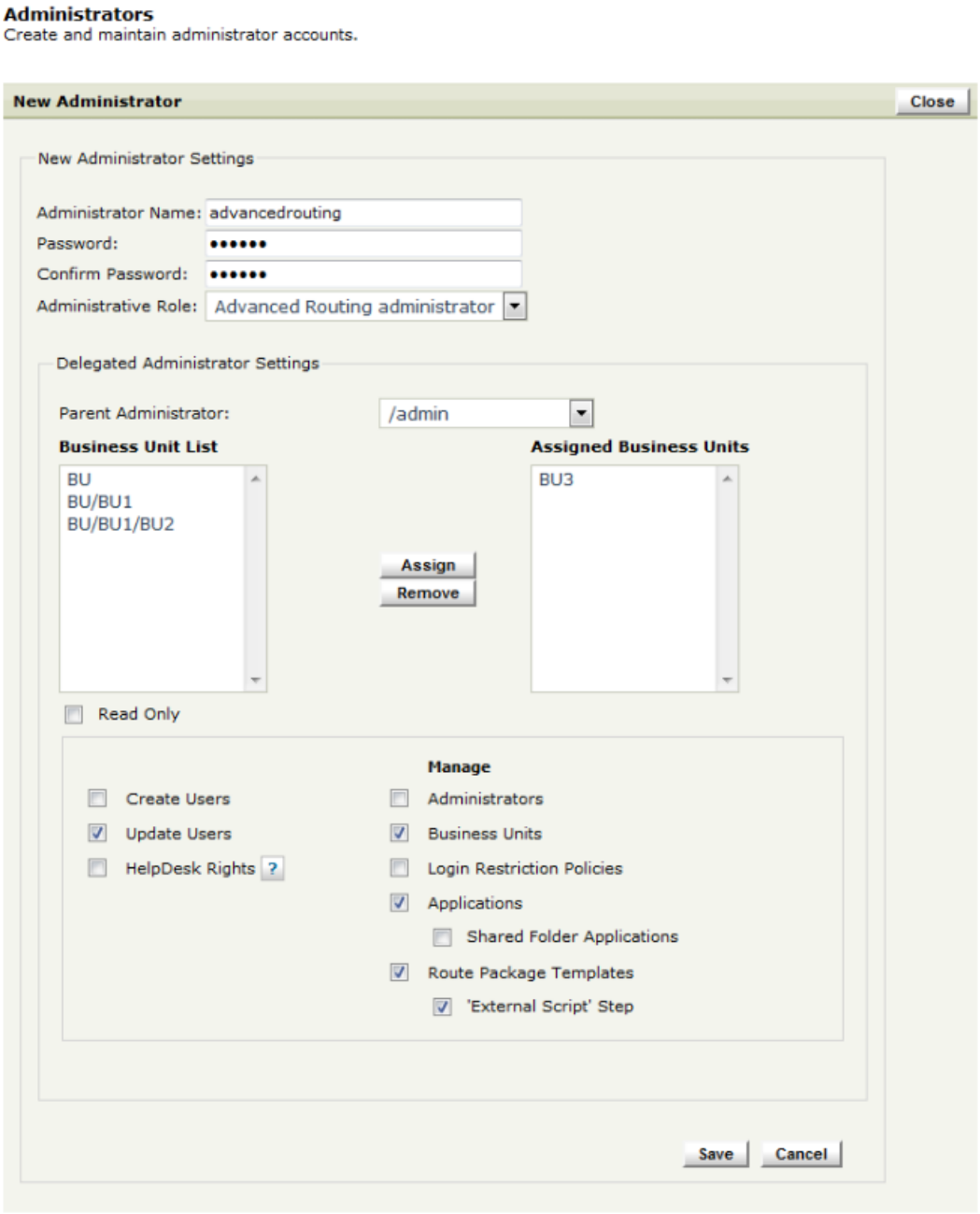
For details on creating administrators, refer to *Manage administrator accounts* on page 700. An Advanced Routing administrator should be created with settings as shown in the following table and figure.

Note At least one business unit must be assigned to the Advanced Routing administrator.

Function	Selections
Administrator Name:	User determined
Password:	User determined
Confirm Password:	Must match password entry
Administrative Role:	Advanced Routing Administrator

Function	Selections
Parent Administrator:	/admin
Assigned Business Units	User determined - At least one Business Unit should be assigned
Advanced Routing Administrator Selections	Update Users, Business Units, Applications, Route Package Templates, 'External Script' Step

Note Manage Route Package Templates and Manage 'External Script' Step must be selected.

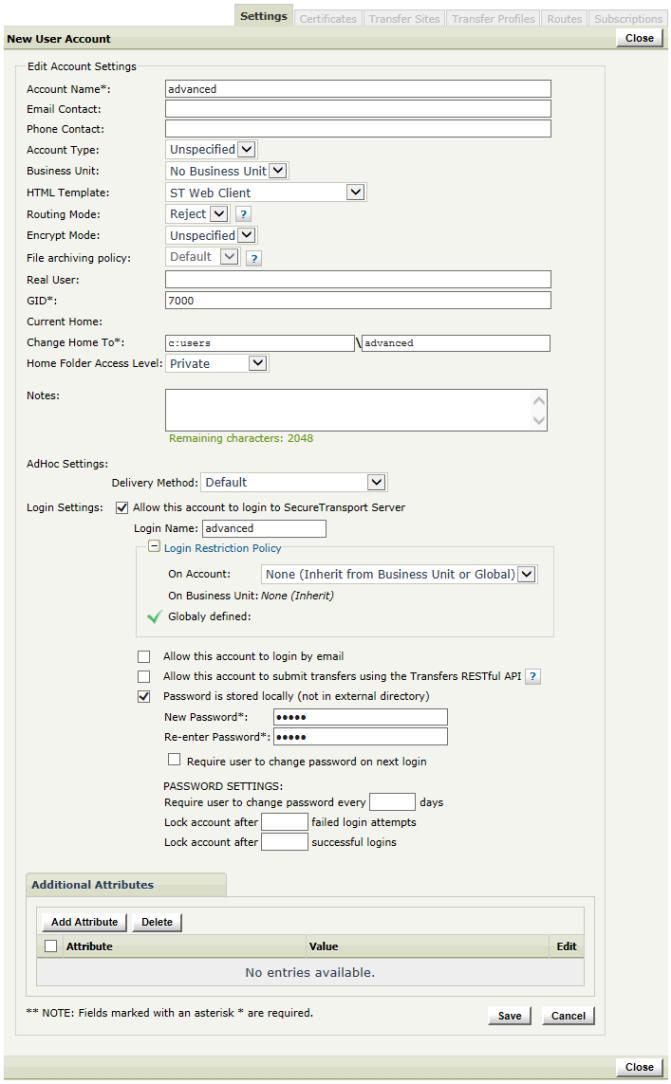


Create user accounts

For details on creating user accounts, refer to *User accounts* on page 501. Accounts which have advanced routes can be created using the instructions for adding user accounts.

Note The *Home Folder Access Level* property (which specific to Advanced Routing functionality) defines whether or not files can be routed to the account's home folder.

The following figure shows an example of an account which has advanced routes assigned.

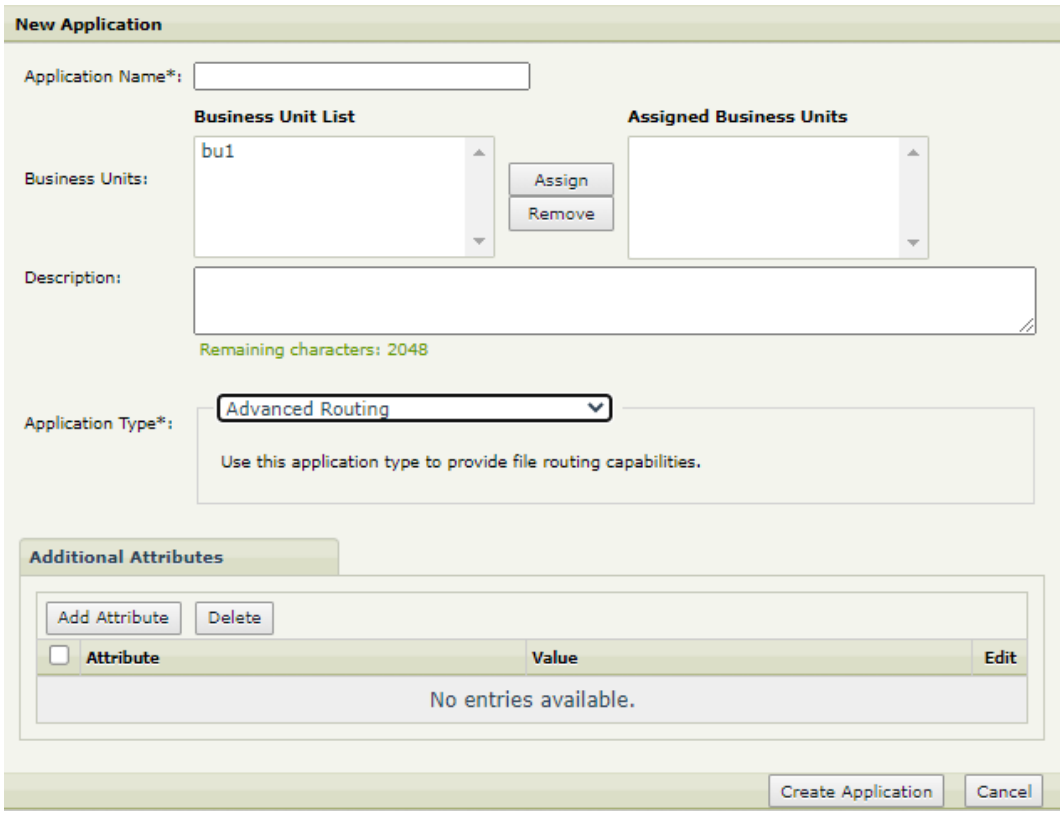


Create Advanced Routing application

1. Select **Application** and click **Add New**.

   The *New Application* page is displayed.

2. Select Advanced Routing from the mandatory **Application Type** list.

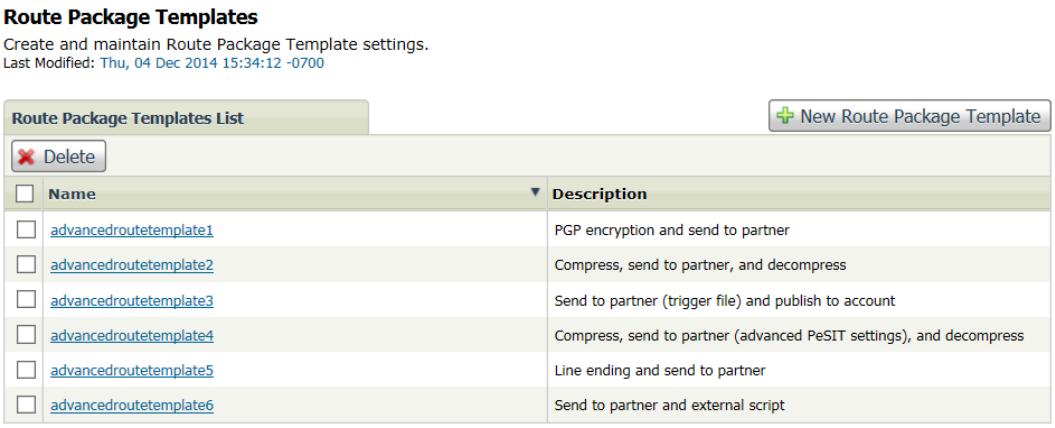


3. Enter a unique **Application Name** of up to 80 characters. You cannot enter forward slashes (/) or spaces-only values in this field. For more information, see *Spaces in required fields on page 514*.
4. (Optional) Use the **Assign** and **Remove** buttons to assign business units for the application. The **Business Unit List** contains the names of business units you create. For details, see *Business units on page 746*.
5. (Optional) Enter an application **Description**.
6. (Optional) Set **Additional attributes**: you can add custom attributes as *attribute:value* pairs.

   Click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes. For details, see *Additional attributes* on page 759.
7. Click **Create Application**.

Manage Route Package Templates

Use the *Route Package Templates* page to create, edit, or delete a Route package template. To access it, select **Routes > Route Packages**. Master administrators and administrators with the Manage Route Package Templates privilege can access the page.



The following topics provide how-to instructions for managing Route Package Templates:

- *Add Route Package Template* on page 876
- *Edit Route Package Template* on page 879
- *Enable Route* on page 880
- *Disable Route* on page 880
- *Reorder Routes* on page 880
- *Delete Route* on page 880
- *Delete Route Package Template* on page 881

Add Route Package Template

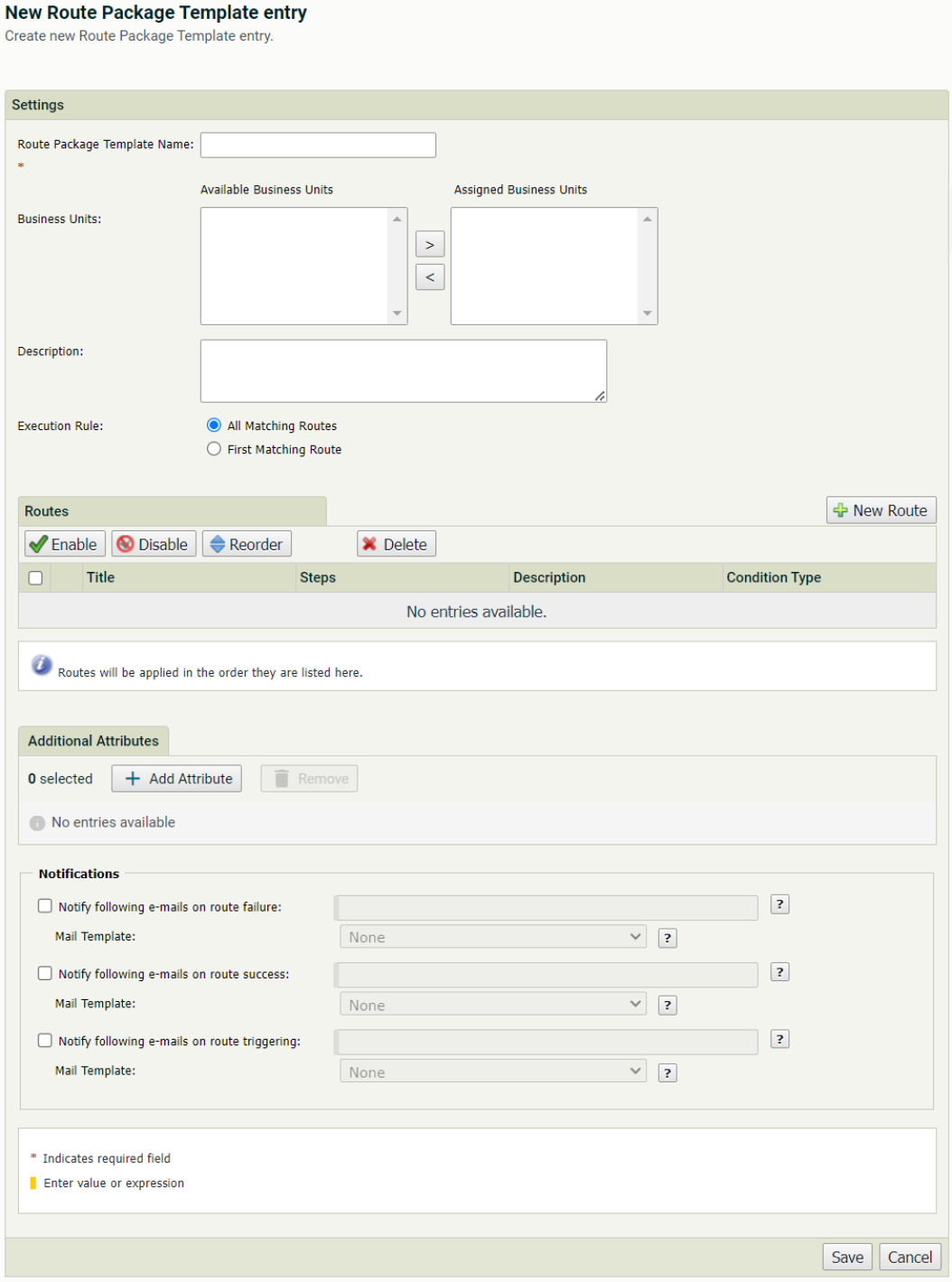
Use the following procedure to add a Route Package Template.

1. Select **Routes >Route Packages**.

   The *Route Package Template* page is displayed.

2. Click **New Route Package Template**.

   The *New Route Package Template Entry* page is displayed.



3. Enter a unique **Route Package Template Name**. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
4. (Optional) Use the **Left** and **Right** buttons to assign business units to the Route Package Template. The **Business Unit List** contains the names of business units you create. For details, *Business units* on page 746.
5. (Optional) Enter a **Description**.

6. Determine the **Execution Rule**. Select either **All Matching Routes** (default) or **First Matching Route**.

   When **All Matching Routes** is selected, all matching routes are executed. When **First Matching Route** is selected, only the first matching route is executed.

7. Click **New Route**.

   The *New Route Entry* page is displayed. For route entry configuration information, see *Manage Routes* on page 881.

8. (Optional) Add **Additional attributes** to your route package templates: you can use the group of fields to add custom attributes as *attribute:value* pairs. Expression Language is not supported. For details, see *Additional attributes* on page 759.

   To add a new attribute, click **Add Attribute**, fill in the fields and click the **Save** (✓) icon. To remove an attribute, select the corresponding checkbox and click **Remove**. You can also edit existing attributes.

9. Determine email **Notifications**.
   To enable SecureTransport to send notifications about various events, you need to first configure an SMTP mail server in **Setup > Miscellaneous > SMTP Configuration**. For more information, see *Set up email notifications via SMTP on page 202*. SecureTransport uses email templates to create the email notification messages. Therefore, the administrator must also have access to the Mail Templates page; otherwise, the *Notification* pane is disabled. The access to Mail Templates is configurable through the Administrative role settings. See *Manage administrator accounts on page 700*.

   - Specify notifications for a route failure

     Select **Notify following e-mails on route failure** to specify where notifications about route failures are sent. The field can contain either one or more comma-separated email addresses or an expression. Then, choose the **Mail Template** you want to use for the messages for a route failure.

   - Specify notifications for a route success

     Select **Notify following e-mails on route success** to specify where notifications on route success to be sent. The field can contain either one or more comma-separated email addresses or an expression. Then, choose the **Mail Template** you want to use for the messages for a route success.

   - Specify notifications on route triggering

     Select the **Notify following e-mails on route trigger** checkbox and specify where the notifications to be sent. The filed can contain either an email addresses, a mail relay, or an SMTP port. Then, choose the **Mail Template** you want to use for route triggering notifications.

10. Click **Save**.

Note You can add, enable, disable, reorder, and delete routes as part of editing a Route Package Template.

Edit Route Package Template

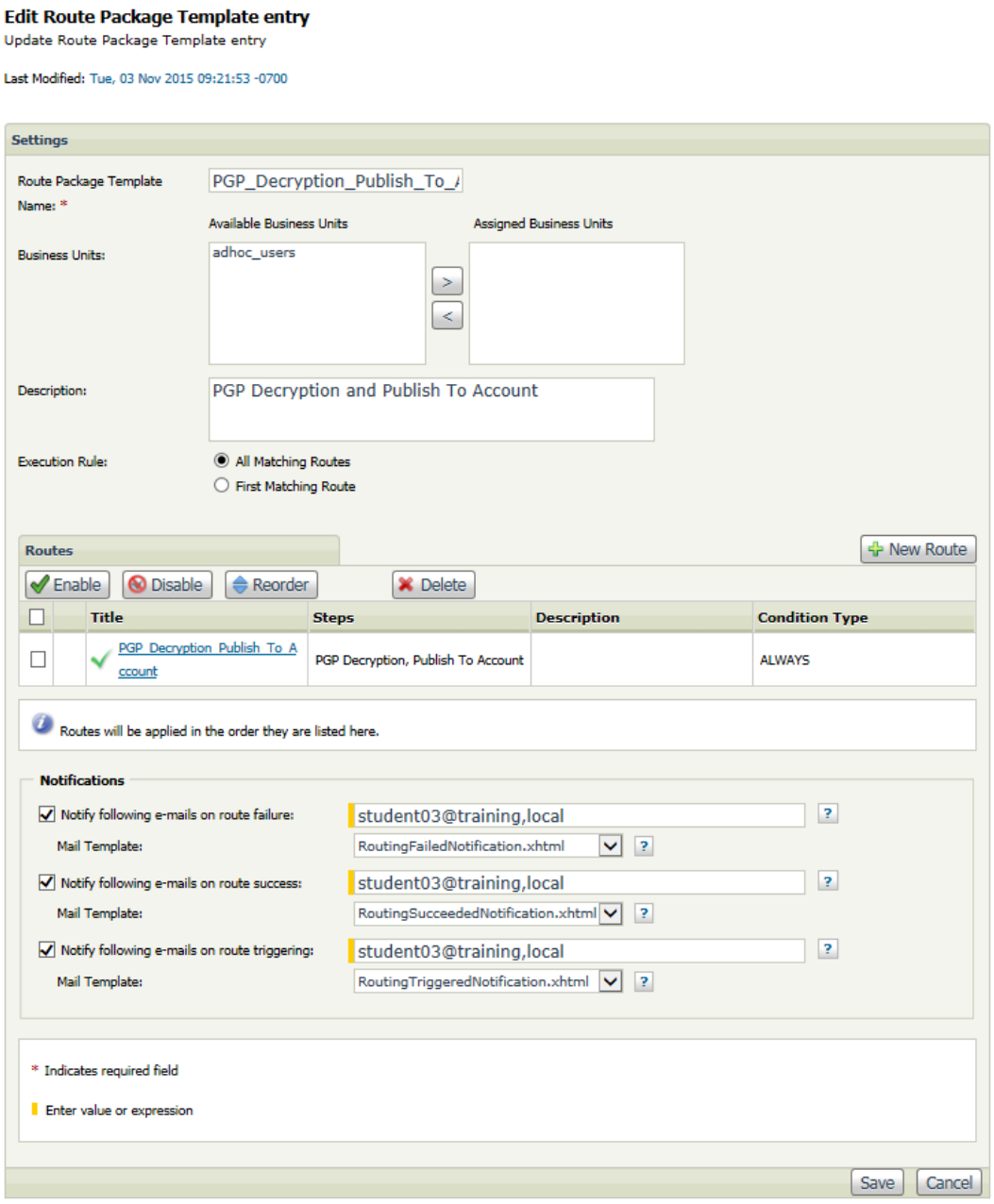
Use the following procedure to edit a Route Package Template.

1. Select **Routes >Route Packages**.

   The *Route Package Template* page is displayed.

2. Click on the name of the Route Package Template to edit in the *Route Package Templates List*.

   The *Edit Route Package Template Entry* page is displayed.



3. Edit the information displayed. To edit a route, click on the name of the Route and see *Manage*

Routes on page 881.

4. Click **Save** to apply the changes.

Note You can *Manage Routes* on page 881, *Enable Route* on page 880, *Disable Route* on page 880, *Reorder Routes* on page 880, and *Delete Route* on page 880 as part of editing a Route Package Template.

Enable Route

Use the following procedure to enable a Route.

1. Select the Route to enable from the *Routes* list.
2. Click **Enable**.

   The selected Route is enabled and a Enabled icon ( ) is displayed next to the selected Route name.

Disable Route

Use the following procedure to disable a Route.

1. Select the Route to disable from the *Routes* list.
2. Click **Disable**.

   The selected Route is disabled and a Disabled icon ( ) is displayed next to the selected Route name.

Reorder Routes

Use the following procedure to reorder the Routes

1. Click **Reorder**.

   A Reorder tool ( ) appears for each Route in the *Routes* list.
2. Use the Reorder tool ( ) to move the Routes in the *Routes* list into the desired order.
3. Click **Save Order**.

Delete Route

Use the following procedure to delete a Route.

1. Select the Route to delete from the *Routes* list.
2. Click **Delete**.
3. When prompted, confirm that you would like to delete the selected Route.

Delete Route Package Template

Use the following procedure to delete a Route Package Template.

1. Select **Routes >Route Packages**.

   The *Route Package Template* page is displayed.
2. Select the checkbox for the Route Package Template to delete from the *Route Package Templates List*.
3. Click Delete.
4. When prompted, confirm that you would like to delete the selected Route Package Template.

Manage Routes

If you have access to the *Route Package Templates* page, you can create, edit, and delete Routes as part of managing Route Package Templates. You can also add, edit, enable, disable, reorder, and delete Route steps.

The following topics provide how-to instructions for managing Routes:

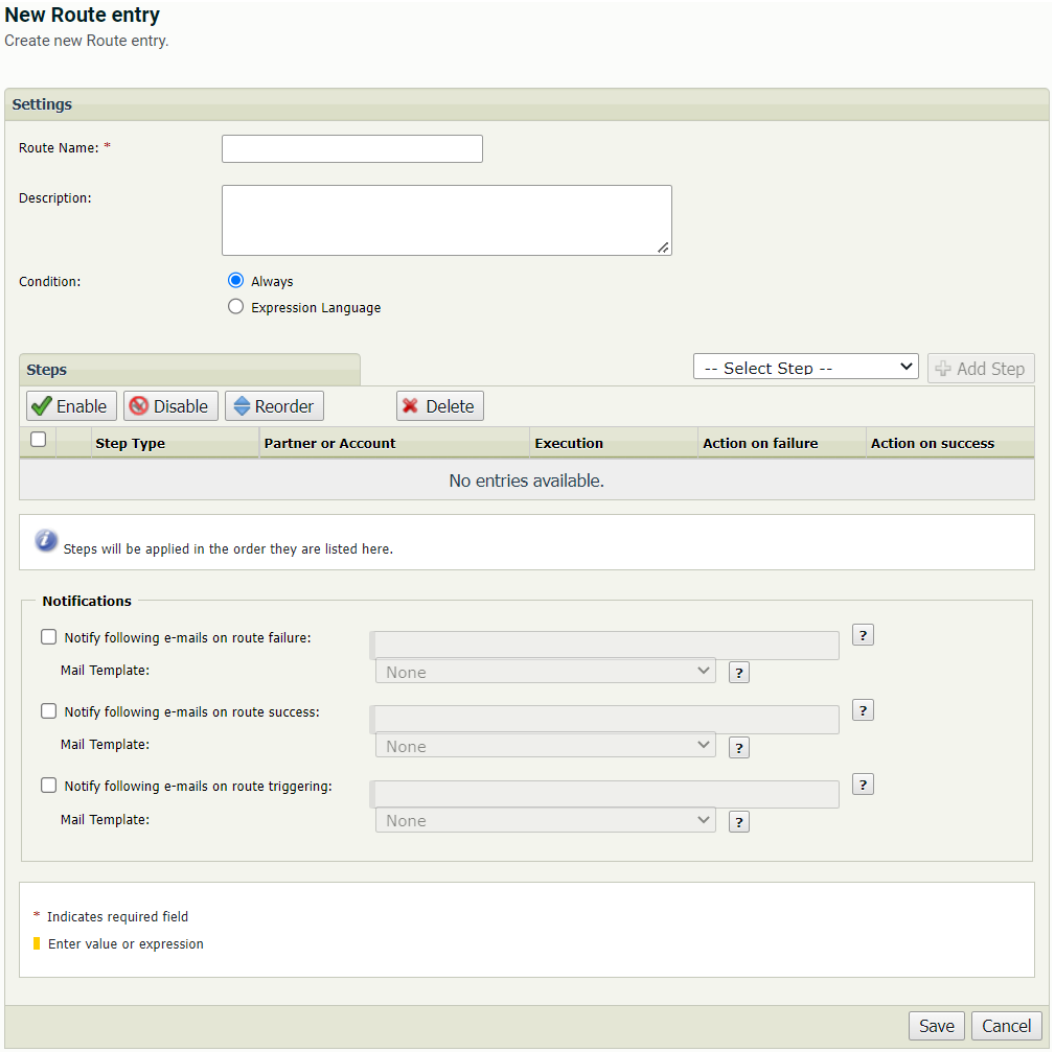
- *New Route* on page 881
- *Edit Route* on page 884
- *Enable Step* on page 885
- *Disable Step* on page 886
- *Reorder Steps* on page 886
- *Delete Step* on page 886
- *Delete Route* on page 886

New Route

Use the following procedure to create a Route.

1. From the *Route Package Template Entry* page, click **New Route**.

   The *New Route Entry* page is displayed.



2. Enter a unique **Route Name**. You cannot enter spaces-only values in this field. For more information, see *Spaces in required fields* on page 514.
3. (Optional) Enter a **Description**.
4. Specify when the route should be executed: Either select **Always** or set a condition using **Expression Language**. For example, to start a route upon receiving a PeSIT acknowledgment, define PeSIT acknowledgment-based criteria in the **Expression Language** text field.
5. Select steps (**Transformation** or **Routing**) from the *Select Step* menu and click **Add Step**. Refer to the following tables for **Transformation** or **Routing** configuration information.

Transformation	Configuration Reference
PGP Encryption	*PGP Encryption* on page 896
PGP Decryption	*PGP Decryption* on page 901

Transformation	Configuration Reference
Compress	*Compress* on page 904
Decompress	*Decompress* on page 908
Line Ending	*Line Ending* on page 912
External Script	*External Script* on page 916
Encoding Conversion	*Encoding Conversion* on page 921
Characters Replace	*Characters Replace* on page 924
Line Padding	*Line Padding* on page 929
Line Folding	*Line Folding* on page 937
Line Truncating	*Line Truncating* on page 933

Routing	Configuration Reference
Pull From Partner	*Pull From Partner* on page 956
Publish To Account	*Publish To Account* on page 943
Send To Partner	*Send To Partner* on page 948

6. Determine email **Notifications**. In order to add email notifications the administrator must have access to Mail Templates, otherwise this selection is disabled. Mail Templates access is configurable through the Administrative role settings. For additional administrative role configuration information, refer to *Manage administrator accounts* on page 700.
    a. Select **Notify following e-mails on route failure**. A route failure occurs when the transformation or sending of the payload file fails, or in the case of Pull From Partner, when SecureTransport fails to submit the pull request.

    You need to have configured SMTP settings on the **Administration Tool > Setup > Miscellaneous > SMTP Configuration** page (notify email, mail relay and SMTP port). The *Notify following e-mails on route failure* field supports EL and you can enter:

    - An email address
    - An expression, for example `ldap.attributes.Mail, ${account.name}, ${account.email}`.
    - A list of email addresses (delimiters depend on the SMTP server)

    For additional email configuration information, see *Set up email notifications via SMTP* on page 202.

b. Select the **Mail Template** from the menu to used for route failure notifications. For email template configuration information, see *Mail templates* on page 195.

c. Select **Notify following e-mails on route success** to be notified on route success and enter a notification email address, mail relay, or SMTP port in the field.

d. Select the **Mail Template** from the menu to used for route failure notifications.

e. Select **Notify following e-mails on route trigger** to be notified on route trigger and enter a notification email address, mail relay, or SMTP port in the field.

f. Select the **Mail Template** from the menu to used for route trigger notifications.

7. Click **Save**.

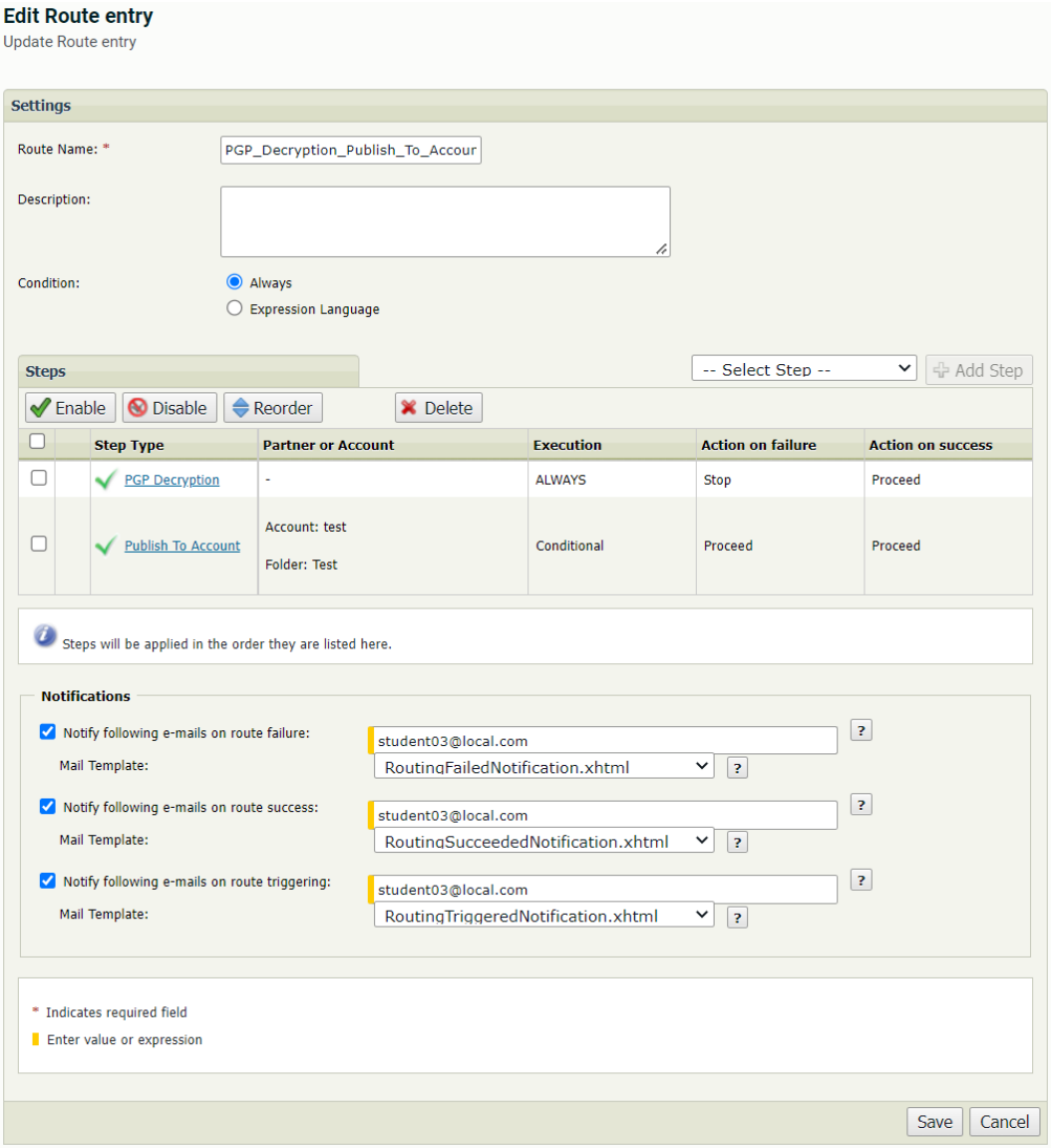
Note You can Edit, *Enable Step* on page 885, *Disable Step* on page 886, *Reorder Steps* on page 886, and *Delete Step* on page 886 as part of adding a Route.

Edit Route

Use the following procedure to edit a Route.

1. From the *Route Package Template Entry* page, click on the name of the Route to edit in the *Routes* list.

   The *Edit Route Entry* page is displayed.



2. Edit the information displayed. To edit a Route Step, click on the name of the Route Step and refer to *Transformations* on page 895 to edit Transformations and to *Routing steps* on page 943 to edit Route Steps.
3. Click **Save** to apply the changes.

Note You can Edit, *Enable Step* on page 885, *Disable Step* on page 886, *Reorder Steps* on page 886, and *Delete Step* on page 886 as part of adding a Route.

Enable Step

Use the following procedure to enable a Route step.

1. Select the Step to enable from the *Steps* list.
2. Click **Enable**.

The selected Route Step is enabled and an Enabled icon ( ) is displayed next to the selected Step name.

Disable Step

Use the following procedure to disable a Route step.

1. Select the Step to disable from the *Steps* list.
2. Click **Disable**.

   The selected Step is disabled and a Disabled icon ( ) is displayed next to the selected Step name.

Reorder Steps

Use the following procedure to reorder Route steps.

1. Click **Reorder**.

   A Reorder tool ( ) appears for each Step in the *Steps* list.
2. Use the Reorder tool ( ) to move the Steps in the *Steps* list into the desired order.
3. Click **Save Order**.

Delete Step

Use the following procedure to delete a Route step.

1. Select the Step to delete from the *Steps* list.
2. Click **Delete**.
3. When prompted, confirm that you would like to delete the selected Step.

Delete Route

Use the following procedure to delete a Route.

1. From the *Route Package Template Entry* page, select the checkbox for the Route to delete in the *Routes* list.
2. Click **Delete**.
3. When prompted, confirm that you would like to delete the selected Route.

Assign Route Package Template

To assign a Route Package Template to a user account or account template, you must create at least one Route Package Template prior to assigning a route to a user account or account template. For configuration details on managing and creating Route Package Templates, refer to *Manage Route Package Templates* on page 875.

1. Select **Accounts > User Accounts**.
2. On the *User Accounts* page, click the name of the account that you want to assign a route.
3. On the *User Account Settings* page, click the **Routes** tab.
4. From the **Route Package Template** drop-down, select the desired Route Package Template.
5. Click **Assign Route**.

   The *Create Route Package* page is displayed. You can navigate to the *Edit Route Package Template* page for the selected Route Package Template by clicking the **Created From** link.
6. In the **Route Name** field, type a name for the route. This name must no more than 294 characters and must not contain any of the following characters: `* < > ? " / \ | :`
7. (Optional) Enter a **Description**.
8. Under *Inherited Settings*, you will see all simple routes that are inherited from the assigned Route Package Template. Those routes will be applied in the order they are listed. You cannot reorder or delete inherited routes, but you can enable or disable any of them.

   Caution:

   - Any inherited simple routes from the global Route Package Template are executed first.
   - The inherited execution rule cannot be changed.
   - Routes, that are disabled in the inherited Route Package Template, cannot be enabled from the *Inherited Settings* pane.
9. In the *Specific Settings* section, you configure local simple routes. Local routes are executed after the inherited routes, listed in the *Inherited Settings* section.
   *Specific Settings*offers controls to add, edit, enable, disable, reorder and delete local routes. You can also select an execution rule between **All Matching Routes** and **First Matching Route**.

Setting	Description
All Matching Routes	When selected, all matching simple routes are executed.
First Matching Route	When selected, only the first simple route to match gets executed.

For more information on managing simple routes, see *Manage Routes* on page 881.

For more information on managing Route Package Templates, see *Manage Route Package Templates* on page 875.

10. In the *Notifications* pane, you configure email notifications for route triggering, success, and/or failure.

    Prerequisites:

    - Have an SMTP mail server configured in **Setup > Miscellaneous > SMTP Configuration**. See *Set up email notifications via SMTP on page 202*.
    - Have the needed email templates uploaded on the *Mail Template Repository* page (**Setup > Mail Templates**) and have access permissions. The latter is configurable through the Administrative role settings. See *Manage administrator accounts on page 700*.

    To enable email notifications:

    a. Select the checkbox for the activity you want to be notified of: route failure, success, and/or triggering.

    b. In the corresponding text field, enter an email address, a comma-separated list of email addresses, or an expression. If **Notify following e-mails on route trigger** is selected, the filed can also contain a mail relay or an SMTP port.

    c. From the **Mail Template** drop-down, select a template for the notification message.

11. Click **Save**.

Subscribe to Advanced Routing application

An Advanced Routing subscription uses a folder beneath the home directory of an account (subscription folder) and triggers a selected route when a file arrives in it. Based on the route execution status - success, temporary or permanent failure, SecureTransport can either retry the route or trigger a file operation.

Before you subscribe a user account or an account template to an Advanced Routing application, you need the following objects:

Prerequisites

- An Advanced Routing application. See *Create Advanced Routing application* on page 874.
- At least one Route Package Template to instantiate to create local routes on the user account.
- A user account (or an account template) with a transfer site and at least one local route or route package assigned. See *Assign Route Package Template* on page 887.

Workflow

1. Subscribe the account to an AR application.
2. Specify the subscription folder and general settings for the files arriving in it.
3. (Optional) Configure pull settings.
4. Select a route for the subscription to use.
5. Configure route triggers and post-transmission settings on success.
6. Configure route execution or a file operation on temporary and/or permanent failure.
7. Specify an action to take on files downloaded from the subscription folder.

8. Specify an action to take on files that have triggered a route.
9. Click **Add**.

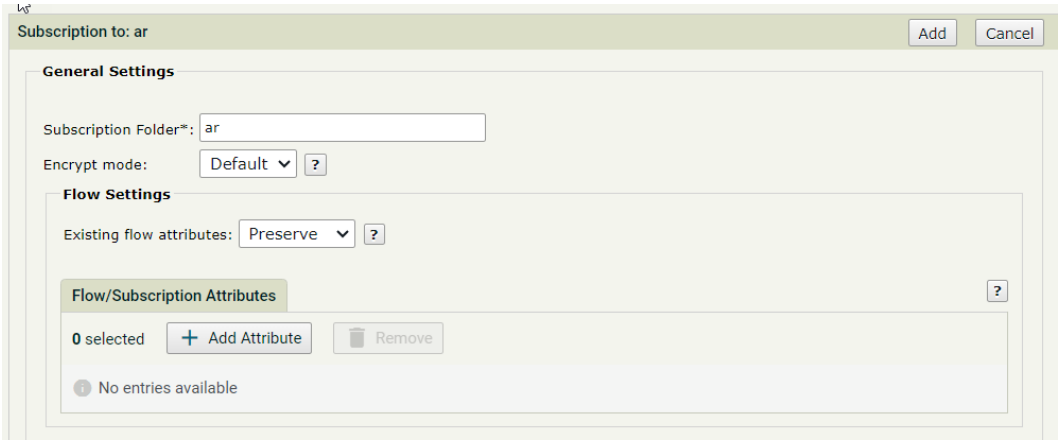
Subscribe the user account to an AR application

1. Select **Accounts > User Accounts**.
2. Click on the name of the account that you want to subscribe to the Advanced Routing application.
3. On the *User Account Settings* page, click the **Subscriptions** tab for the selected account.
4. Click **Subscribe**.

   The subscription configuration page is displayed.

Configure the subscription folder and general settings

In the *General Settings* pane, you set the subscription folder and manage encryption and attribute settings for the files that arrive in this folder.



1. In the **Subscription Folder** field, type the full path to the folder that the subscription will use. This folder must be located inside the account's home folder. Its name can be up to 254 characters long and must not contain any of the following characters: `* < > ? " \ | :`
2. From the **Encrypt mode** drop-down, select an encryption mode for the files in the subscription folder. The encryption of files transferred via Advanced Routing depends on the encryption setting of the target subscription folder repository. For configuration instructions, see *Repository encryption* on page 46.
   There are three options:

Option	Description
Default	The subscription folder inherits the encryption mode from the account or the global settings.

Option	Description
Enable	Encrypts files as they arrive in the subscription folder.
Disable	Disables encryption for files being uploaded to this folder.

3. The *Flow Settings* section enables the creation and modification of the flow and subscription attributes for files arriving in the subscription folder.
   Flow attributes can be used for expression evaluation in Advanced Routing only when the application operates with files. They can be accessed using the expression:
   `${flow.attributes['userVars.ATTRIBUTE_NAME']}`
   Subscription attributes are bound to the subscription, therefore, they can be used for expression evaluation in all Advanced Routing fields. They can be accessed using the expression:
   `${subscription.attributes['userVars.ATTRIBUTE_NAME']}`

   To set a new flow or subscription attribute:

   a. Click **+ Add Attribute** and define the key/value pair. Check the tooltip for details and usage examples.

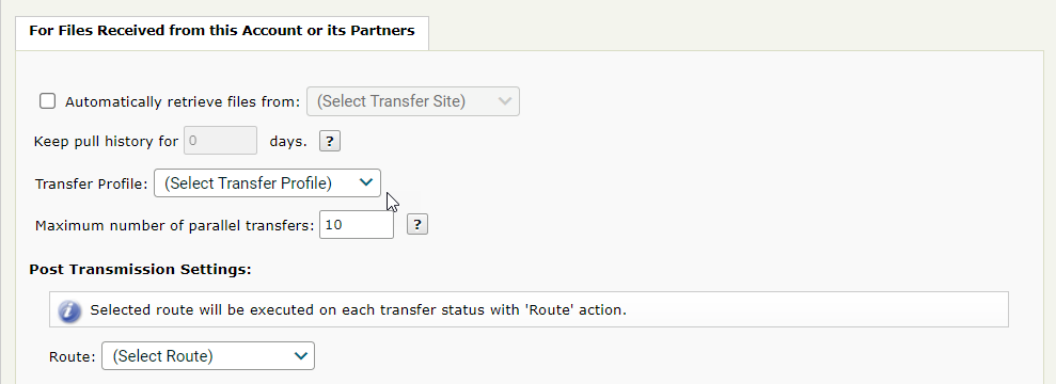
   b. From the **Existing flow attributes** drop-down, specify what will happen with the existing attributes of the incoming files. There are three options:

Options	Description
Preserve	Preserves the existing attributes of the incoming files. The attributes in the *Flow/Subscription Attributes* table get applied only to newly received files that do not have associated flow attributes.
Overwrite	Overwrites any existing attributes of the incoming files with the ones defined in the *Flow/Subscription Attributes* table.
Append	Appends the newly added attribute in case it is not set in the incoming file. If it is set, the original value is preserved.

   c. Click ✓ to save the attribute.

Set the subscription pull settings

A subscription can be configured to automatically pull files from a transfer site to the subscription folder at a scheduled time.



1. Under **For Files Received from this Account or its Partners**, enable **Automatically Retrieve Files From** and select the transfer site to be pulled from the drop-down.

   Specifics:

   - If you select a PeSIT transfer site, you can also set a **Transfer Profile**. If no transfer profile is selected, the default one will be used. For more information, see *Transfer sites* on page 540.
   - With SSH transfer sites you have the option to pull only files that have been added or changed since the last retrieval. See instructions.

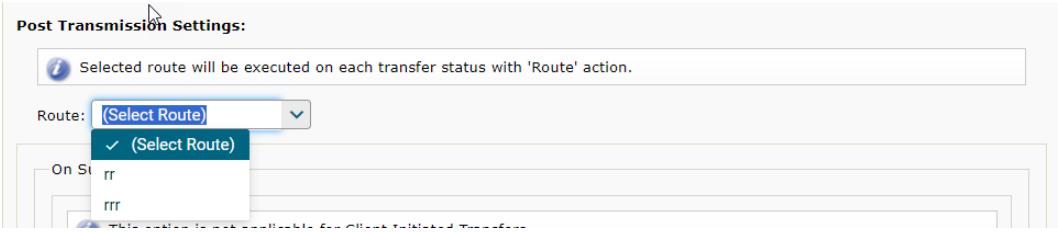
2. To schedule automatic SIT pulls, in the **Schedule** pane, click **Configure**. You can set up a one-time pull event or a recurring schedule. For instructions, see *Scheduled downloads and tasks* on page 674 To initiate a pull immediately, click the **Retrieve files now**.
3. (Optional) Set the **Maximum number of parallel transfers**.
   This limit is applied cluster wide.

   Note Due to limitations in Standard Cluster communication mode, the parallel pulls limit can be exceeded when there are several connections. If you want to force the limit, then the `force.standard.cluster.sit.pulls.sync=true` system property should be added to the `start_tm_console`. Adding the property to the `start_tm_console` has a performance penalty due to increased cluster communication.

4. Use the **Execute route when the remote server returns no files** checkboxes to specify whether or not to trigger the route when the pull returns no file. There are three of these checkboxes, one for each pull status: success, temporary failure, and permanent failure.

Select a route

From the **Route** drop-down, select a route for the subscription to use.



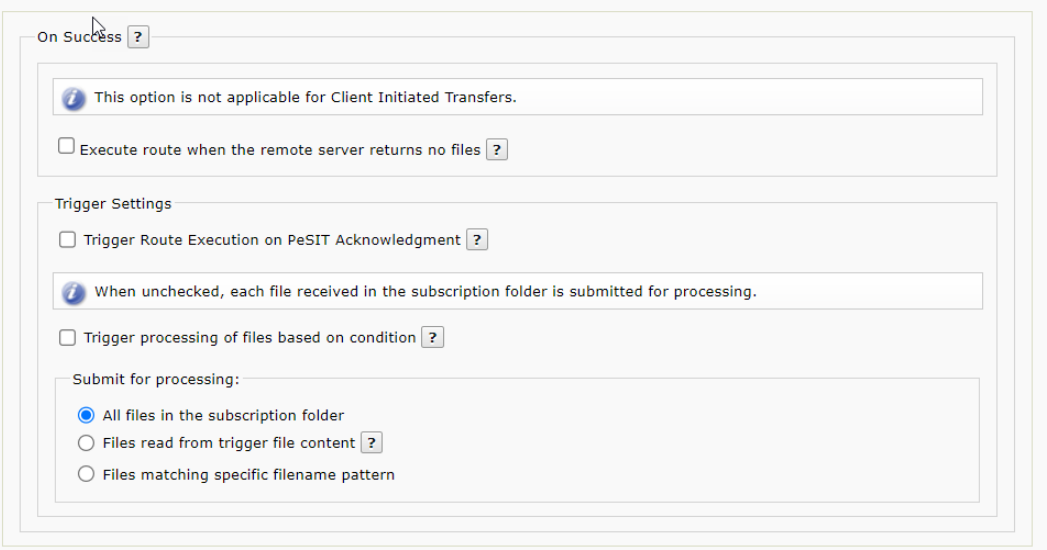
It could be either a composite or simple route. A subscription can trigger only one route. If it is a composite route, you can use the Route Package settings to add conditional processing to a flow. If it is a simple route, you can add conditions to your step execution.

Post-transmission settings

In *Post Transmission Settings* you specify an action to take, depending on the outcome of the route execution: success, temporary failure or permanent failure.

Post-transmission settings on success

By default, any file landing in the subscription folder initiates the specified route. You can change that by using one or a combination of the following settings:



Execute route when the remote server returns no files

Triggers the route without a file, therefore the route should contain a Pull From Partner or an External Script step. The rest of the AR steps work with files, and in this case, will result in a warning message.

Trigger processing of files based on condition

When this checkbox is selected, the subscription will not trigger the route until the condition, specified in the **Trigger condition** field, is met. Additionally, you can specify which files to be submitted for processing by selecting a **Submit for processing** option:

- All files in the subscription folder
- Files read from trigger file content
  When selected, SecureTransport waits until a trigger file that match the **Trigger condition** arrives and then submits for routing only the files specified in the trigger file's contents.
  The trigger file must follow the format described in the tooltip next to the option. To avoid problems, make sure it does not contain duplicate entries and specify what should happen with the AR processing if the trigger file contains a reference to a missing file. Select one of the following options:

Option	Description
Fail	The AR processing will fail.
Continue	The AR processing continues with the existing files.
Retry	SecureTransport will retry executing the trigger file the specified number of times, with an optional delay between attempts.

- Files matching specific filename pattern - You can use this option independently or in combination with **Trigger processing of files based on condition**. It instructs SecureTransport to route only the files that match the expression set in the **Filename pattern** filed.

Trigger Route Execution on PeSIT Acknowledgment

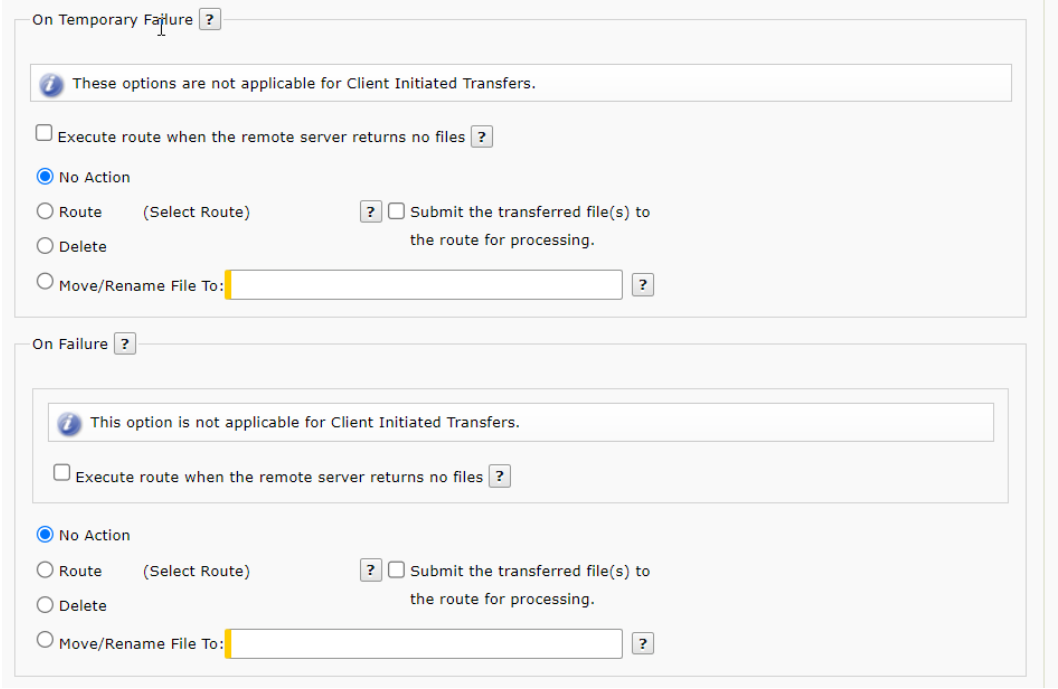
If you enable this option, route execution will be triggered on either of the following events:

- Successful upload of a file to the subscription folder: In this case, the file is automatically submitted for AR processing.
- When a PeSIT acknowledgment is received for a transfer of a file that has been in the subscription folder. If the file is still present in the subscription folder, it will be submitted for AR processing and the route with the matching PeSIT acknowledgment-based criteria will be executed. Otherwise, AR will be triggered without payload, allowing you to use this functionality to execute an External Script or a Pull From Partner step.

Refer to *Advanced use cases* on page 975 for use case examples that can help you create complex flows using PeSIT acknowledgment-based triggers.

Post-transmission settings on temporary and permanent failure

In case of route failure, SecureTransport can trigger a file operation or retry the route.



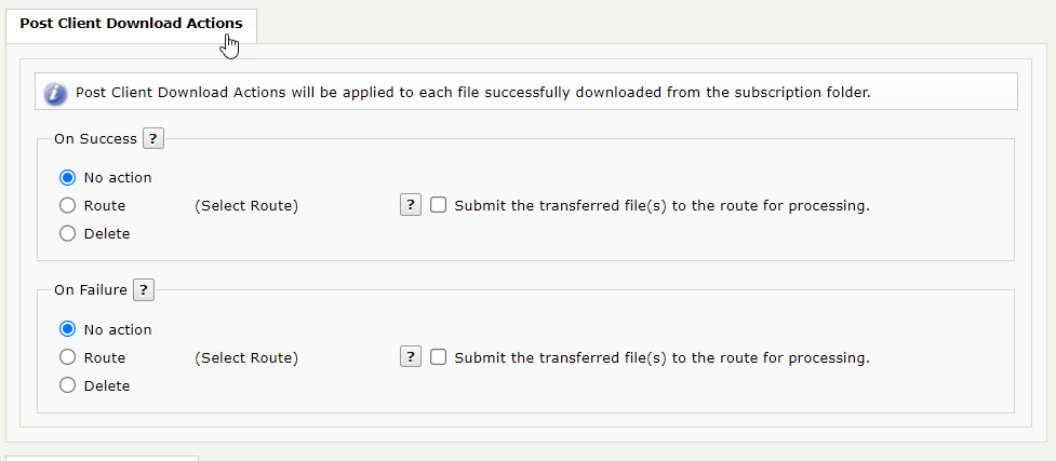
You can specify a different action to take in case of a permanent and temporary failure. The available actions are:

Action	Description
No Action	No actions take place on the failing files.
Route	The selected route will be triggered again without the failing files. To perform transformations on the failing files inside the selected route, select **Submit the transferred file(s) to the route for processing**.
Delete	Deletes the failing files.
Move/Rename File To	Moves or renames the failing file to a location or using the expression specified in the field.

Note The temporarily failed transfers will be retried. The temporary failure option is not applicable for client-initiated transfers.

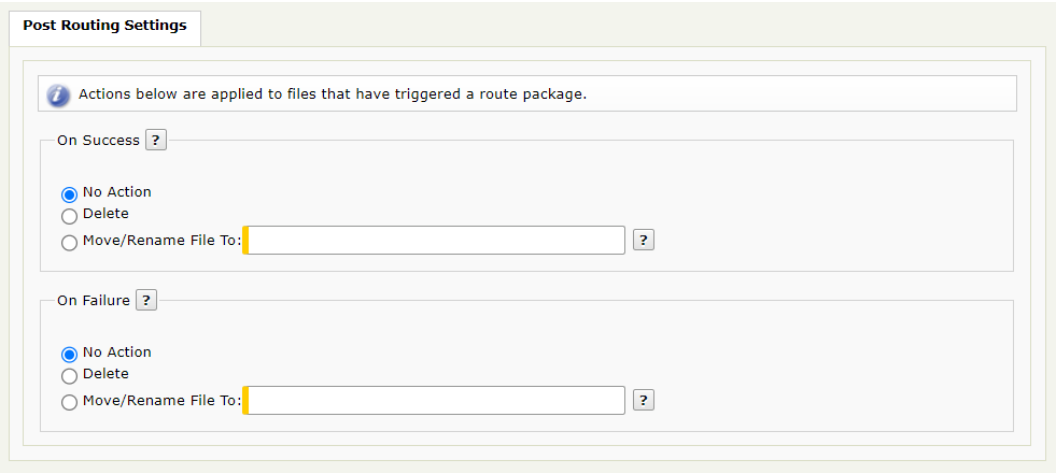
Select the post client download actions

Post Client Download Actions will be applied to each file downloaded from the subscription folder. You can configure a different action for successful and failed client downloads. The options are: **No Action**, **Route** with/without the downloaded file, and **Delete** the downloaded file from the subscription folder.



Select the post routing actions

The Post Routing Settings actions are applied to files that have triggered a route. You can configure a different action based on the whether the route has been successfully triggered or not.



Transformations

The following topics provide detailed how-to instructions for configuring transformations:

- *PGP Encryption on page 896*
- *PGP Decryption on page 901*
- *Compress on page 904*
- *Decompress on page 908*
- *Line Ending on page 912*

- *External Script on page 916*
- *Encoding Conversion on page 921*
- *Characters Replace on page 924*
- *Line Padding on page 929*
- *Line Truncating on page 933*
- *Line Folding on page 937*
- *Rename on page 940*

PGP Encryption

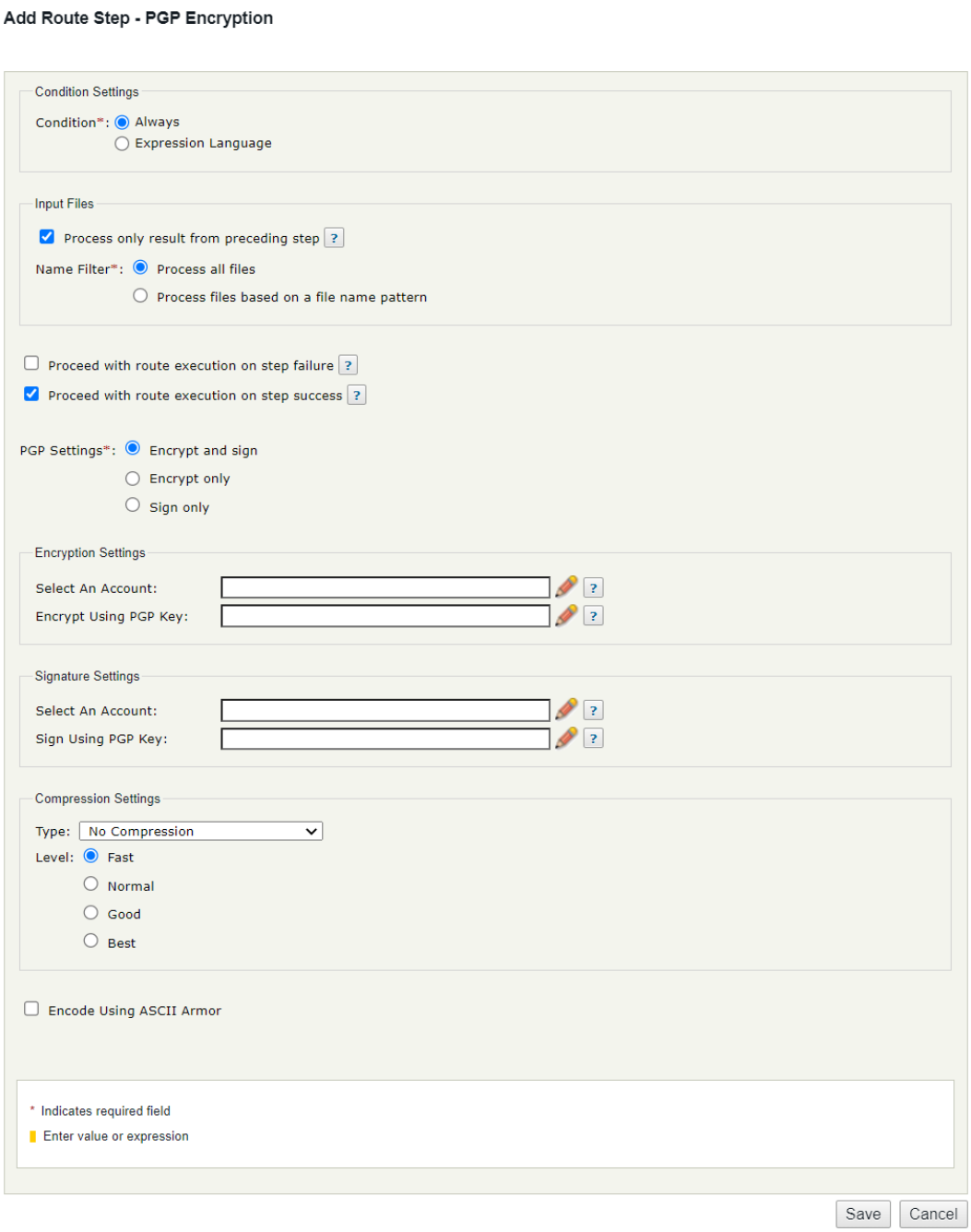
The PGP Encryption transformation step enables the encryption and signing of designated files as part of a route.

Use the following procedure to add a PGP Encryption transformation step to a route package template take the following steps:

1. Designate the files to be encrypted.
2. Determine whether to proceed with route execution on step failure and success.
3. Select PGP settings, including encryption and signature settings.
4. Set compression level and type.
5. Determine whether or not to ASCII armor encode transformed files.
6. Determine post transformation actions.

Note Steps 1 and 3 are mandatory. All other steps are optional.

The following figure provides an example of the PGP Encryption transformation step.



Note The buffer size for PGP encryption and decryption is controlled by the `Pgp.BufferSize` server configuration option.

The following sections provide configuration details for the PGP Encryption transformation step:

- *Condition Settings on page 898*
- *Input Files on page 898*
- *Proceed with route execution on step failure or success on page 899*
- *PGP Settings on page 899*

- *Encryption Settings on page 899*
- *Signature Settings on page 900*
- *Compression Settings on page 900*
- *Encode using ASCII Armor on page 901*
- *Post transformation action on page 901*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* settings consist of the selection *Process only result from preceding step* and determining the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **selected** only files produced by the preceding step will be used as input for this step. When the checkbox is or this is the first step all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

The Name Filter can either be set to process all files forwarded by the selection of *Process only result from preceding step* or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.
If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.
If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.

- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

PGP Settings

The PGP setting can be encrypt and sign, encrypt only, or sign only. If *Encrypt and sign* is selected, the files being processed by the route step are encrypted and signed. If *Encrypt only* is selected, the files being processes by the route step are encrypted but not signed. If *Sign only* is selected, the files being processed by the route step are signed but not encrypted.

By default, SecureTransport does not allow you to use the same RSA modulus to both sign and encrypt or verify and decrypt. To allow the use of the same RSA modulus for multiple purposes, add the following Java option in `<FILEDRIVEHOME>/bin/start_tm_console`:

```
-Dorg.bouncycastle.rsa.allow_multi_use=true
```

and restart the TM Server.

Encryption Settings

The encryption settings consist of selecting an account and the PGP key to use for encryption.

Select an account

Either an account name or an Expression Language (EL) string can be specified to determine the recipient based on the environment information (such as filename).

The *Select an account* field has auto-completion which shows a list of existing accounts containing the same letter.

Once an account is selected its publicly available PGP certificates are populated in the *Select an account* field. The certificates can be public for all SecureTransport accounts, or public for an account assigned to the same Business Unit.

If an account name is unknown (for example, expression based) its PGP certificates are determined at run time. PGP certificates can be expression based as well.

Encrypt using PGP key

A PGP Encryption key can be selected from PGP Public Keys (within the selected account) or by entering an expression string. The access level of PGP keys is determined by the select access level. The PGP key selected access level can be private, business unit, or public.

Wildcard symbols ('`*`' and '`?`') can be used when specifying the PGP key alias (for example, `.*-pgp`). If multiple keys match the pattern the first one is picked up and used.

Signature Settings

The signature settings consist of selecting the account and the PGP key to use for signing.

Select an account

Either an account name or an EL string can be specified to determine the recipient based on the environment information (such as filename).

The *Select an account* field has auto-completion which shows a list of existing accounts containing the same letter.

Once an account is selected its publicly available PGP certificates are populated in the *Select an account* field. The certificates can be public for all SecureTransport accounts, or public for an account assigned to the same Business Unit.

If an account name is unknown (for example, expression based) its PGP certificates are determined at run time. PGP certificates can be expression based as well.

Sign using PGP key

A PGP signature key can be selected from PGP Public Keys (within the selected account) or by entering an expression string.

Wild card symbols ('*' and '?') can be used when specifying the PGP key alias (for example, `.*-pgp`). If multiple keys match the pattern the first one is picked up and used.

Compression Settings

The compression settings consist of selecting the type and level of compression.

Type

The types of compression that can be selected are:

- No Compression
- Use Preferred
- ZIP
- ZLIB
- BZIP2

Level

The levels of compression that can be selected are:

- Fast
- Normal
- Good
- Best

Encode using ASCII Armor

If *Encode using ASCII Armor* is checked, the files processed by the route step are ASCII armor encoded.

Post transformation action

The output file names will be the same as the input file names. To change the file names use a Rename transformation step. To configure a rename transformation step, refer to *Rename* on page 940.

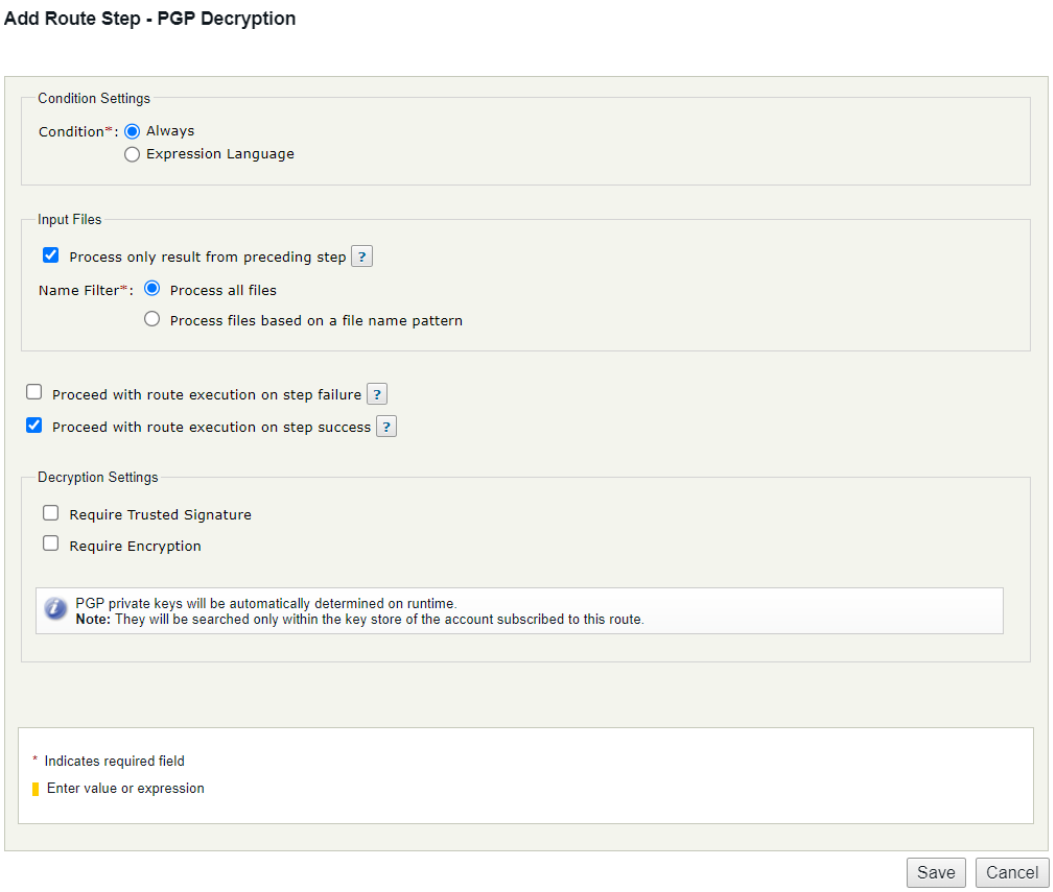
PGP Decryption

The PGP Decryption transformation step enables the decryption and signature verification of designated files as part of a route. To add a PGP Decryption transformation step to a route package template take the following steps:

1. Designate file filtering.
2. Determine whether to proceed with route execution on step failure and success.
3. Select decryption settings.
4. Determine post transformation actions.

Note Step 1 is mandatory. All other steps are optional.

The following figure provides an example of a PGP Decryption transformation step.



Note The buffer size for PGP encryption and decryption is controlled by the `Pgp.BufferSize` server configuration option.

The following sections provide configuration details for the PGP Decryption transformation step:

- *Condition Settings on page 902*
- *Input Files on page 903*
- *Proceed with route execution on step failure or success on page 903*
- *Decryption Settings on page 903*
- *Post transformation action on page 904*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* settings consist of the selection *Process only result from preceding step* and determining the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **enabled** only files produced by the preceding step will be used as input for this step.

When *Process only result from preceding step* is **disabled** or this is the first step all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

The Name Filter can either be set to process all files forwarded by the selection of *Process only result from preceding step* or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.
If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.
If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.

- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Decryption Settings

The two selections for decryption settings are:

- Require Trusted Signature
- Require Encryption

PGP private keys are automatically determined on runtime.

Note PGP private and public keys are only searched for within the key store of the account subscribed to this route.

Require Trusted Signature

If *Require Trusted Signature* is selected, the transformation of the designated file or files requires a PGP partner key for signature verification. If a signature is required, but the file is not signed, the signature verification fails. If encryption is required but the file is not encrypted, the PGP Decryption fails. If both trusted signature and encryption are required but the file is neither encrypted or signed, the decryption fails. PGP Decryption step also fails if there is a problem with the certificate selected (not valid, expired, not signed, and so forth).

Require Encryption

If *Require Encryption* is selected, the transformation of the designated file or files requires a PGP private key for decryption.

Post transformation action

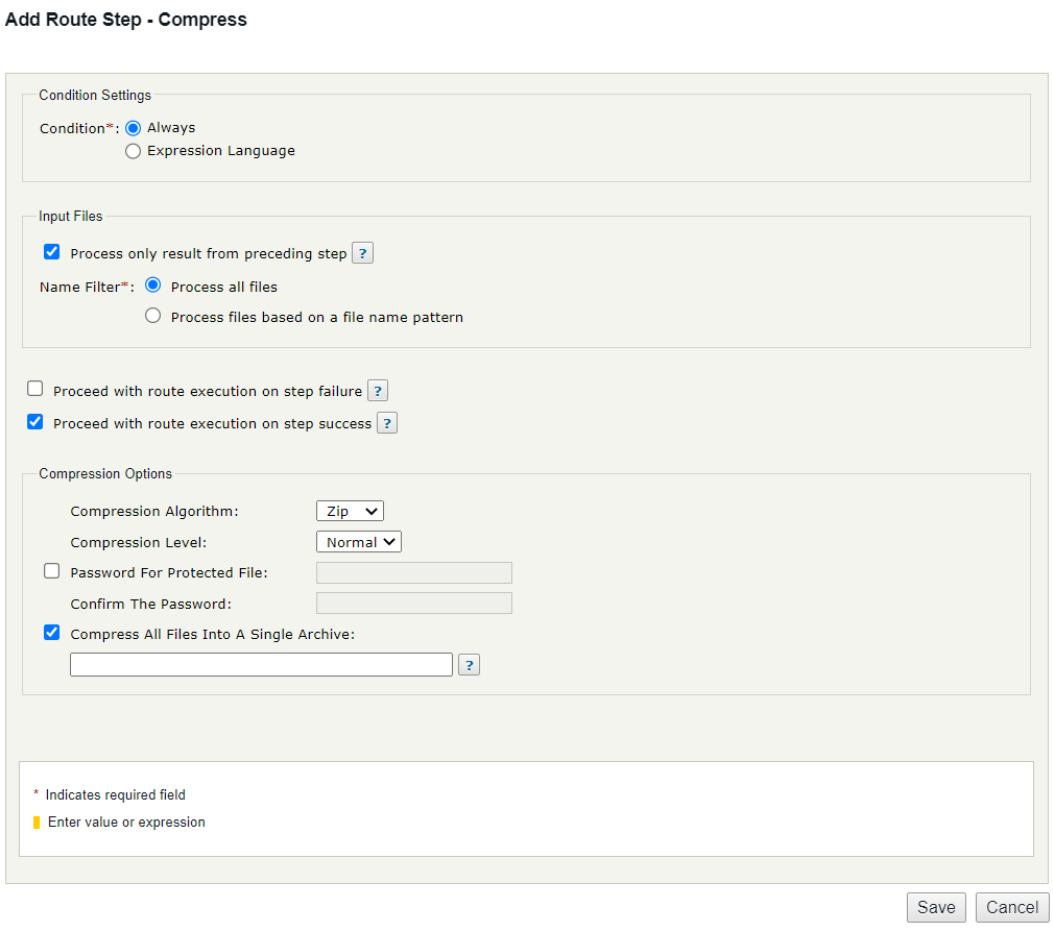
The output file names will be the same as the input file names. To change the file names use a Rename transformation step. To configure a Rename transformation step, refer to *Rename* on page 940.

Compress

The Compress transformation step compresses designated files as part of a route.

Use the following procedure to add a Compress step to a route package template:

1. Designate the files to be compressed.
2. Determine whether to proceed with the route execution on step failure and success.
3. Specify file compression options.
4. Click **Save**.



The following sections provide configuration details for the Compress transformation step:

- *Condition Settings on page 905*
- *Input Files on page 906*
- *File name patterns on page 906*
- *Proceed with route execution on step failure or success on page 906*
- *Compression Options on page 907*
- *Compress on page 904*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* pane identifies files used as input to the Compress step. By default, the step will compress all files that were produced by the preceding step. Use the filtering options to designate the files to be compressed.

The Compress step processes files in the following manner:

Process only result from preceding step checkbox status	**Name filter** option selected	**Input files** result
selected	**Process all files**	all output files generated by the preceding step
selected	**Process files based on a file name pattern**	the output files generated by the preceding step that match the specified file name pattern
not selected	**Process all files**	all available files
not selected	**Process files based on a file name pattern**	the available files that match the specified file name pattern

File name patterns

To use a file name pattern to identify files to compress:

1. Select **Process files based on a file name pattern**.
2. Choose pattern syntax.
   - When **File Globbing** is selected, you can use '`?`' and '`*`' wildcard characters to define the pattern. For example, `*.txt` matches all files with the extension `.txt`.
   - When **Regular expressions** is selected, you can use `Perl5.003` or Perl5 extended regular expressions. For example, `.*\.(txt|xml)` matches all files with the extension `.txt` or `.xml`.

   More examples of `glob` and `regexp` patterns are provided in the tooltip.
3. Provide the pattern in the text box.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.

- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Compression Options

In the *Compression Options* pane, you set the compression algorithm and level and choose whether to consolidate all input files in a single output archive. You can also password-protect the output ZIP files.

Compression algorithms

The supported compression algorithms are ZIP, JAR, TAR, and GZIP. The compression of `tar.gz` archives requires two separate Compress steps. The first one is to add the files to a `tar` archive, and the second - to compress the `tar` file with `gzip`.

Compression levels

The supported compression levels depend on the selected compression algorithm. With ZIP, GZip and JAR, you can choose among seven levels: *Store, Fastest, Fast, Normal, Good, Better,* and *Best*. *Store* adds the input files to archive without compression, and *Best* provides the highest, but slowest compression. As the compressed file's size decreases (from *Store* to *Best*), the time to compress increases.

With TAR, the only available level is *Normal* - compression with balanced settings.

Password protection

Password protection is only available with ZIP. By default, it uses 128-bit AES encryption.
To enable it, select the **Password for protected file** checkbox, type the desired password, and confirm it.

Number of output files and naming

With GZIP, the Compress step creates a separate archive for each input file.
With ZIP, TAR and JAR, you can use the **Compress all files into a single archive** checkbox to control the number of output files.

- When selected, all input files are compressed into a single archive. The name of that archive is defined using an EL expression that you enter in the text field below the option.

- When not selected, each input file is added to a separate archive. The name of a compressed file is composed of the full name (with the extension) of the input file and the compression algorithm appended as an extension.

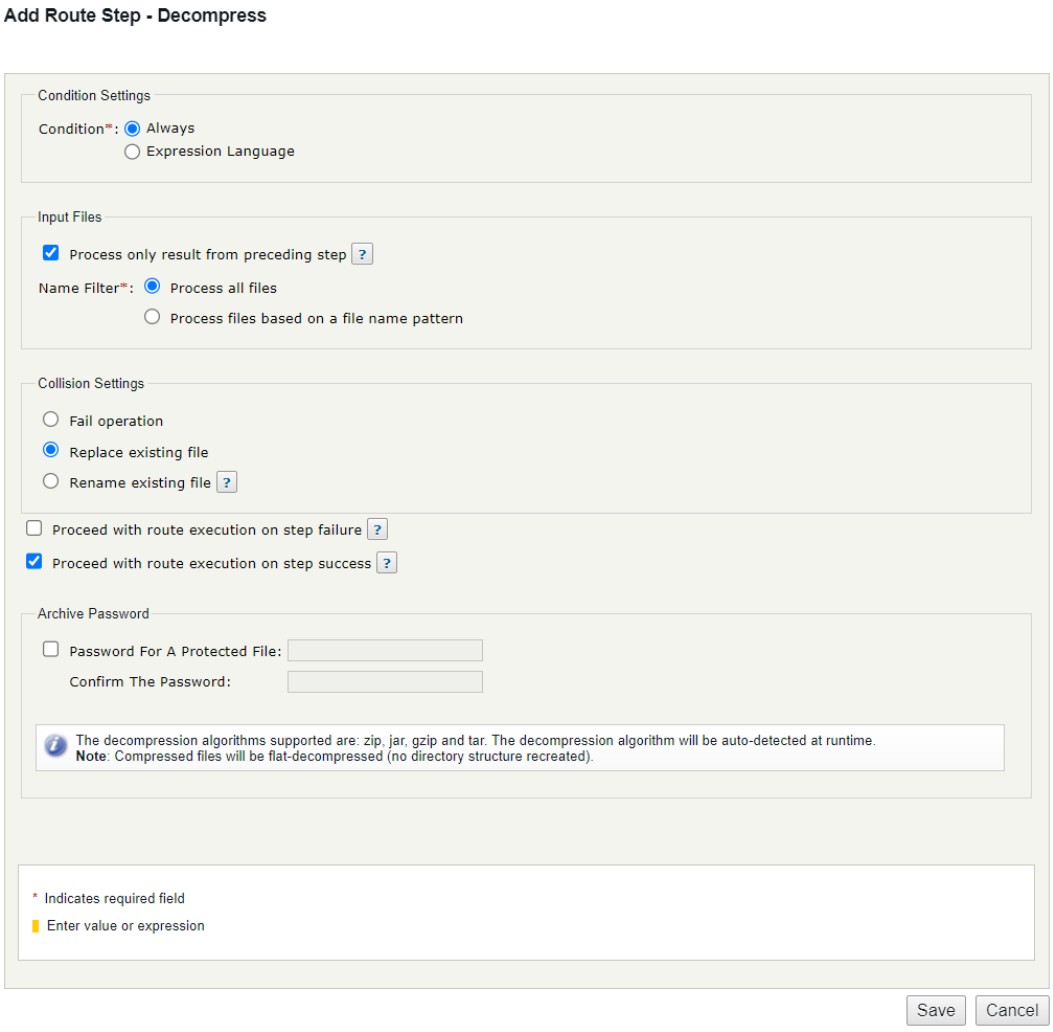
Afterward, you can change the name of the output files by adding a *Rename* on page 940 transformation step.

Decompress

The Decompress transformation step enables the decompression of designated archived files as part of a route. SecureTransport can extract files from ZIP, JAR, TAR and GZIP archives. The decompression algorithm is automatically detected at run time. A `tar.gz` archive requires two separate Decompress steps. The first one is to extract the `tar` from the `gzip` archive, and the second one - to decompress the `tar` archive.

Use the following procedure to add a Decompress step to a route package template:

1. Designate the files to be decompressed.
2. Choose the action to be taken if an archive contains a file with the same name.
3. Determine whether to proceed with the route execution on step failure and success.
4. Enter the password to unzip a ZIP file.
5. Click **Save**.



The following sections provide configuration details for the Decompress transformation step:

- *Condition Settings on page 909*
- *Input Files on page 910*
- *File name patterns on page 910*
- *Collision Settings on page 911*
- *Proceed with route execution on step failure or success on page 911*
- *Archive Password on page 911*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* pane identifies files used as input to the Decompress step. By default, the step will decompress all archive files that were produced by the preceding step. Use the filtering options to designate the files to be decompressed.

The Decompress step processes files in the following manner:

Process only result from preceding step checkbox status	**Name filter** option selected	**Input files** result
selected	**Process all files**	all output files generated by the preceding step
selected	**Process files based on a file name pattern**	the output files generated by the preceding step that match the specified file name pattern
not selected	**Process all files**	all available archive files
not selected	**Process files based on a file name pattern**	the available archive files that match the specified file name pattern

File name patterns

To use a file name pattern to identify files to decompress:

1. Select **Process files based on a file name pattern**.
2. Choose pattern syntax.
   - When **File Globbing** is selected, you can use '`?`' and '`*`' wildcard characters to define the pattern. For example, `*.txt` matches all files with the extension `.txt`.
   - When **Regular expressions** is selected, you can use `Perl5.003` or Perl5 extended regular expressions. For example, `.*\.(txt|xml)` matches all files with the extension `.txt` or `.xml`.

   More examples of `glob` and `regexp` patterns are provided in the tooltip.
3. Provide the pattern in the text box.

Collision Settings

The Decompress step discards the directory structure of the archive and extracts all files into the same directory. To avoid name collisions, you can instruct SecureTransport how to handle archives containing files with the same name:

The following options are available:

- **Fail operation** - When selected, an archive member with the same name as the archive prevents its extraction; an error is reported, and the step fails.
- **Replace existing file** - Default. When selected, the archive member is extracted into a file with the same name, and the archive is deleted.
- **Rename existing file** - When selected, name collisions are resolved automatically by appending *(new copy <number>)* to the extracted archive member that shares its name with the archive. The archive name remains unchanged.

  For example, if an archive named *myFile* contains a file named *myFile*, the archived file will be renamed to *myFile (new copy 1)*. If a file with the name *myFile (new copy 1)* already exists, the extracted file will be renamed to *myFile (new copy 2)*.

If there are files with the same name in different folders inside the archive, only one of them is extracted. For example, if a ZIP file contains two folders (`folder1` and `folder2`) and each of the folders contains a file named `file.txt` (`folder1\file.txt` and `folder2\file.txt`), only one copy of `file.txt` is extracted.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.
- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Archive Password

This option is only applicable for password-protected ZIP files. To unzip such files, select the **Password for a protected file** checkbox, enter the password for the file and confirm it.

Output file naming

The output file name is the same as the input file name. To change the names of the output files, add a *Rename* on page 940 transformation step.

Line Ending

The Line Ending transformation step enables converting line ending formats as part of a route.

Note Currently, the Line Ending transformation accepts Unicode or ASCII as an input, or mixed input when the custom line ending format is selected.

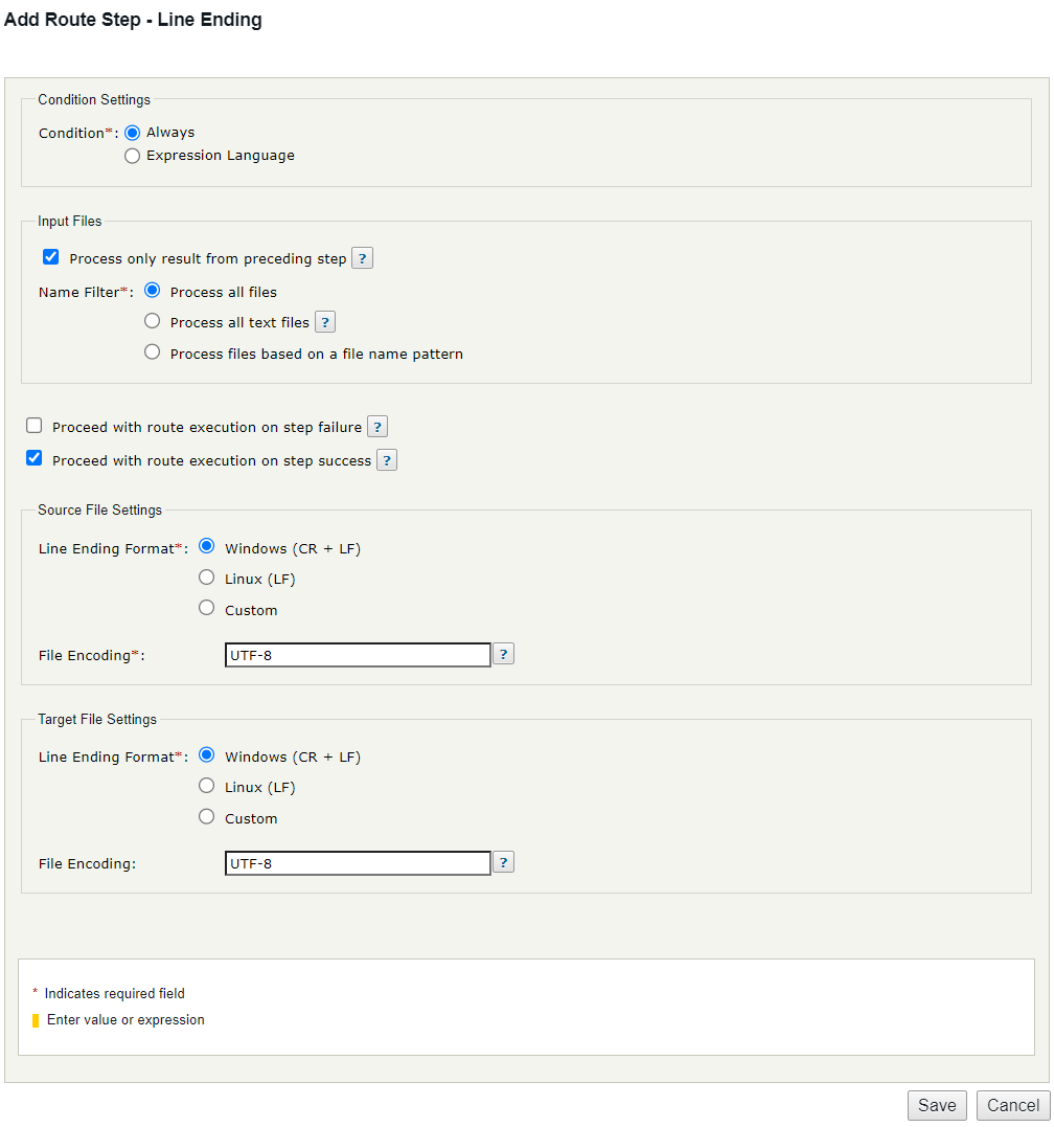
Note If the encoding of a file is changed to IBM500, IBM037, or IBM424, the line ending of the input file should be LF and not CRLF. The CR is preserved in some cases and a new line appears between every original line.

To add a Line Ending transformation step to a route package template take the following steps:

1. Designate file filtering.
2. Determine whether to proceed with route execution on step failure and success.
3. Select source file setting options.
4. Select target file setting options.
5. Determine post transformation actions.

Note Steps 1, 3, and 4 are mandatory. All other steps are optional.

The following figure provides an example of a Line Ending transformation step.



The following sections provide configuration details for the Line Ending transformation step:

- *Condition Settings on page 913*
- *Input Files on page 914*
- *Proceed with route execution on step failure or success on page 915*
- *Source file settings on page 915*
- *Target file settings on page 916*
- *Post transformation action on page 916*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* settings consist of the selection *Process only result from preceding step* and determining the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **enabled** only files produced by the preceding step will be used as input for this step.

When *Process only result from preceding step* is **disabled** or this is the first step all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

Name filtering can either be set to process all files, process all text files, or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.

If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.

If *Process all text files* is selected, all text files forwarded by the selection of *Process only result from preceding step* are processed.

Text file selection is determined based on the file extension and the MIME type that corresponds to that extension. MIME types are defined in the `mime.types` configuration file located in:

Administrator's Tool > Server Configuration > Configuration Files

Furthermore, some MIME types can be defined as text by configuring the `AdvancedRouting.TextMimeSubtypes` server configuration option. For example, to configure the `application/xml` MIME type to be treated as text, add `xml` to the server configuration option.

If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.

- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Source file settings

The source file settings consists of selecting a source file line ending format and encoding.

Line ending format

The selectable line ending formats are:

- Windows (CR + LF)
- Unix (LF)
- Custom

If *Custom* is selected, specify line ending characters in ASCII or Unicode format (\uXXXX). The hex encoded value of the line ending is any character `\n`, `\r`, and the combination of both. The custom line ending char in Unicode notation:

- Windows:

  ```
  \u000d\u000a
  ```

- *nix, MacOS:

  ```
  \u000a
  ```

- Mainframe:

  ```
  \u0025
  ```

File encoding

The file encoding format can be selected from a long list of available formats. Start typing the desired file encoding format in the field and select it from the list. For a supported list of source and target encoding, refer to Java SE 11 Documentation.

Target file settings

The target file settings consists of selecting a target file line ending format and encoding.

Line ending format

The selectable line ending formats are:

- Windows (CR + LF)
- Unix (LF)
- Custom

If *Custom* is selected, the hex encoded value of the line ending character must be specified. The hex encoded value of the line ending is any character `\n`, `\r` combination of both. The custom line ending char in Unicode notation:

- Windows:

  ```
  \u000d\u000a
  ```

- *nix, MacOS:

  ```
  \u000a
  ```

- Mainframe:

  ```
  \u0025
  ```

File encoding

The file encoding format can be selected from a long list of available formats. Start typing the desired file encoding format in the field and select it from the list.

Post transformation action

The output file names will be the same as the input file names. To change the file names use a Rename transformation step. To configure a Rename transformation step, refer to *Rename on page 940*.

External Script

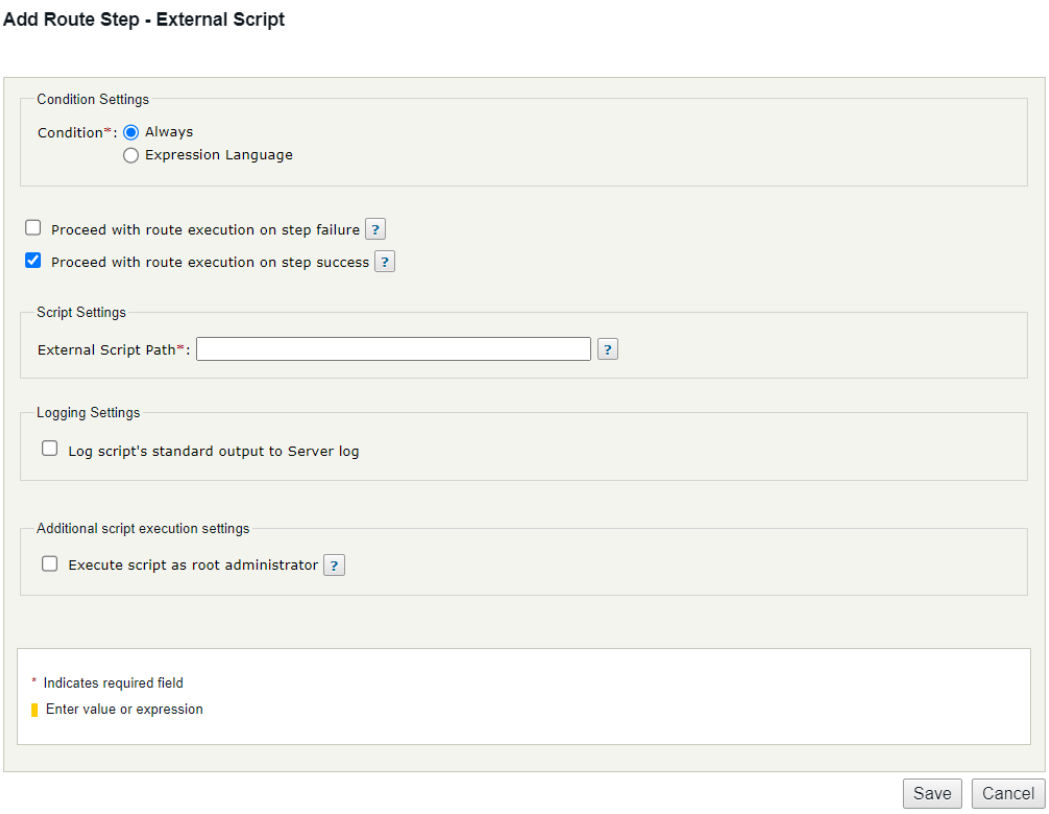
Note The External Script transformation step does not function with repository encryption.

The External Script transformation step enables adding the execution of an external script as part of a route. To add an External Script transformation step to a route package template, take the following steps:

1. Determine whether to proceed with route execution on step failure and success.
2. Select the external script path.
3. Determine whether or not to log the external script's standard output to the server log.
4. Select whether or not to run scripts as the root administrator.

Note Step 2 is mandatory. All other steps are optional.

The following figure provides a configured example of an External Script transformation step.



The following sections provide configuration details for the External Script transformation step:

- *Condition Settings on page 918*
- *Proceed with route execution on step failure or success on page 918*
- *Script Settings on page 918*
- *Logging Settings on page 919*
- *Additional script execution settings on page 919*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.
- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Script Settings

The script settings consist of the selecting the external script path.

External script path

The external script path is an absolute path to external script.

Example script expressions:

- For *nix environment:
  ```
  /usr/bin/env bash -c "${FILEDRIVEHOME}/bin/agents/example.sh
  ${date('yyyyMMdd')} ${currentfulltarget}
  ${transfer.transferredBytes} ${routing.originalFiles}"
  ```

- For Windows environment:
  ```
  cmd /c ${FILEDRIVEHOME}\bin\agents\example.bat ${date
  ('yyyyMMdd')} ${currentfulltarget} ${transfer.transferredBytes}
  ${routing.originalFiles}
  ```

Logging Settings

The logging settings consists of determining whether the external script's standard output is logged to the server log.

Log script's standard output to Server log

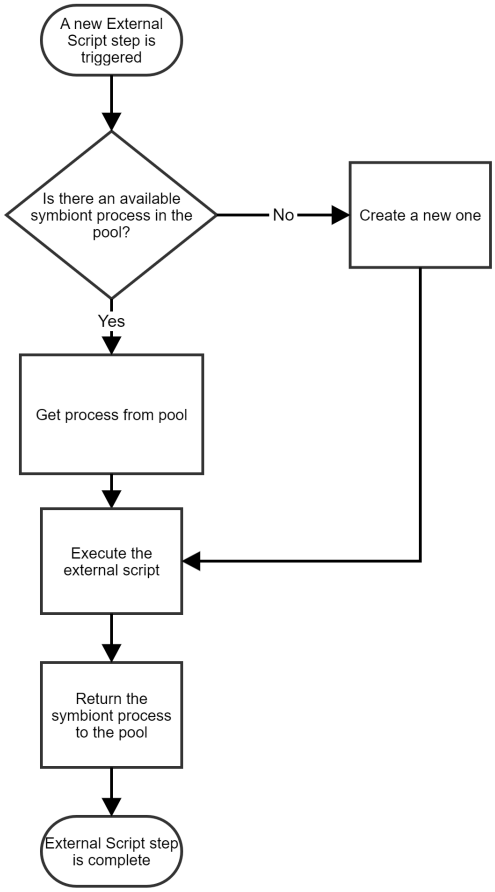
If *Log script's standard output to Server log* is selected, the external script's standard output is logged to the server log.

Additional script execution settings

To perform an External Script transformation step, SecureTransport starts a process known as *symbiont*. The symbiont then runs a second process which allows for the external script to be executed with specific user permissions (instead of root) for security reasons. The external script is executed with the UID and GID of the user account which triggered the route.

Before an External Script step is performed, SecureTransport checks if there is an available symbiont in the symbiont pool. If there isn't, a new one is created. If there is, it is used to execute the script, and is then returned back to the pool. If the symbiont pool becomes too big and resource-intensive, you can clear it by restarting the Transaction Manager, or you can use the `kill` command on separate symbiont process IDs. Note that terminating a symbiont while it's running a script results in termination of the script execution.

The following diagram provides a visual representation of the symbiont life cycle.



External script steps can also be configured to run scripts as a root administrator for each individual step instance. When running scripts as a system superuser, the scripts have the ability to execute the full scope of commands, thus exposing the system in the hands of the writer of the script.

The general recommendation is to avoid using this option, or to use it with caution. It is therefore unchecked by default.

For more information, refer to the *SecureTransport 5.5 Security guide*.

Note Running scripts as root is not the default behavior. Enabling it makes it possible to run commands which the routing step might not have permissions to execute otherwise.

Note This option is available only to administrators with sufficient levels of permissions.
By default, master and account administrators can manage the option.
For delegated administrator privileges, see *Delegated administration* on page 707.

Note When running scripts via the External Script routing step, the execution environment might not have full scope of environment variables initialized. The script writer is responsible for properly exporting and initializing the necessary environment in the script itself before the actual script execution specifics.

Note The option is applicable only for root and non-Windows deployments.

Encoding Conversion

The Encoding Conversion transformation converts the character encoding of an input file to another configured encoding. Both source file encoding and output file encoding must be configured in the transformation step settings.

Specifics:

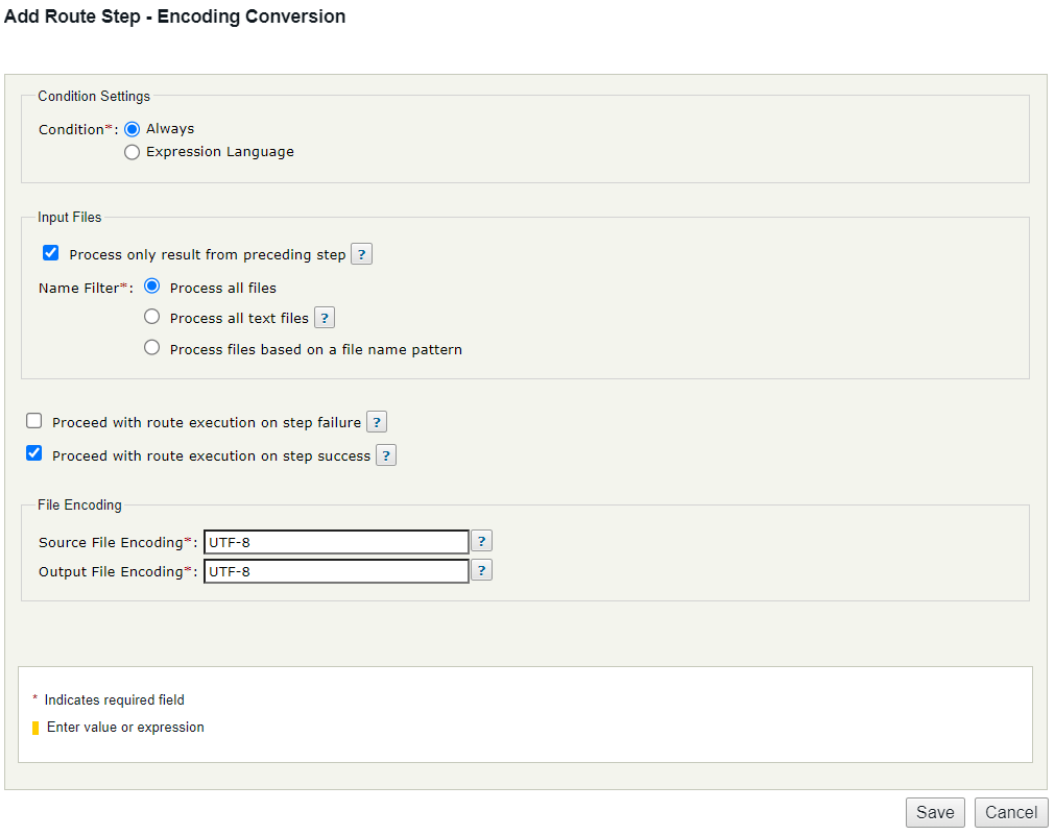
- When using the AR Encoding Conversion step to convert an EBCDIC file from IBM1047 to UTF-8, the EBCDIC line feed hex'25 is converted to NEL (hex'85) in UTF-8. To convert the line feeds to UTF-8 LF hex'0A, use the Transfer Profile Advanced Properties. See *Transfer profile: Advanced Properties on page 645*.
- If the encoding a file is changed to IBM500, IBM037, or IBM424; the line ending of the input file should be LF and not CRLF. The CR is preserved in some cases and a new line appears between every original line. If the line ending is not LF, a Line Ending step can be used before the Encoding Conversion step.

To add an Encoding Conversion transformation step to a route package template, take the following steps:

1. Designate file filtering.
2. Determine whether to proceed with route execution on step failure and success.
3. Select source file encoding option.
4. Select output file encoding option.
5. Determine post transformation actions.

Note Steps 1, 3, and 4 are mandatory. All other steps are optional.

The following figure provides an example of an Encoding Conversion transformation step.



The following sections provide configuration details for the Encoding Conversion transformation step:

- *Condition Settings on page 922*
- *Input Files on page 923*
- *Proceed with route execution on step failure or success on page 923*
- *File encoding on page 924*
- *Post transformation action on page 924*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The Input Files settings consist of the selection *Process only result from preceding step* and determining the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **enabled** only files produced by the preceding step will be used as input for this step.

When *Process only result from preceding step* is **disabled** or this is the first step all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

Name filtering can either be set to process all files, process all text files, or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.

If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.

If *Process all text files* is selected, all text files forwarded by the selection of *Process only result from preceding step* are processed.

Text file selection is determined based on the file extension and the MIME type that corresponds to that extension. MIME types are defined in the `mime.types` configuration file located in:

Administrator's Tool > Server Configuration > Configuration Files

Furthermore, some MIME types can be defined as text by configuring the `AdvancedRouting.TextMimeSubtypes` server configuration option. For example, to configure the `application/xml` MIME type to be treated as text, add `xml` to the server configuration option.

If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.

- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

File encoding

Both source file encoding and output file encoding must be configured.

Source file encoding

Encoding of the file that will be transformed specified by the name of a SecureTransport supported character set. The **Source file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Note Specifying an encoding that doesn't match the original file encoding may produce an invalid result.

Output file encoding

Encoding of the file that will be produced specified by the name of a SecureTransport supported character set. The **Output file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Post transformation action

The output file names will be the same as the input file names. To change the file names use a Rename transformation step. To configure a Rename transformation step, refer to *Rename on page 940*.

Characters Replace

Note Currently, the Character Replace transformation accepts Unicode or ASCII as an input, or mixed input for the **Find** and **Replace** fields.

Note If the encoding of a file is changed to IBM500, IBM037, or IBM424, the line ending of the input file should be LF and not CRLF. The CR is preserved in some cases and a new line appears between every original line. If the line ending is not LF, a Line Ending step can used before the Characters Replace step.

Note The comma (", ") symbol can be used in ASCII only as a separator and not as find criteria. For find criteria, the comma symbol can only be used in Unicode.

The Characters Replace transformation has two working modes. Only one mode can be active at a time.

Find/Replace mode

The Characters Replace transformation will search the input file for character sequences matching the specified search sequences and if a match is found it will be replaced with the replace character sequence. The search and replace character sequences are specified in the transformation step settings. More than one search sequence can be configured separating the sequences with a comma character.

This mode can be configured in one of two ways:

1. The number of find sequences is one or greater. The replace sequence is only 1.

   Multiple find sequences are specified separated with commas (", ") and only a single replace sequence. In this case all find sequences have the same corresponding replace sequence. The transformation will search the input file for matches to the specified find sequences. If multiple sequences match the same text for replacement, only the first sequence that is found to fully match the text will be replaced.

   Note Empty replace sequence is a valid configuration. If such is specified, when a search match is found the matching sequence will effectively be removed from the file content.

2. The number of find and replace sequences is the same.

   In this case each find character sequence has its own replace character sequence. The correlation between find and replace sequences is based on their position in the configuration (the first find sequence corresponds to the first replace sequence, the second find to the second replace and so on).

   The transformation will search the input file for matches to the specified search sequences. When a match to a find sequence is found, it will be replaced by its own corresponding replace sequence. If multiple sequences match the same text for replacement, only the first sequence that is found to fully match the text will be replaced.

Find/Line strip

The Characters Replace transformation will strip all file lines starting with a specified search character sequence. The search character sequence is specified in the transformation step settings. More than one search sequence can be configured separating the sequences with a comma character.

The transformation will search the input file for matches at the start of each row to the specified find sequences. If a match is found, the row will be removed from the file content.

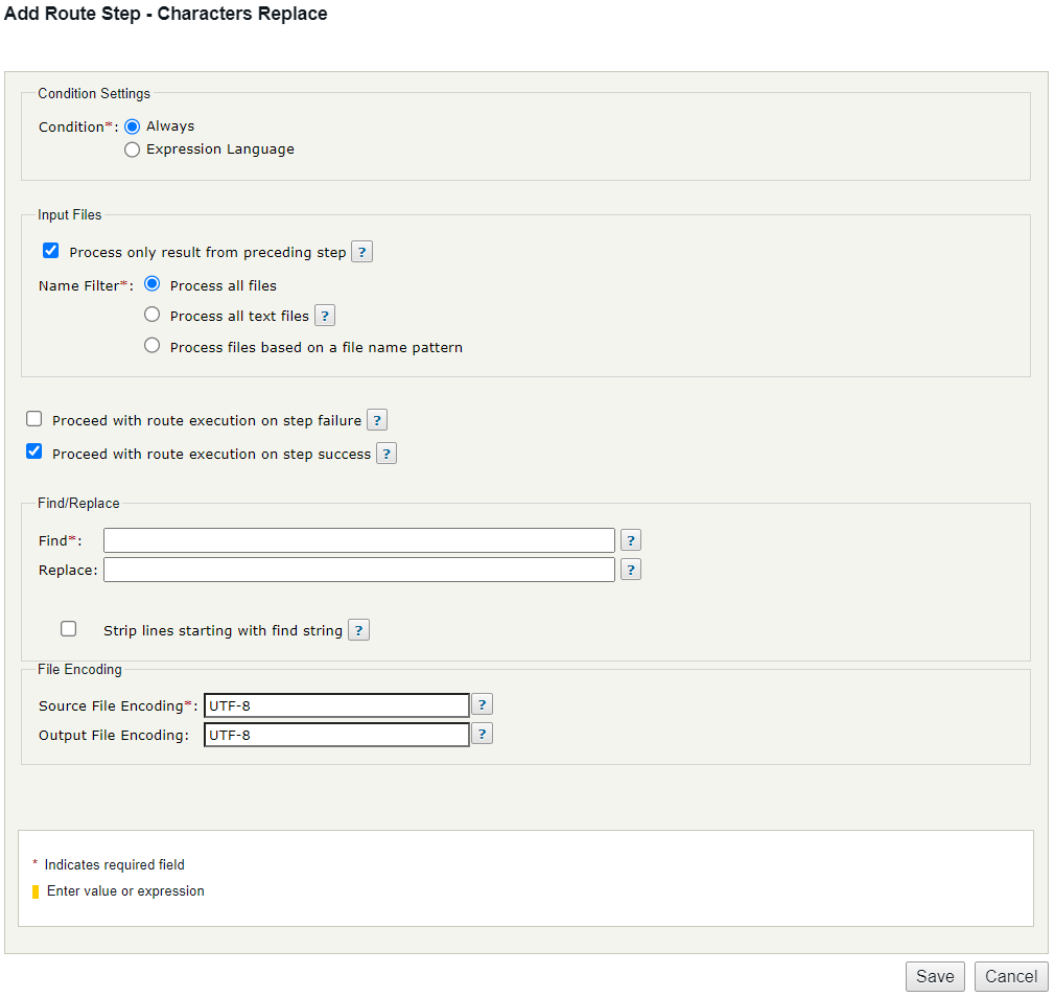
Characters Replace configuration

To add a Characters Replace transformation step to a route package template, take the following steps:

1. Designate file filtering.
2. Determine whether to proceed with route execution on step failure and success.
3. Determine the working mode. If you select the Find/Replace mode, complete the **Find** and **Replace** fields. If you select the Find/Line strip mode, select **Strip lines starting with find string.** and complete the **Find** field.
4. Select source file encoding option.
5. Select output file encoding option.
6. Determine post transformation actions.

Note Steps 1, 3, and 4 are mandatory. All other steps are optional.

The following figure provides an example of a Characters Replace transformation step.



The following sections provide configuration details for the Replace transformation step:

- *Condition Settings on page 927*
- *Input Files on page 927*
- *Proceed with route execution on step failure or success on page 928*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* settings consist of the selection *Process only result from preceding step* and determining the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **enabled** only files produced by the preceding step will be used as input for this step.

When *Process only result from preceding step* is **disabled** or this is the first step all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

Name filtering can either be set to process all files, process all text files, or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.

If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.

If *Process all text files* is selected, all text files forwarded by the selection of *Process only result from preceding step* are processed.

Text file selection is determined based on the file extension and the MIME type that corresponds to that extension. MIME types are defined in the `mime.types` configuration file located in:

Administrator's Tool > Server Configuration > Configuration Files

Furthermore, some MIME types can be defined as text by configuring the `AdvancedRouting.TextMimeSubtypes` server configuration option. For example, to configure the `application/xml` MIME type to be treated as text, add `xml` to the server configuration option.

If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.

- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Find/Replace mode

If Find/Replace mode, complete the **Find** and **Replace** fields.

Find

Specify the find character sequence. Unicode notation (\uXXXX) can be used. Multiple find character sequences separated with a comma (,) can be specified.

Note Comma must be Unicode encoded (\002c) if used in the find sequence.

Replace

Specify the replace character sequence. Unicode notation (\uXXXX) can be used. Multiple replace character sequences separated with a comma (,) can be specified. The number of replace sequences must be equal to the number of find sequences or just a single sequence.

Note Leaving this field blank is a valid configuration. Empty replace sequence is treated as an empty character.

Note Comma must be Unicode encoded (\002c) if used in the replace sequence.

Find/Line strip

If Find/Line strip, select **Strip lines starting with find string.** and complete the **Find** field.

File encoding

The source file encoding must be configured. Configuring the output file encoding is not mandatory. If it is not set, the output file encoding will be the same as the source file encoding.

Source file encoding

Encoding of the file that will be transformed specified by the name of a SecureTransport supported character set. The **Source file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Note Specifying an encoding that doesn't match the original file encoding may produce an invalid result.

Output file encoding

Encoding of the file that will be produced specified by the name of a SecureTransport supported character set. The **Output file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Post transformation action

The output file names will be the same as the input file names. To change the file names use a Rename transformation step. To configure a Rename transformation step, refer to *Rename on page 940*.

Line Padding

The Line Padding transformation will pad input text file lines to a configured length "X" with a configured character "C". If a file line length is longer than or equal to "X" number of characters then the line will be outputted to the result file without change. If a line length is shorter than "X" number of characters, then it will be padded out with character "C" to the length "X".

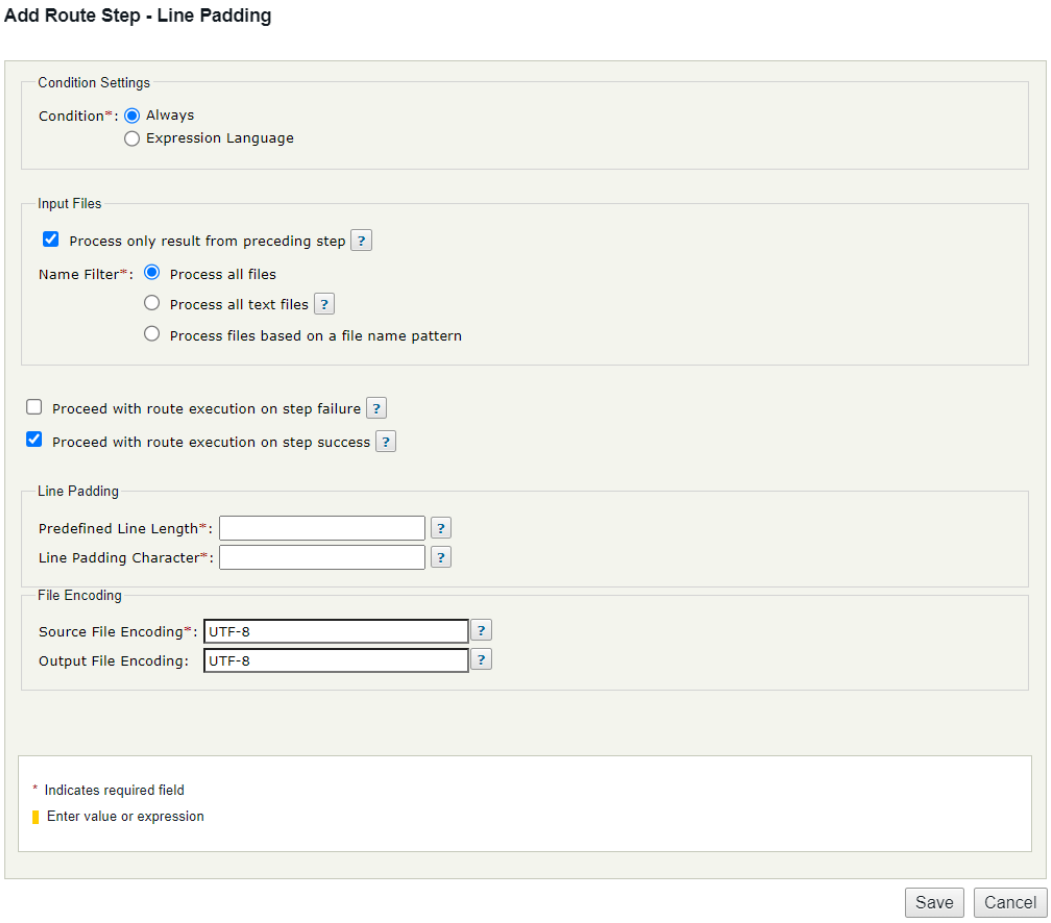
To add a Line Padding transformation step to a route package template, take the following steps:

1. Designate file filtering.
2. Determine whether to proceed with route execution on step failure and success.
3. Determine line padding length.
4. Determine line padding character.
5. Select source file encoding.

6. Select output file encoding.
7. Determine post transformation actions.

Note Steps 1, 3, 4, and 5 are mandatory. All other steps are optional.

The following figure provides an example of a Line Padding transformation step.



The following sections provide detailed Line Padding transformation configuration information:

- *Condition settings on page 930*
- *Input Files on page 931*
- *Proceed with route execution on step failure or success on page 932*
- *Line Padding on page 932*
- *File encoding on page 932*
- *Post transformation action on page 933*

Condition settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* settings consist of the selection *Process only result from preceding step* and determining the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **enabled** only files produced by the preceding step will be used as input for this step.

When *Process only result from preceding step* is **disabled** or this is the first step all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

Name filtering can either be set to process all files, process all text files, or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.

If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.

If *Process all text files* is selected, all text files forwarded by the selection of *Process only result from preceding step* are processed.

Text file selection is determined based on the file extension and the MIME type that corresponds to that extension. MIME types are defined in the `mime.types` configuration file located in:

Administrator's Tool > Server Configuration > Configuration Files

Furthermore, some MIME types can be defined as text by configuring the `AdvancedRouting.TextMimeSubtypes` server configuration option. For example, to configure the `application/xml` MIME type to be treated as text, add `xml` to the server configuration option.

If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.

- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Line Padding

Both the line padding length and line padding character must be configured.

Line padding length

The line padding length must be specified in number of characters. The maximum line padding length is 1024 characters.

Line padding character

The line padding character is specified in Unicode format with `\uXXXX` where `XXXX` is the hexadecimal representation of the characters. Only one character is accepted.

File encoding

The source file encoding must be configured. Configuring the output file encoding is not mandatory. If it is not set, the output file encoding will be the same as the source file encoding.

Source file encoding

Encoding of the file that will be transformed specified by the name of a SecureTransport supported character set. The **Source file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Note Specifying an encoding that doesn't match the original file encoding may produce an invalid result.

Output file encoding

Encoding of the file that will be produced specified by the name of a SecureTransport supported character set. The **Output file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Post transformation action

The output file names will be the same as the input file names. To change the file names use a Rename transformation step. To configure a Rename transformation step, refer to *Rename on page 940*.

Line Truncating

The Line Truncating transformation will truncate each line of input text file to a maximum length. The maximum length is specified in the transformation step settings. If a file line length is shorter than or equal to the configured maximum truncate length, then the line will be outputted to the result file without change. If a line length is longer than the configured maximum length "X" then only the first "X" number of characters will be outputted to the result file.

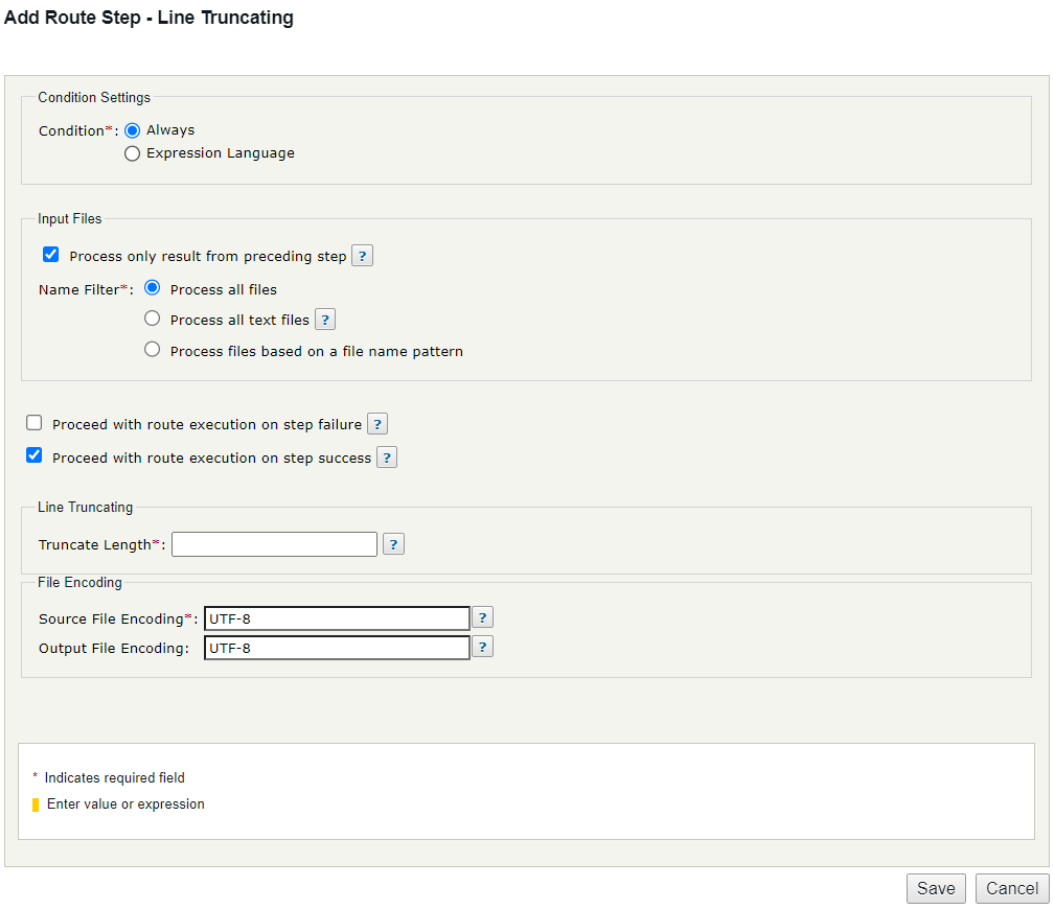
Note If the encoding of a file is changed to IBM500, IBM037, or IBM424, the line ending of the input file should be LF and not CRLF. The CR is preserved in some cases and a new line appears between every original line. If the line ending is not LF, a Line Ending step can used before the Line Truncating step.

To add a Line Truncating transformation step to a route package template, take the following steps:

1. Designate file filtering.
2. Determine whether to proceed with route execution on step failure and success.
3. Determine truncate length.
4. Select source file encoding.
5. Select output file encoding.
6. Determine post transformation actions.

Note Steps 1, 3, and 4 are mandatory. All other steps are optional.

The following figure provides an example of a Line Truncating transformation step.



The following sections provide configuration details for the Line Truncating transformation step:

- *Condition Settings on page 934*
- *Input Files on page 935*
- *Proceed with route execution on step failure or success on page 935*
- *Line truncating on page 936*
- *File encoding on page 936*
- *Post transformation action on page 936*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* settings consist of the selection *Process only result from preceding step* and determining the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **enabled** only files produced by the preceding step will be used as input for this step.

When *Process only result from preceding step* is **disabled** or this is the first step all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

Name filtering can either be set to process all files, process all text files, or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.

If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.

If *Process all text files* is selected, all text files forwarded by the selection of *Process only result from preceding step* are processed.

Text file selection is determined based on the file extension and the MIME type that corresponds to that extension. MIME types are defined in the `mime.types` configuration file located in:

Administrator's Tool > Server Configuration > Configuration Files

Furthermore, some MIME types can be defined as text by configuring the `AdvancedRouting.TextMimeSubtypes` server configuration option. For example, to configure the `application/xml` MIME type to be treated as text, add `xml` to the server configuration option.

If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.

- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Line truncating

The truncate length must be configured.

Truncate length

Maximum file line truncate length as specified in number of characters. The maximum number of characters is **TBD**.

File encoding

The source file encoding must be configured. Configuring the output file encoding is not mandatory. If it is not set, the output file encoding will be the same as the source file encoding.

Source file encoding

Encoding of the file that will be transformed specified by the name of a SecureTransport supported character set. The **Source file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Note Specifying an encoding that doesn't match the original file encoding may produce an invalid result.

Output file encoding

Encoding of the file that will be produced specified by the name of a SecureTransport supported character set. The **Output file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Post transformation action

The output file names will be the same as the input file names. To change the file names use a Rename transformation step. To configure a Rename transformation step, refer to *Rename on page 940*.

Line Folding

The Line Folding transformation will break file lines to a maximum width specified in number of characters. The maximum file width is specified in the transformation step settings. If a file line exceeds the maximum width, the extra characters will be moved to the next line. The moved extra characters are moved to a line of their own and not appended in front of the next line.

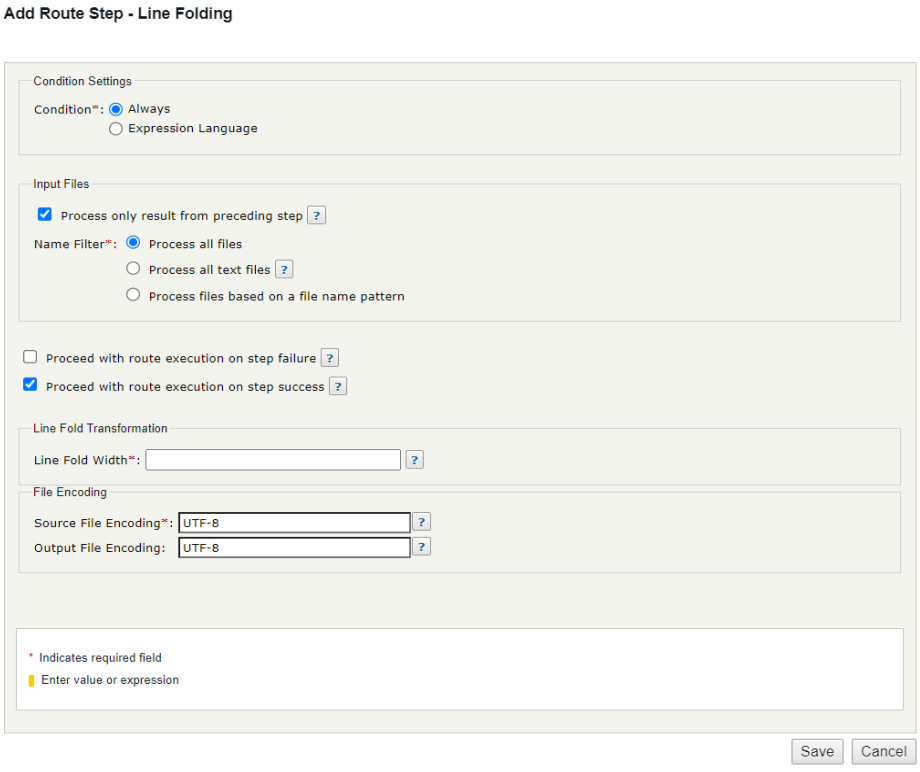
Note If the encoding of a file is changed to IBM500, IBM037, or IBM424, the line ending of the input file should be LF and not CRLF. The CR is preserved in some cases and a new line appears between every original line. If the line ending is not LF, a Line Ending step can used before the Line Folding step.

To add a Line Folding transformation step to a route package template take the following steps:

1. Designate file filtering.
2. Determine whether to proceed with route execution on step failure and success.
3. Determine file fold width.
4. Select source file encoding.
5. Select output file encoding.
6. Determine post transformation actions.

Note Steps 1, 3, and 4 are mandatory. All other steps are optional.

The following figure provides an example of a Line Folding transformation step.



The following sections provide configuration details for the Line Folding transformation step.:

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* settings consist of the selection *Process only result from preceding step* and determining the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **enabled** only files produced by the preceding step will be used as input for this step.

When *Process only result from preceding step* is **disabled** or this is the first step all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

Name filtering can either be set to process all files, process all text files, or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.

If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.

If *Process all text files* is selected, all text files forwarded by the selection of *Process only result from preceding step* are processed.

Text file selection is determined based on the file extension and the MIME type that corresponds to that extension. MIME types are defined in the `mime.types` configuration file located in:

Administrator's Tool > Server Configuration > Configuration Files

Furthermore, some MIME types can be defined as text by configuring the `AdvancedRouting.TextMimeSubtypes` server configuration option. For example, to configure the `application/xml` MIME type to be treated as text, add `xml` to the server configuration option.

If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.
- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

File fold transformation

The file fold width must be configured.

File fold width

Maximum allowed file line width as specified in number of characters. The maximum number of characters is 1024.

File encoding

The source file encoding must be configured. Configuring the output file encoding is not mandatory. If it is not set, the output file encoding will be the same as the source file encoding.

Source file encoding

Encoding of the file that will be transformed specified by the name of a SecureTransport supported character set. The **Source file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Note Specifying an encoding that doesn't match the original file encoding may produce an invalid result.

Output file encoding

Encoding of the file that will be produced specified by the name of a SecureTransport supported character set. The **Output file encoding:** field has an auto-complete function that lists matching SecureTransport supported character sets.

Post transformation action

The output file names will be the same as the input file names. To change the file names use a Rename transformation step. To configure a Rename transformation step, refer to *Rename on page 940*.

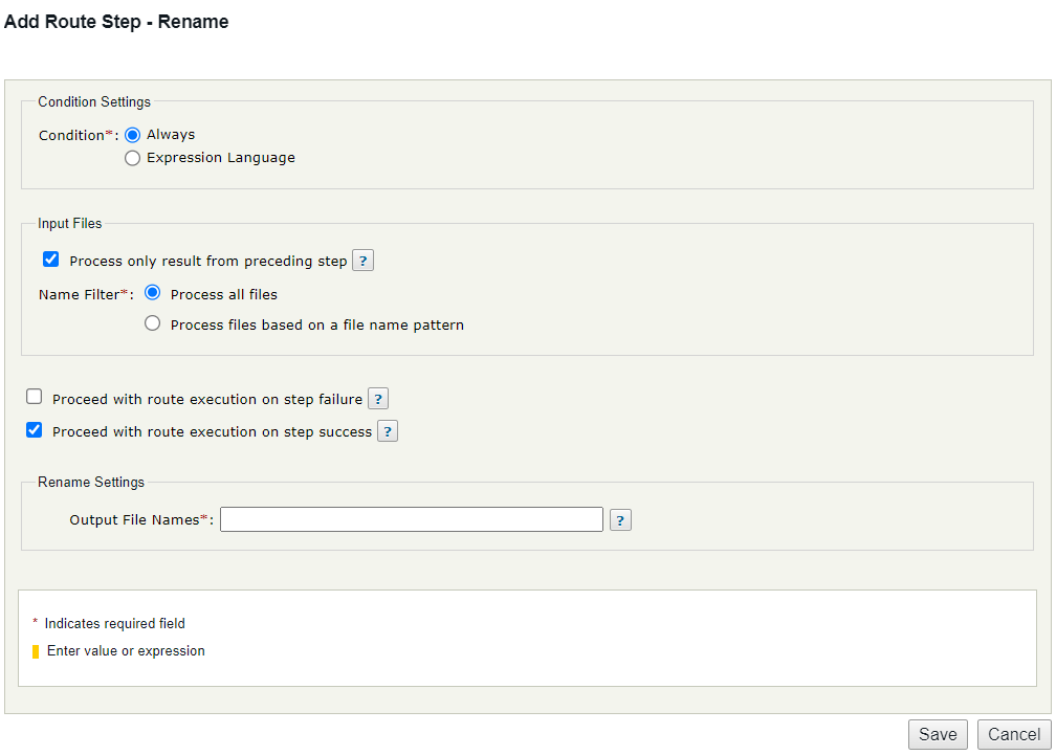
Rename

The Rename transformation step enables renaming of a designated file or files as part of a route. To add a Rename transformation step to a route package template, take the following steps:

1. Designate file filtering.
2. Determine whether to proceed with route execution on step failure and success.
3. Determine output file names.

Note Step 1 and 3 are mandatory. Step 2 is optional.

The following figure provides a configured example of a rename transformation step.



The following sections provide configuration details for the Rename transformation step:

- *Condition Settings on page 941*
- *Input Files on page 941*
- *Proceed with route execution on step failure or success on page 942*
- *Rename on page 940*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* settings consist of the selection *Process only result from preceding step* and determining the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **enabled** only files produced by the preceding step will be used as input for this step.

When *Process only result from preceding step* is **disabled** or this is the first step all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

The Name Filter can either be set to process all files forwarded by the selection of *Process only result from preceding step* or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.
If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.
If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is not selected by default.

  If not selected, the processing stops on the first failed transformation (if there are several files being transformed by the step). The route execution also stops.
- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful transformation (if there are several files being transformed by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Rename settings

The rename settings determine the naming of the output file or files.

Output file names

The output file or files are renamed according to the expression entered in the *Output File Names* field. All input files are renamed based on the configured expression.

Examples:

- New filename based on the current filename:

```
${basename(currentfulltarget)}.transformed
```

- New filename bases on the original filename with a timestamp:

```
${basename(currentfulltarget)}.${timestamp}.${extension
(currentfulltarget)}
```

The path of the new file (if any) will be stripped off and only the filename will be left.

Routing steps

This topic highlights essential details of using the routing steps - Publish To Account, Send To Partner, and Pull From Partner - independently and collectively within a route.

Consider the following key points when using routing steps:

- Each route containing transformation and routing steps must end with either Publish To Account or Send To Partner. These steps move the transformed files out of the sandbox folder, which is deleted when the route execution completes. For configuration instructions, see *Publish To Account* on page 943 and *Send To Partner* on page 948.
- Pull From Partner downloads files directly to the specified location. The resulting files can then be passed to another step for transformation or routing. For configuration instructions, see *Pull From Partner* on page 956.
- For routes consisting solely of Publish To Account and/or Send To Partner steps that do not involve any file transformation, SecureTransport is set by default via the `AdvancedRouting.DontCopyPayload` configuration option not to copy the Advanced Routing payload to a sandbox folder. Instead, it directly transmits the original files to speed up the route execution. However, it's important to note that any file operation within a route still necessitates the use of a sandbox folder, and it will be created in the following scenarios:
  - If there is a Post Routing Action Rename or Delete set for a step in the route.
    The Delete action is controlled by the Delete files after step is complete checkbox in the step configuration from the Administration Tool. The Rename action is configurable only through the REST API using the `postRoutingActionRenameExpression` parameter.
  - When Send Trigger File is enabled for a Send To Partner step
  - When the PeSIT Store And Forward Mode is enabled for a Send To Partner step

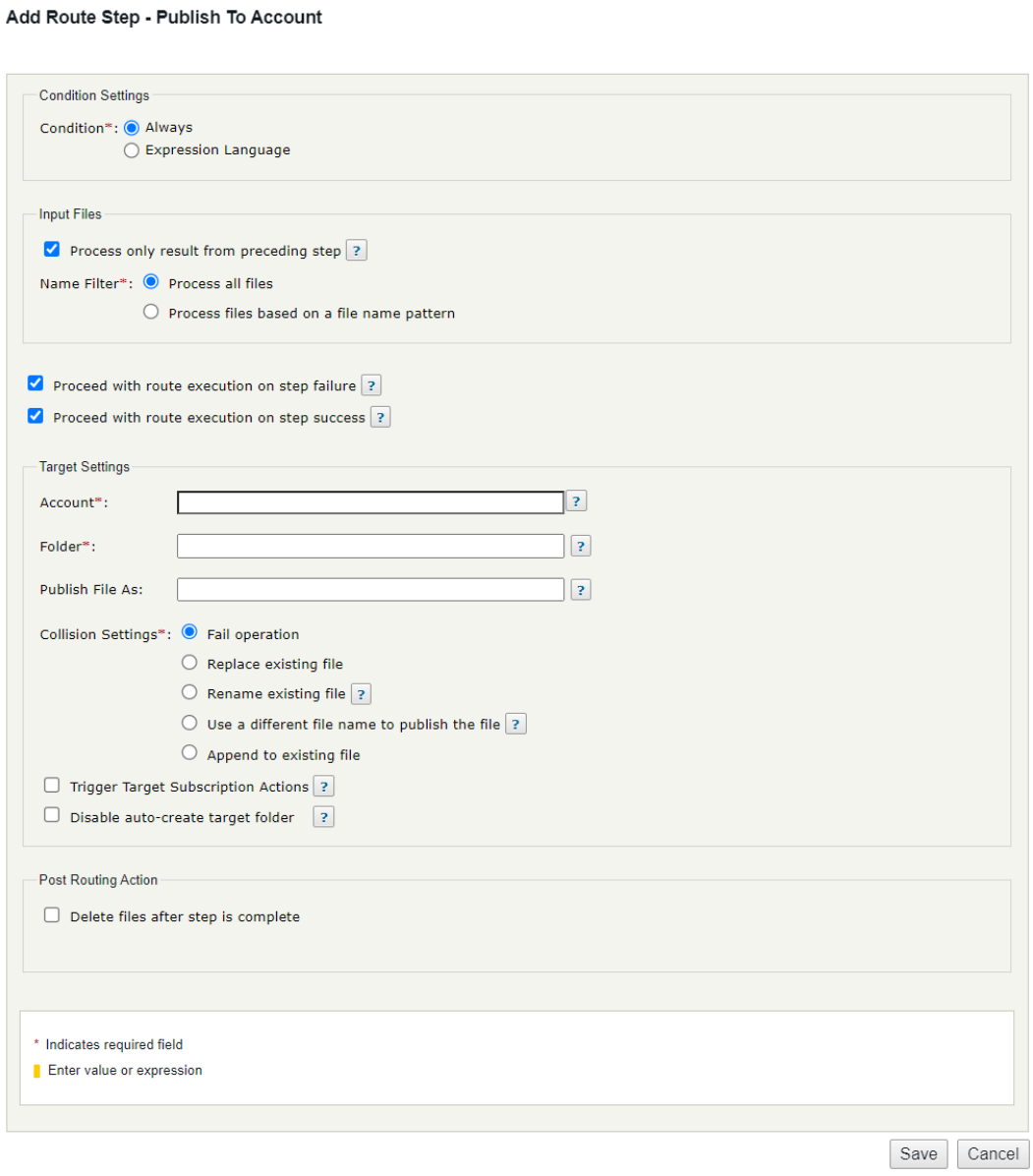
Publish To Account

The Publish To Account routing step enables publishing files to a specified account as part of a route. This step only publishes files to accounts that belong to the same SecureTransport Server. It cannot be used to publish files to accounts managed by other products or other instances of SecureTransport Server. To avoid this restriction, you could use a Send To Partner routing step. For more information, refer to *Send To Partner* on page 948.

To add a Publish To Account step to a route package template or a route package, take the following steps:

1. (Mandatory) Designate file filtering.
2. (Optional) Determine whether to proceed with route execution on step failure and success.
3. (Mandatory) Enter and select the target settings.
4. (Optional) Determine post routing actions.

The following figure provides an example of a Publish To Account routing step.



The following sections provide detailed Publish To Account route step configuration information:

- *Condition Settings on page 945*
- *Input Files on page 945*

- *Proceed with route execution on step failure or success on page 946*
- *Target Settings on page 946*
- *Post Routing Action on page 948*

Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The *Input Files* settings consist of the determining whether to *Process only result from preceding step* and setting the Name Filter.

Process only result from preceding step

When *Process only result from preceding step* is **enabled**, only files produced by the preceding step will be used as input for this step.

When *Process only result from preceding step* is **disabled** or this is the first step, all current working files will be used as input for this step.

Note The Name Filter settings will also be applied on a given set of input files.

Name Filter

The Name Filter can either be set to process all files forwarded by the selection of *Process only result from preceding step* or process files based on a designated filename pattern forwarded by the selection of *Process only result from preceding step*.
If *Process all files* is selected, all files forwarded by the selection of *Process only result from preceding step* are processed.
If *Process files based on a filename pattern* is selected, only files with names that match the pattern specified in the **File Name Pattern** field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops, depending on step success or failure.

- If **Proceed with route execution on step failure** is selected, route execution continues even if the step execution fails. This checkbox is selected by default.

  If not selected, the processing stops on the first failed file (if there are several files being published by the step). The route execution also stops.

- If **Proceed with route execution on step success** is selected, route execution continues if the step execution is successful. This checkbox is selected by default.

  If not selected, the processing stops after the last successful file (if there are several files being published by the step). The route execution also stops.

Note The two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Target Settings

Target settings consist of selecting the account and the folder location to which to publish the designated files. The settings also include determining the name of the published file and selecting how to handle collisions.

Account

This is the account to publish the file(s) to. You can specify either an account name or use an EL expression to determine the recipient based on the environment information (such as filename). If no account with such name exists, SecureTransport will try to match an account by its login name (either virtual or external user).

The *Account* field has auto-completion which shows a list of existing accounts containing the given search term.

Note Auto-completion will suggest account names and user (login) names.

Publish To Account could publish files to internal (virtual, unlicensed or service) users or external ones using an existing account template. In order to publish files to an external account (real, LDAP account, or SiteMinder), the administrator must specify the login name of this account (not the name of the account template which will be used).

Folder

This is the folder in the designated account to publish the file to. If the folder does not exist, it is automatically created. A folder name can be specified or an EL expression can be used to determine the folder based on the environment information. The folder name is a relative path to the home folder of the specified account.

Note A file will be successfully published to the specified folder only if the home folder of the designated account has a proper home folder access level. For more details, see *User accounts on page 501* and *Advanced Routing on page 864*.

Publish File As

This is the name of the published file. If this field is not empty, the file is published with the specified name. An EL script can also be used to specify a file name.

Examples:

- New file name based on the current file name (since the transformation might have change it):

  ```
  ${basename(currentfulltarget)}.sent
  ```

- New file name based on the original file name with a timestamp:

  ```
  ${basename(transfer.target)}.${timestamp}.${extension
  (transfer.target)}
  ```

- You can access the first value of a given SSO attribute with name `attributeName` with the expression `${sso.attributes['attributeName'][0]}`.

Note The path of the new file, if present, is removed, leaving only the file name.

Collision settings

Collision settings handle the cases of duplicate file names. These settings take into account:

- the name of the source file: the file that will be published by the Publish To Account step
- the name of the destination file: a file that already exists in the account's folder

Caution It is not recommended to publish multiple source files with the same name as the collision settings will not apply.

To determine the course of action in the event of duplicate file names, select one of the following settings:

- **Fail operation**: The step fails.
- **Replace existing file**: The destination file is replaced by the source file.
- **Rename existing file**: The name of the destination file is modified using the following pattern:

  If a source file with the name `myFile` is published, the destination file with the name `myFile` will be renamed to `myFile (old copy 1)`. If the latter already exists, the destination file will be renamed to `myFile (old copy 2)`, and so on.

- **Use a different file name to publish the file**: The name of the source file is modified using the following pattern:

  If a destination file with the name `myFile` exists, the source file with the name `myFile` will be published as `myFile (new copy 1)`. If the latter already exists, the source file will be

published as `myFile (new copy 2)`, and so on.

- **Append to existing file**: The contents of the source file are appended to the destination file.

Trigger target subscription actions

This option allows administrators to define the behavior if the target folder is a subscription or other special folder. When deselected (default), no further processing in the target folder is done. When selected, the post-processing actions defined in the target folder are executed.

Disable auto-create target folder

This option allows administrators to disable the automatic creation of a target folder. The checkbox is deselected by default. When it is selected and a target folder does not exist, it will not be created upon step execution and will result in step failure.

Post Routing Action

If *Delete files after step is complete* is selected, the files processed by the route step are deleted.

Note The output file names will be the same as the input file names. To change the file names use a Rename transformation step. To configure a Rename transformation step, refer to *Rename on page 940*.

Control File Visibility During Publishing

Depending on the transfer protocol used, files routed with the *Publish To Account* step could be visible to the receiver account while still being copied. Attempting to download an uploading file would result in an error, as you would either receive a partial file or one with zero bytes. In such cases, go to **Operations > Server Configuration** and check the value of the `ShowInprocessFiles` server configuration option. It controls whether a protocol client should display or hide files currently being uploaded to a listed folder. It is set to `true` by default, meaning files that are still uploading are visible.

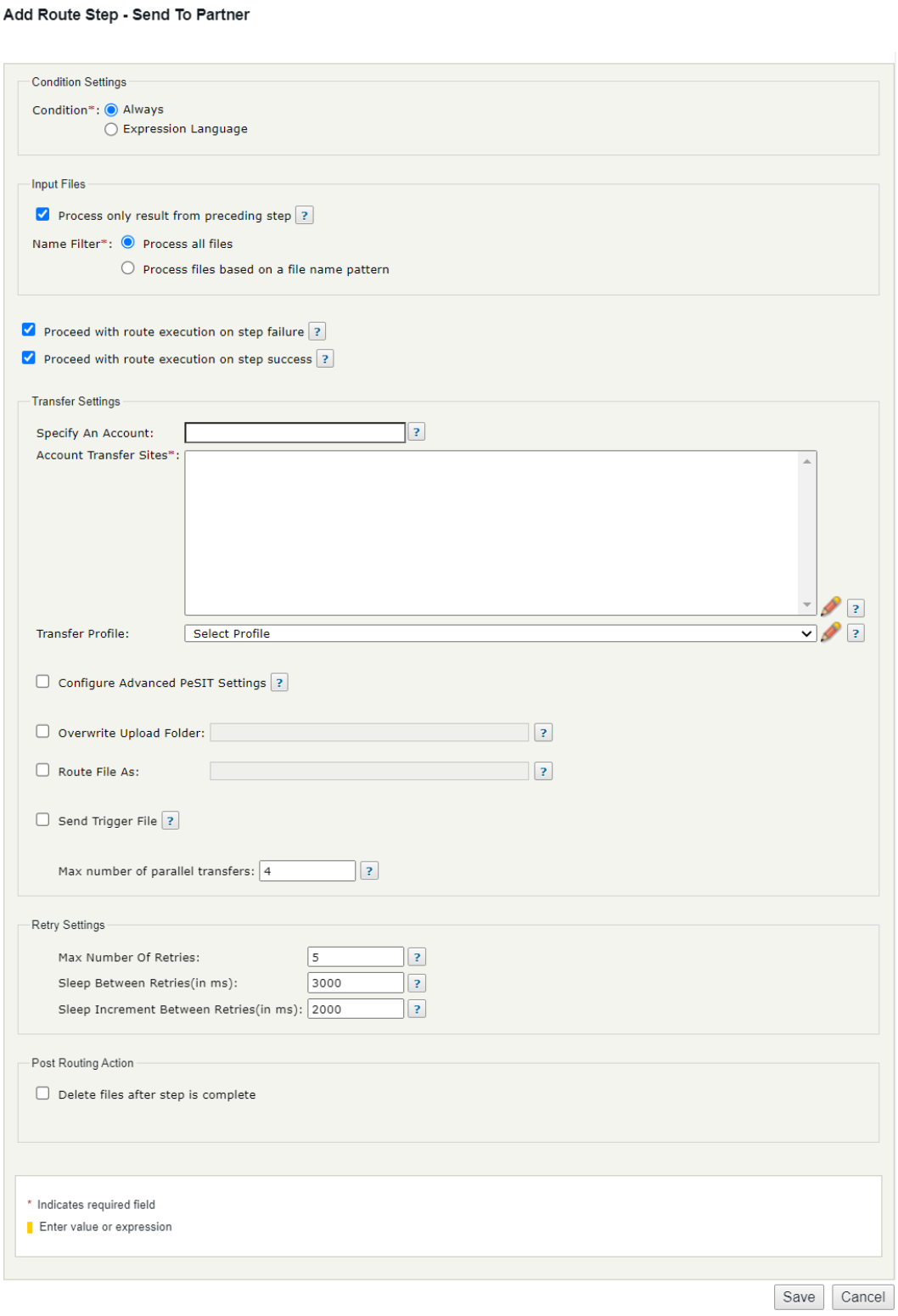
Send To Partner

The Send To Partner routing step enables routing to a specified partner account as part of a route. To add a Send To Partner step to a route package template or route package, take the following steps:

1. Define conditions for the execution of the step. See *Condition Settings on page 950*.
2. Specify the files to send. See *Input Files on page 950*.
3. Determine whether to proceed with route execution on step failure and success. See *Proceed with route execution on step failure or success on page 950*.
4. Configure the transfer settings. See *Transfer Settings on page 951*.

5. Configure retries. See *Retry settings on page 955*.
6. Set a post-routing action. See *Post Routing Action on page 956*.

The following figure provides an example of a Send To Partner routing step.



Condition Settings

Condition settings determine whether the step should be executed based on an evaluated condition. Select either Always or Expression Language.

- **Always** (default): The step is executed on every occasion.
- **Expression Language**: The step is executed only when the condition specified in the *Expression Language* field is met.
  Example: To execute the step only after a successful execution of the preceding one, enter the expression:
  `${routing.precedingStepExitStatus eq 'success'}`

Input Files

The settings under *Input Files* determine which files the step will process.

Process only result from preceding step

- When selected, only files produced by the preceding step are used as input.
- If not selected, or if this is the first step, all current working files are used as input.

Name Filter

The *Name Filter* allows you to further filter the files selected by the "Process only result from preceding step" option.

- *Process all files*: All files resulting from *Process only result from preceding step* are processed.
- *Process files based on a filename pattern*: Only files with names that match the pattern specified in the *File Name Pattern* field are processed. For more information, see *Globbing in SecureTransport* on page 1122 and *Regular expressions* on page 1117.

Proceed with route execution on step failure or success

These settings consist of determining whether route execution continues or stops based on step success or failure.

- **Proceed with route execution on step failure**: If selected (default state), route execution continues even if the step execution fails.

  If not selected, the processing stops on the first failed transfer (if several files are being transferred). The route execution also stops.
- **Proceed with route execution on step success**: If selected (default state), route execution continues if the step execution is successful.
  If not selected, the processing stops after the last successful transfer (if several files are being transferred). The route execution also stops.

These two checkboxes are independent. If neither is selected, the execution of the current route stops after this step.

Transfer Settings

The Transfer Settings consist of specifying an account, selecting account transfer sites, and selecting a transfer site profile.

Setting	Description
Specify An Account	Specify the account that holds information (transfer site) about the destination. You can use an account name, login name, or an EL expression based on the environment information (such as filename). This field supports auto-completion, showing a list of existing account and login names matching the search term. **Caution** Do not specify account template names in this field. The transfer sites that belong to the template will be populated but the step will fail. When using account templates, enter the login name instead.
Account Transfer Sites	You can specify transfer sites in two ways: by selecting existing sites from the chosen account or by using Expression language. • Choose from the list: Once an account is selected, its transfer sites are populated in the select box only if the account name matches the login name of the user. Only transfer sites with a proper access level will be listed. For more details see *Transfer sites on page 540*. If the account name is unknown (for example, expression-based) or the login name does not match the account name, its transfer sites are determined at runtime. • Use an EL expression: Click and enter the expression in the field. Expression language and wildcard symbols ('*' and '?') are supported.
Transfer Profile	This setting is used only if the transfer site is of type PeSIT. Otherwise, it is ignored. You can either select from the listed profiles or use an EL expression. To use an expression, click and enter the expression in the field. When the transfer profile is specified via EL expression, there are three possible cases: • EL expression does not match any account transfer profiles - the default transfer profile is used then. • EL expression matches more than one transfer profile - the default transfer profile is used. • EL expression matches exactly one transfer profile - the matched transfer profile is used.

Configure advanced PeSIT settings

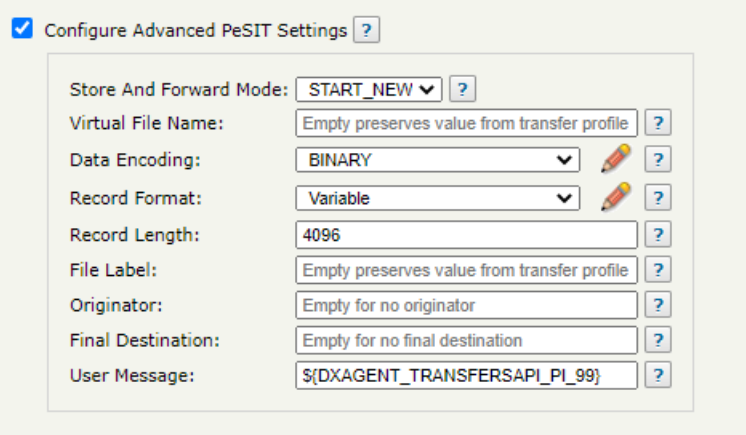
Select the *Configure Advanced PeSIT Settings* checkbox to display additional settings for PeSIT transfers.

Setting	Description
Store And Forward Mode	The Store and Forward mode selections are: • START_NEW: initiates a new store and forward transfer and the current transfer (if any) gets acknowledged. • PRESERVE: preserves the current store and forward transfer (if any). The transfer will fail if the PeSIT transfer site is used for sending files that were received via a protocol other than PeSIT. Automatic acknowledgment is not supported in PRESERVE mode, however, manual acknowledgment can still be sent.
Virtual File Name	Use this option to overwrite the virtual file name (PI12) predefined in the transfer profile. To preserve the predefined name, either leave the field empty or enter the expression `${pesit.file.filename}`. This configuration parameter corresponds to the **IDF** parameter in Axway Transfer CFT.
Data Encoding	Use this option to overwrite the data encoding (PI16) defined in the transfer profile. • If you want to overwrite it, either select from the options (ASCII, EBCDIC, BINARY, EBCDIC_NATIVE) or click **Edit** () and enter an EL expression in the field. • To preserve the data encoding specified in the transfer profile, do not make a selection or use the EL expression `${pesit.pi.dataEncoding}`. This configuration parameter corresponds to the **FCODE** parameter in Axway Transfer CFT.
Record Format	Choose between Variable and Fixed. In *START_NEW* transfer mode, the **Record format** field is used to overwrite the record format (PI31) predefined in the transfer profile. In *PRESERVE* transfer mode, the **Record format** field defines PI31. It can be an empty value, EL expression `${pesit.pi.recordFormat}`, or a numeric value. When the **Record format** field is empty and the file was originally received over PeSIT, the Record format (PI31) is preserved from the original file transfer. If the file is received over a different protocol, the Record format (PI31) is preserved from the transfer profile. This configuration parameter corresponds to the **FRECFM** parameter in Axway Transfer CFT.
Record Length	In *START_NEW* transfer mode, the **Record length** field is used to overwrite the record format (PI32) predefined in the transfer profile. In *PRESERVE* transfer mode, the **Record length** field defines PI32. It can be an empty value, EL expression `${pesit.pi.recordLength}`, or a numeric value. When the **Record length** field is empty and the file was originally received over PeSIT, the Record length (PI32) is preserved from the original file transfer. If the file is received over a different protocol, the Record length (PI32) is preserved from the transfer profile. This configuration parameter corresponds to the **FLRECL** parameter in Axway Transfer CFT.

Setting	Description
File Label	Used this option to overwrite the file label (PI37) predefined in the transfer profile. To preserve the predefined file label, either leave the field empty or enter the EL expression `${pesit.pi.fileLabel}` in the field. This configuration parameter corresponds to the **NFNAME** parameter in Axway Transfer CFT.
Originator	This field is used to overwrite the original sender (PI61) of the transfer. To preserve the originator of the previous transfer, enter the EL expression `${pesit.pi.originalSenderID}` in the field. To make a Store and Forward PeSIT transfer, specify the originator and choose the intermediate partner (**IPART** parameter in Axway Transfer CFT) in the transfer site list. **Note** The *Originator* can only be changed when the *Store and Forward* mode is set to `START_NEW`.
Final Destination	This field is used to overwrite the final destination (PI62) of the transfer. To preserve the final destination of the previous transfer, enter the EL expression `${pesit.pi.finalDestinationID}` in the field. To make a Store and Forward PeSIT transfer specify the final destination and choose the intermediate partner (**IPART** parameter in Axway Transfer CFT) in the transfer site list. **Note** *Final Destination* can only be changed when the *Store and Forward* mode is set to `START_NEW`.
User Message	This field is used to overwrite the user message (PI99) for outgoing files defined in the PeSIT transfer site (the "User Message Send" field). It accepts ASCII text and EL expressions. • To preserve the user message defined in the transfer site, in the Send To Partner step, either leave the *User message* field empty or enter the expression: `${pesit.pi.serviceParam}`. • The user message defined in the transfer site can be overwritten from REST API transfer requests using custom properties. In the step's *User message* field, enter the expression `{DXAGENT_TRANSFERSAPI_*}`. Check the usage example below. This configuration parameter corresponds to the **PARM** parameter in Axway Transfer CFT.

Example: User Message configuration

1. In the **User Message** field of the step, enter the expression:
   `${DXAGENT_TRANSFERSAPI_*}`
   where * is an arbitrary name, for example, "PI_99"



2. To trigger the transfer and specify the value of the User Message, you need to set the value of the dynamic property in the request body in the form:
"YourCustomPropertyName":"value"

   For example, "PI_99": "User Message sent via Sent to Partner Step".

```
{
"accountName": "YourAccount",
"site": "YourSite",
"destinationDirectory": "DirectoryName",
"customProperties": {
  "PI_99": "User Message sent via Sent to Partner Step",
  }
}
```

   This way, when executing transfers via the REST API and specifying a custom value for User Message, the value is preserved and passed to subsequent PeSIT transfer, triggered by the Send To Partner step.

Overwrite upload folder

Use this option to overwrite the upload folder specified in the transfer site definition (if allowed in the transfer site). The new folder must be located on the target file system; otherwise, the sending of the trigger file could fail. The field supports Expression Language.

Note The upload folder of a custom Pluggable Transfer Site cannot be overwritten.

Route files as

Use this field to change the names of files being routed. If *Route files as* is selected, the entry in the field overwrites *Send File As* property set in the transfer site. The field supports expression language.

Send trigger file

If the `Send Trigger File` option is selected, a designated trigger file is sent to one or more transfer sites after the successful routing of a file or files. The trigger file exists only in the sandbox folder. It is deleted immediately after it is successfully transferred to the selected destination(s).

1. Specify the name of the trigger file. You can use an EL expression.

   Note The *Send File As* option in the transfer site definition does not affect the trigger file name; it will be named as defined here. However, the other *Post Transmission Send Options* configured in the transfer site apply to the transferred trigger file.

2. Use the **Send trigger file for each transferred file** checkbox to determine trigger frequency:
   - If selected, SecureTransport sends a trigger file after each successful routing of a data file to the transfer site.
   - If not selected, one trigger file is sent after all transformed file are successfully routed to the transfer site. If the sending of one or more files fails, the trigger file is not sent.
3. In the **Trigger file content** box, define the content of the trigger file. Expression language is supported. `\n\r` (CRLF) is used as a line separator. If the box is left empty, an empty (zero byte) trigger file is sent.
4. Select the trigger file destinations by specifying a destination account, choosing account transfer sites, selecting the transfer profile (if applicable), and deciding whether to overwrite the upload folder. For detailed descriptions of these settings, see *Transfer settings*.

Max number of parallel transfers

The entry in the *Max number of parallel transfers* field determines the maximum number of parallel transfer connections for each transferred file. The default number of maximum parallel transfers is 4, which means you cannot have more than 4 concurrent transfer connections at a time for a given file. Note that files are published in bulk to the configured transfer sites, and are processed according to the number of *Maximum parallel transfers* specified for each site.

No retries are triggered if the reason for the failure is permanent (for example, the wrong credentials are specified in the transfer site being used).

Note *Max number of parallel transfers* and *Retry settings* are applied for both, the data file and the trigger file.

Retry settings

The retry settings include setting the maximum number of retries, setting the sleep time between retries, and setting the sleep increment between retries. Those settings are used for both the data file and trigger file, if configured.

Setting	Description
Max Number of Retries	Number of times the transfer needs to be retried in case of error. Default: 5.
Sleep Between Retries	Amount of time to wait in milliseconds before making the next retry attempt. Default: 3000 milliseconds.
Sleep Increment Between Retries	Incremental interval in milliseconds. SecureTransport waits for the interval specified in the **Sleep Between Retries** field before the first retry and then exponentially increases the time between each subsequent retry. Default: 2000 milliseconds.

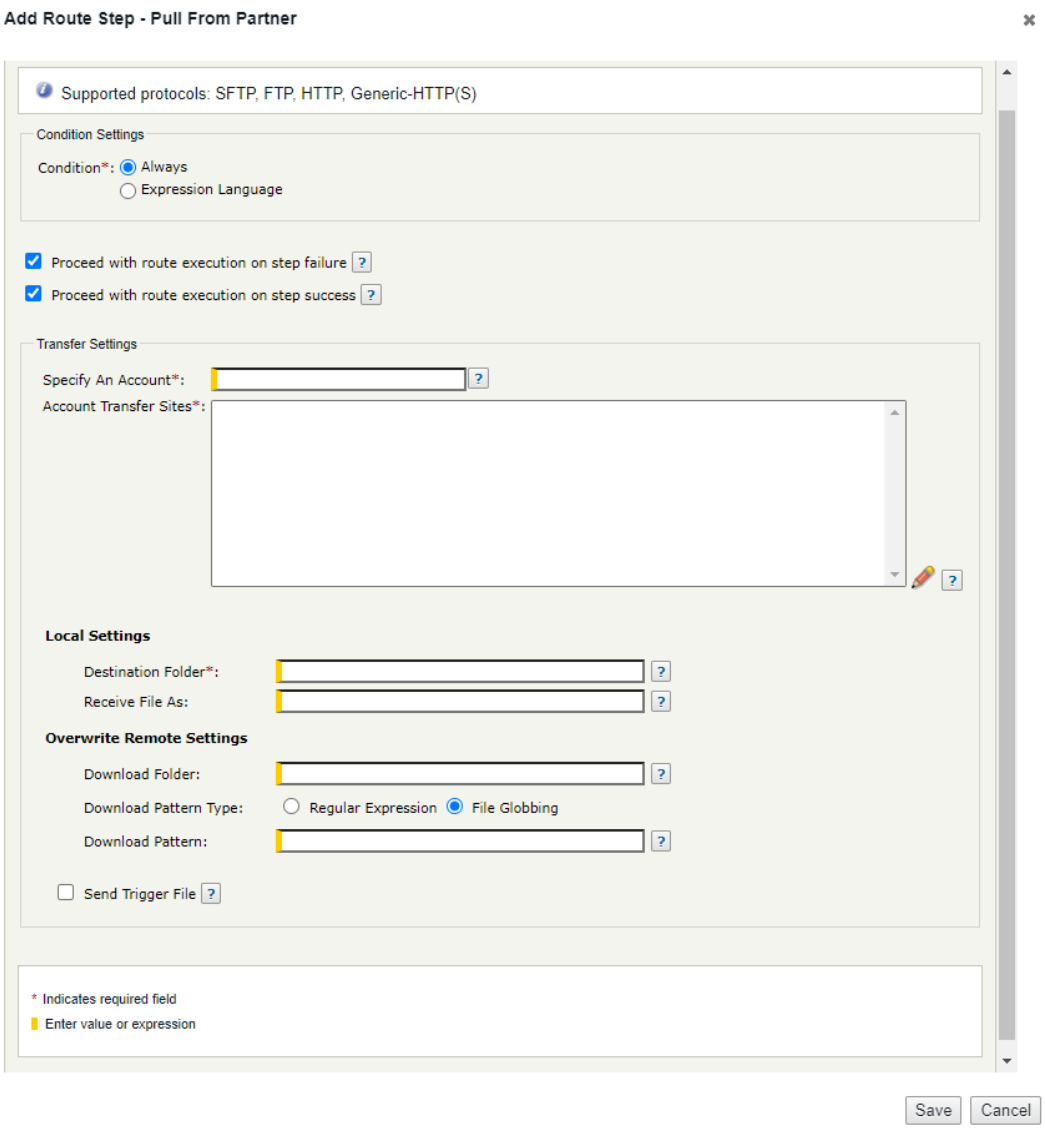
Post Routing Action

The post-routing actions occur once the route step is completed. With the *Send To Partner* step, the only supported action is file deletion.
If *Delete files after step is complete* is selected, the files processed by the route step are deleted from the sandbox folder. Post-routing actions are not performed on the trigger file.

Pull From Partner

This step enables SecureTransport to pull files from an FTP, HTTP, Generic-HTTP(S), or SSH transfer site as part of a route. Unlike the other two routing steps (Send To Partner and Publish To Account), Pull From Partner is intended to provide input files for another step.

The files downloaded with this step do not reach the SecureTransport's sandbox folder because the pull operation (including pull via the RestAPI) is asynchronous. Therefore, in order to route a file that has been downloaded via Pull From Partner to another target via the Send To Partner step, the file should be downloaded in the subscription folder that triggers a route that contains a Send To Partner step. You can see an example configuration here.



To configure a Pull From Partner step, follow the instructions:

1. At the *New Route entry* page, from the **Select Step** drop-down, select **Pull From Partner**.
2. Click **+ Add Step**.

   The *Add Route Step - Pull From Partner* configuration window opens.
3. In the **Condition Settings** section, specify when the step to be executed:
   - Select **Always** if you want this step to be executed every time the route is triggered.
   - Select **Expression Language** to use a custom expression to define a trigger condition.
4. Use the **Proceed with route execution on step failure** checkbox to specify whether or not to continue route execution if the step fails. The step is considered failed when SecureTransport fails to submit the pull request. If the checkbox is not selected, SecureTransport terminates the route when the pull request submission fails.

5. Use the **Proceed with route execution on step success** checkbox to specify whether or not to continue route execution if the step completes successfully. If the checkbox is not selected, SecureTransport terminates the route after successfully submitting the pull requests to the event queue.
6. In the **Specify An Account** field, specify the partner account that holds the transfer site you want to pull files from. You may enter an account name or login name, or use an expression.

   SecureTransport checks the account information:

Check	Result
If the login and the account name match	All supported transfer sites - FTP(S), HTTP(S), Generic-HTTP(S), and SSH - under the specified account appear in the **Account Transfer Sites** field.
If the account name doesn't match the login name or the account is specified using an EL expression	The transfer site to use is determined at runtime.

7. In the **Account Transfer Sites**, specify the transfer site that contains the files you want to pull in one of the following ways:
   - Select from the list of the account's transfer sites.
   - Click the **Edit** button, and enter a name pattern for the transfer site. The field supports expression language, regular expressions and the wildcard symbols '*' and '?'.

     Note Only transfer sites with proper access level will be taken into account. For more details, see Access level.
8. Mandatory: Under **Local Settings**, in the **Destination** field, enter the path of a local folder for the downloaded files.
9. In the **Receive File As** field, type a file name pattern for downloaded files. The field supports expression language.
   If it is left empty, SecureTransport will use the pattern that is defined in the transfer site's **Receive File As** field (in the Receive Options tab). If both **Receive File As** fields are empty, the remote filename (target) will be kept.

   Note The EL expression that refers the remote (target) filename, `${stenv ['target']}`, is not yet supported for this field. The workaround is to use such expression in the transfer site's **Receive File As** field and leave the respective field in Pull From Partner empty.
10. If you want to pull files based on the transfer site configuration, leave the fields under **Overwrite Remote Settings** section empty. Otherwise, provide the following information:

Field	Description

Download Folder	Enter the path of the folder to pull files from. Expression Language is supported. If left empty, the transfer site's Download Folder is used.
Download Pattern Type	Select the file name pattern syntax: Regular Expression or File Globbing
Download Pattern	Enter a name pattern using the selected syntax to identify the files to be downloaded. If this field is left empty, SecureTransport will use the transfer site's Download Pattern.

11. Use the **Send Trigger File On Failure** option to determine what happens with the AR processing if the pull fails or returns no files. For more information, see *Send Trigger File on Failure* on page 959.
12. Click **Save**.

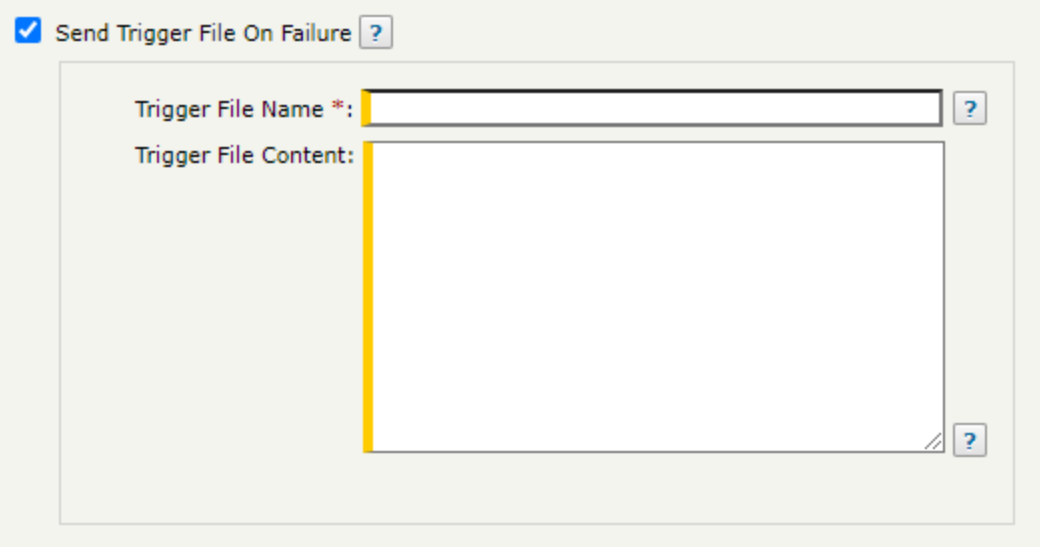
Send Trigger File on Failure

When *Send Trigger File on Failure* is enabled, a designated trigger file is sent to the Advanced Routing sandbox folder when a pull fails or no files were returned by the remote partner (wildcard pull). This file is intended to trigger further Advanced Routing processes and serve as input for subsequent steps. It will be deleted from the sandbox folder when the route execution completes.

Setup

The *Send Trigger File on Failure* functionality is disabled by default. Follow the steps below to enable it and define the trigger file:

1. Select **Send Trigger File On Failure** checkbox.



2. In the **Trigger File Name** field, specify a name for the trigger file. Expression language is supported.

3. In the **Trigger File Content** field, specify the content of the trigger file. Expression language is supported.
   If this field is left empty, an empty (zero byte) trigger file will be sent.

Limitation: *Send Trigger File on Failure* is not compatible with the *ZeroByteWildcardPullAllowed* server configuration option. When the latter is set to `true`, further advanced routing processing will not be triggered.

Retry Mechanism

The Pull From Partner step relies on the global retry mechanism, dictated by the following server configuration options:

- `EventQueue.maxRetryCount`
- `EventQueue.retryDelayInterval`
- `EventQueue.internalRetryDelayInterval`

The retry settings include setting the maximum number of retries, sleep time between retries, and sleep increment between retries. For more information, see Configure retry parameters for server-initiated transfers.

Usage Examples

Refer to the "Advanced use cases" section in the Administration Guide for use case examples that can help you create complex flows using the Pull From Partner step:

- *Notify external messaging systems (JMS) of pull failures* on page 988
- *Route files downloaded with Pull From Partner to an external target* on page 985
- *Send a trigger file via SFTP on pull failures* on page 990
- *Use JMS metadata values for Pull From Partner step* on page 985

Advanced Routing scenarios: configuration examples

This topic describes basic and complex use cases. Use cases provide customer oriented examples of Advanced Routing configurations.

The following basic use cases are described:

- PGP Decryption and Publish To Account
- Line Ending and Publish To Account
- Send To Partner
- Compress and Send To Partner
- Decompress and Publish To Account

- External Script and Send To Partner
- Send To Partner (PeSIT)

The advanced use cases are described:

- Route files based on file name extension
- PGP Decryption, PGP Encryption (partner's certificate), and send to multiple partners
- Decompress and Send to Partner (Trigger File Output)

The following topics describes the basic and advanced use cases:

- *Basic use cases* on page 961 - Describes the basic use cases.
- *Advanced use cases* on page 975 - Describes the advanced use cases.

Basic use cases

An overview, prerequisites, flow configuration steps, and flow of events of the following basic use cases are provided:

- *PGP Decryption and Publish To Account* on page 961 - Provides the overview, prerequisites, flow configuration steps, and flow of events for the PGP Decryption and Publish To Account use case.
- *Line Ending and Publish To Account* on page 963 - Provides the overview, prerequisites, flow configuration steps, and flow of events for the Line Ending and Publish To Account use case.
- *PGP Encryption and Send To Partner* on page 965 - Provides the overview, prerequisites, flow configuration steps, and flow of events for the PGP Encryption and Send To Partner use case.
- *Compress and Send To Partner* on page 967 - Provides the overview, prerequisites, flow configuration steps, and flow of events for the Compress and Send To Partner use case.
- *Decompress and Publish To Account* on page 969 - Provides the overview, prerequisites, flow configuration steps, and flow of events for the Decompress and Publish To Account use case.
- *External Script and Publish To Account* on page 970 - Provides the overview, prerequisites, flow configuration steps, and flow of events for the External Script and Publish To Account use case.
- *Send To Partner (PeSIT)* on page 973 - Provides the overview, prerequisites, flow configuration steps, and flow of events for the Send To Partner (PeSIT) use case.

PGP Decryption and Publish To Account

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the PGP Decryption and Publish To Account use case:

- *Overview* on page 962
- *Prerequisites* on page 962
- *Steps to configure the flow* on page 962
- *Flow of events* on page 963

Overview

PGP decrypt each incoming file and publish the files to the local account.

Prerequisites

- Create a Route Package Template. For instructions on creating a Route Package Template, refer to *Add Route Package Template* on page 876.
- Create an Advanced Routing application instance. For instructions on creating an Advanced Routing application instance, refer to .
- Create an user account in SecureTransport. For instructions on creating an user account, refer to *User accounts* on page 501.
- Generate or import a private PGP key which is used for decryption. For instructions on generating or importing a private PGP key, refer to *Manage login certificates* on page 527or *Manage login certificates* on page 527.

Steps to configure the flow

1. Create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.
    a. Configure the subscription folder.
    b. (Optional) configure the rest of the settings.
    c. Click **Add** when done.
2. Navigate to the *Routes* tab of the created account and assign a new route package to the account by choosing the created Route Package Template and clicking the **Assign Route** button. For assigning a route configuration details, refer to *Assign Route Package Template* on page 887.
3. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 1.
4. Create a new route by clicking the **New Route** button in the *Specific Settings* pane. For route configuration details, refer to *New Route* on page 881.
    a. Configure the new route's name and (optionally) description.
    b. Add and configure a PGP Decryption step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For PGP Decryption configuration details, refer to *PGP Decryption* on page 901.
        i. (Optional) Enable **Require Encryption** and/or **Require Trusted Signature**.
        ii. Click **Save** when done.

        Note The PGP Decryption step automatically detects the PGP private key for decrypting the content. The step only searches in the account key store.

c. Add and configure a Publish to Account step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For Publish To Account configuration details, refer to *Publish To Account* on page 943.

   i. Uncheck **Proceed with route execution on step failure**.

   ii. Select an account to publish to (for example, the current account).

   iii. Select a folder to publish the file to (for example, the subscription folder configured in Step 1).

   iv. (Optional) Configure the rest of the settings.

   v. Click **Save** when done.

5. Save the route and the route package.

Flow of events

1. A PGP encrypted file is uploaded via any protocol to the Advanced Routing subscription folder.
2. The Advanced Routing application triggers the route.
3. The uploaded file is decrypted and published to the specified folder.

Note Upon completion of the route there will be two files present – the PGP encrypted (original) file and the PGP decrypted file. To automatically remove the PGP encrypted file, select **Post Processing Action > On Success > Delete** in the subscription settings.

Note Publishing the PGP decrypted file in Advanced Routing subscription folder does not trigger the route execution once again, because *Trigger target subscription actions* is **unchecked**.

Line Ending and Publish To Account

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the Line Ending and Publish To Account use case:

Overview

Transform end-of-line characters of each incoming file and publish the file to the local account.

Prerequisites

- Create a Route Package Template. For Route Package Template creation details, refer to *Add Route Package Template* on page 876.
- Create an Advanced Routing application instance. For Advanced Routing application instance creation details, refer to *Create Advanced Routing application* on page 874.

- Create a SecureTransport user account. For user account creation details, refer to *User accounts* on page 501.

Step to configure the flow

1. Create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.
    a. Configure the subscription folder.
    b. (Optional) Configure the rest of the settings.
    c. Click **Add** when done.
2. Navigate to the *Routes* tab of the created account and assign a new route package to the account by choosing the created Route Package Template and clicking the **Assign Route** button. For assigning a route configuration details, refer to *Assign Route Package Template* on page 887.
3. Assign the subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 1.
4. Create a new route by clicking the **New Route** button in the *Specific Settings* pane. For route configuration details, refer to *New Route* on page 881.
    a. Configure the new route's name and (optionally) description.
    b. Add and configure a Line Ending step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For Line Ending configuration details, refer to *Line Ending* on page 912.
        i. Select the source line ending format (for example, **Custom -> \u0025** for Mainframe end-of-line characters).
        ii. Select source file encoding (for example, **X-ORACLE-WE8EBCDIC500**).
        iii. Select target line ending format (for example, **Linux(LF)**).
        iv. Select target file encoding (for example, **US-ASCII**).
        v. (Optional) Configure the rest of the options.
        vi. Click **Save** when done.
    c. Add and configure a Publish to Account step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For Publish To Account configuration details, refer to *Publish To Account* on page 943.
        i. Uncheck **Proceed with route execution on step failure**.
        ii. Select an account to publish to (for example, the current account).
        iii. Select a folder to publish the file to (for example, the subscription folder configured in Step 1).
        iv. Enter the following expression in the *Publish file as* field:

```
${basename(transfer.target)}.us-ascii${extension
(transfer.target)}
```

This expression transforms the original file name by adding `.us-ascii` before the filename extension. For example, for file `incoming.txt` the result is `incoming.us-ascii.txt`.

v. (Optional) Configure the rest of the settings.

vi. Click **Save** when done.

5. Save the route and the route package.

Flow of events

1. A file with **\u0025** end-of-line character and **WE8EBCDIC500** file encoding is uploaded via any protocol to the Advanced Routing subscription folder.
2. The Advanced Routing application triggers the route.
3. The uploaded file's end-of-line character is changed to **LF** and the file's encoding is changed to **US-ASCII**.

Note In the end there are two files present – the incoming (original) file and transformed file. To automatically remove the incoming file, select **Post Processing Action -> On Success -> Delete** in the subscription settings.

Note Publishing the transcoded file in Advanced Routing subscription folder does not trigger the route execution once again, because *Trigger target subscription actions* is **unchecked**.

PGP Encryption and Send To Partner

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the PGP Encryption and Send To Partner use case:

Overview

PGP encrypt each incoming file and route the file to a remote transfer site.

Prerequisites

- Create a Route Package Template. For Route Package Template creation details, refer to *Add Route Package Template* on page 876.
- Create an Advanced Routing application instance. For Advanced Routing application instance creation details, refer to *Create Advanced Routing application* on page 874.

- Create SecureTransport user account. For user account creation details, refer to *User accounts* on page 501.
- Create a remote transfer site. For remote transfer site creation details, refer to *Create a transfer site* on page 628.
- Generate or import a Partner PGP key which is used for encryption. For instructions on generating or importing a partner PGP key, refer to *Manage login certificates* on page 527or *Manage login certificates* on page 527.

Step to configure the flow

1. Create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.
    a. Configure the subscription folder.
    b. (Optional) Configure the rest of the settings.
    c. Click **Add** when done.
2. Navigate to the *Routes* menu of the created account and assign a new route package to the account by choosing the created Route Package Template and clicking the **Assign Route** button. For assigning a route configuration details, refer to *Assign Route Package Template* on page 887.
3. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 1.
4. Create a new route by clicking the **New Route** button in the *Specific Settings* pane. For route configuration details, refer to *New Route* on page 881.
    a. Configure the new route's name and (optionally) description.
    b. Add and configure a PGP Encryption step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For PGP Encryption configuration details, refer to *PGP Encryption* on page 896.
        i. Select the **Encrypt only** option.
        ii. Configure an account from which to select PGP certificate for encryption (for example, the current account).
        iii. Select a PGP certificate for encryption from that account. PGP certificate can be specified by alias, EL or wild card symbols. If more that one certificate matches the pattern, the first one is used.
        iv. Click **Save** when done.
        a. Add and configure a Send To Partner step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For Send To Partner configuration details, refer to *Send To Partner* on page 948.

i. Uncheck **Proceed with route execution on step failure**.

ii. Select the account which contains the target transfer site (or select **Use current account**).

iii. Select the transfer site from the selected account to send the file to.

iv. Click **Save** when done.

5. Save the route and the route package.

Flow of events

1. A file is uploaded via any protocol to the Advanced Routing subscription folder.
2. The Advanced Routing application triggers route.
3. The uploaded file is encrypted and sent to the remote transfer site.

Compress and Send To Partner

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the Compress and Send To Partner use case:

Overview

Compress multiple incoming files and route the archive to a remote transfer site.

Prerequisites

- Create a Route Package Template. For Route Package Template creation details, refer to *Add Route Package Template* on page 876.
- Create an Advanced Routing application instance. For Advanced Routing application instance creation details, refer to *Create Advanced Routing application* on page 874.
- Create a SecureTransport user account. For user account creation details, refer to *User accounts* on page 501.
- Create a remote transfer site. For remote transfer site creation details, refer to *Create a transfer site* on page 628.

Steps to configure the flow

1. Create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.
    a. Configure the subscription folder.
    b. To await multiple files, configure a condition (for example, trigger file) on which to submit the files to the route.
    c. (Optional) Configure the rest of the settings.
    d. Click **Add** when done.
2. Navigate to the *Routes* tab of the created account and assign a new route package to the account by choosing the created Route Package Template and clicking the **Assign Route** button. For assigning a route configuration details, refer to *Assign Route Package Template* on page 887.
3. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 1.
4. Create a new route by clicking the **New Route** button in the *Specific Settings* pane. For route configuration details, refer to *New Route* on page 881.
    a. Configure the new route's name and (optionally) description.
    b. Add and configure a Compress step by selecting it from the *-- Select Step --* drop-down menu and clicking the **Add Step** button. For Compress configuration details, refer to *Compress* on page 904.
        i. Choose name for the archive (for example, `archive-${timestamp}.zip`).
        ii. (Optional) Configure the rest of the settings.
        iii. Click **Save** when done.
    c. Add and configure a Send To Partner step by selecting it from the *-- Select Step --* drop-down menu and clicking the **Add Step** button. For Send To Partner configuration details, refer to *Send To Partner* on page 948.
        i. Uncheck **Proceed with route execution on step failure**.
        ii. Select the account which contains the target transfer site (or select **Use current account**).
        iii. Select the transfer site from the selected account to send the file to.
        iv. Click **Save** when done.
5. Save the route and the route package.

Flow of events

1. Multiple files are uploaded via any protocol to the Advanced Routing subscription folder.
2. The trigger file (file with `.trigger` extension) is uploaded to the subscription folder.

3. The Advanced Routing application triggers the route.
4. The uploaded files are compressed into single zip archive and sent to the remote transfer site.

Decompress and Publish To Account

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the Decompress and Publish To Account use case:

Overview

Decompress incoming archives and publish the result files to the local account.

Prerequisites

- Create a Route Package Template. For instructions on creating a Route Package Template, refer to *Add Route Package Template* on page 876.
- Create an Advanced Routing application instance. For instructions on creating an Advanced Routing application instance, refer to .
- Create a SecureTransport user account. For instructions on creating an user account, refer to *User accounts* on page 501.

Steps to configure the flow

1. Create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.
    a. Configure the subscription folder.
    b. (Optional) Configure the rest of the settings.
    c. Click **Add** when done.
2. Navigate to the *Routes* tab of the created account and assign a new route package to the account by choosing the created Route Package Template and clicking the **Assign Route** button. For assigning a route configuration details, refer to *Assign Route Package Template* on page 887.
3. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 1.

4. Create a new route by clicking the **New Route** button in the *Specific Settings* pane. For route configuration details, refer to *New Route* on page 881.
    a. Configure the new route's name and (optionally) description.
    b. Add and configure a Decompress step by selecting it from the *-- Select Step --* drop-down menu and clicking the **Add Step** button. For Decompress configuration details, refer to *Decompress* on page 908.
        i. (Optional) Configure the available options.
        ii. Click **Save** when done.

            Note The Decompress step automatically detects the archive type.

    c. Add and configure a Publish to Account step by selecting it from the *-- Select Step --* drop-down menu and clicking the **Add Step** button. For Publish To Account configuration details, refer to *Publish To Account* on page 943.
        i. Uncheck **Proceed with route execution on step failure**.
        ii. Select an account to publish to (for example, the current account).
        iii. Select a folder to publish the file to (for example, the subscription folder configured in Step 1).
        iv. (Optional) Configure the rest of the settings.
        v. Click **Save** when done.
5. Save the route and the route package.

Flow of events

1. An archive is uploaded via any protocol to the Advanced Routing subscription folder.
2. The Advanced Routing application triggers the route.
3. The uploaded archive is decompressed and published to the specified folder.

Note In the end the subscription folder contains the archive file and the decompressed files. To automatically remove the archive file, select **Post Processing Action > On Success > Delete** in the subscription settings.

Note Publishing the decompressed files in Advanced Routing subscription folder does not trigger the route execution once again, because *Trigger target subscription actions* is **unchecked**.

External Script and Publish To Account

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the External Script and Publish To Account use case:

Overview

Compress incoming files by leveraging an external script and publish the result archive to the local account.

Prerequisites

- Download and install 7zip on your SecureTransport machine.
- Create a Route Package Template. For instructions on creating a Route Package Template, refer to *Add Route Package Template* on page 876.
- Create an Advanced Routing application instance. For instructions on creating an Advanced Routing application instance, refer to .
- Create a SecureTransport user account. For instructions on creating an user account, refer to *User accounts* on page 501.

Step to configure the flow

1. Create a script with the following contents that uses 7zip to compress the incoming files.
    a. Modify the third line of the script and set the correct path to the *7z* executable.
    b. Name the script `7zip-compress.sh`.
    c. Ensure the script is accessible by SecureTransport. For this example, it needs to be deployed in the `/bin/agents` subfolder of the SecureTransport installation folder.
2. Create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.
    a. Configure the subscription folder.
    b. To await multiple files, configure a condition (for example: trigger file) on which to submit the files to the route.
    c. (Optional) Configure the rest of the settings.
    d. Click **Add** when done.
3. Navigate to the *Routes* tab of the created account and assign a new route package to the account by choosing the created Route Package Template and clicking the **Assign Route** button. For assigning a route configuration details, refer to *Assign Route Package Template* on page 887.
4. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 2.

5. Create a new route by clicking the **New Route** button in the *Specific Settings* pane. For route configuration details, refer to *New Route* on page 881.
    a. Configure the new route's name and (optionally) description.
    b. Add and configure an External Script step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For External Script configuration details, refer to *External Script* on page 916.
        i. Specify the external script path and arguments as follows:

        ```
        /bin/sh -C ${FILEDRIVEHOME}/bin/agents/7zip-compress.sh archive-${timestamp}.7z
        ```

        ii. Select **Log script's standard output to Server log**.
        iii. (Optional) Configure the rest of the settings.
        iv. Click **Save** when done.
    c. Add and configure a Publish to Account step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For Publish To Account configuration details, refer to *Publish To Account* on page 943.
        i. Uncheck **Proceed with route execution on step failure**.
        ii. Select an account to publish to (for example, the current account).
        iii. Select a folder to publish the file to (for example, the subscription folder configured in Step 2).
        iv. (Optional) Configure the rest of the settings.
        v. Click **Save** when done.
6. Save the route and the route package.

Script example

```
#!/bin/sh

SEVENZIP="<path to '7z' executable>"

if [ "X${ST_ACCOUNT_HOME}" = "X" ]; then
        echo "ST_ACCOUNT_HOME environment variable not set, aborting."
        exit 1
fi

# Dump the environment in the account home folder
env > ${ST_ACCOUNT_HOME}/dumpenv.${$}

if [ ! -x $SEVENZIP ]; then
        echo "\"$SEVENZIP\" does not exist or is not an executable,
```

```
aborting."
        exit 2
fi

# Go to the sandbox folder
cd $SANDBOX_FOLDER

# Keep track of the files that will be archived to delete them later
FILELIST=`ls`

$SEVENZIP a $1 *
exitcode=$?

if [ $exitcode -ne 0 ]; then
        echo "Failed to compress files \"$FILELIST\", aborting."
        exit $exitcode
fi

# Delete the archived files
for file in $FILELIST ;
do
rm -f $file;
done
```

Flow of events

1. Multiple files are uploaded via any protocol to the Advanced Routing subscription folder.
2. The trigger file (file with `.trigger` extension) is uploaded to the subscription folder.
3. The Advanced Routing application triggers the route.
4. The uploaded files are compressed into a single 7zip archive and published to the subscription folder.

Note In the end there will be several files present in the subscription folder – the input files, the trigger file, and the archive. To automatically remove the input files and the trigger file, select **Post Processing Action > On Success > Delete** in the subscription settings.

Note Publishing the archive file in Advanced Routing subscription folder does not trigger the route execution once again, because *Trigger target subscription actions* is **unchecked**.

Send To Partner (PeSIT)

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the Send To Partner (PeSIT) use case:

Overview

Each incoming file is routed to a remote transfer site over PeSIT and the file is archived to a local folder.

Prerequisites

- Create a Route Package Template. For Route Package Template creation details, refer to *Add Route Package Template* on page 876.
- Create an Advanced Routing application instance. For Advanced Routing application instance creation details, refer to *Create Advanced Routing application* on page 874.
- Create a SecureTransport user account. For user account creation details, refer to *User accounts* on page 501.
- Create a PeSIT transfer site to route the file to. For PeSIT transfer site creation details, refer to *PeSIT transfer sites* on page 603.
- Create a transfer profile that is used for the transfer. For transfer profile configuration details, refer to *Transfer profiles* on page 640.
- Create a Folder Monitor transfer site that is used to archive the incoming files. For Folder Monitor transfer site configuration details, refer to *Folder Monitor transfer sites* on page 557.

Steps to configure the flow

1. Create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.
    a. Uncheck **Proceed with route execution on step failure**.
    b. Configure the subscription folder.
    c. (Optional) Configure the rest of the settings.
    d. Click **Add** when done.
2. Navigate to the *Routes* tab of the created account and assign a new route package to the account by choosing the created Route Package Template and clicking the **Assign Route** button. For assigning a route configuration details, refer to *Assign Route Package Template* on page 887.
3. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 1.

4. Create a new route by clicking the **New Route** button in the *Specific Settings* pane. For route configuration details, refer to *New Route* on page 881.
    a. Configure the new route's name and (optionally) description.
    b. Add and configure a Send to Partner step by selecting it from the *-- Select Step --* drop-down menu and clicking the **Add Step** button. For Send To Partner configuration details, refer to *Send To Partner* on page 948.
        i. Select the account which contains the target transfer site (or select **Use current account**).
        ii. Select the created PeSIT transfer site.
        iii. Select the created Transfer Profile.
        iv. Optionally, configure the **Advanced PeSIT Settings**. For example, to trigger Store and Forward:
            I. Click the **Configure advanced PeSIT settings** checkbox.
            II. Set the *Final Destination* field to the desired final destination.
        v. (Optional) Configure the rest of the step's settings.
        vi. Click **Save** when done.
    c. Add and configure a Send To Partner step by selecting it from the *-- Select Step --* drop-down menu and clicking the **Add Step** button. For Send To Partner configuration details, refer to *Send To Partner* on page 948.
        i. Select the account which contains the Folder Monitor transfer site (or select **Use current account**).
        ii. Select the Folder Monitor transfer site from the selected account.
        iii. Click **Save** when done.
5. Save the route and the route package.

Flow of events

1. A file is uploaded via any protocol to the Advanced Routing subscription folder.
2. The Advanced Routing application triggers route.
3. The uploaded file is routed to the remote PeSIT transfer site and after that is archived to the folder specified in the Folder Monitor transfer site.

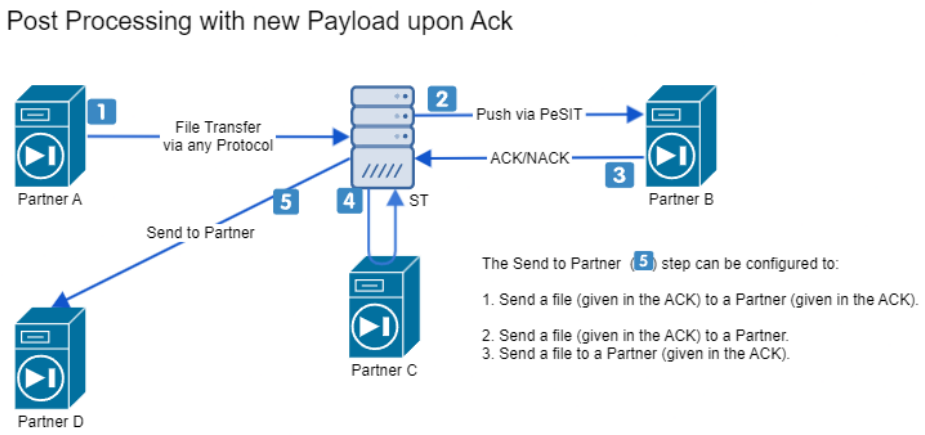
Note To compress the file before sending it to the archive folder, add a Compress step before the second Send To Partner step.

Advanced use cases

An overview, the prerequisites, and the flow of events of the following advanced use cases are provided:

Transmitting specific files to designated destinations as specified in a PeSIT acknowledgment

Use case: The arrival of a file from Partner A triggers an AR flow where SecureTransport forwards the file to Partner B via the PeSIT protocol and employs the information within the PeSIT acknowledgment received from Partner B to identify the files to be retrieved from Partner C and their designated destination for transmission to Partner D.



Let's break down the Advanced Routing flow into steps:

1. The arrival of file from Partner A in ST triggers an Advanced Routing flow.
2. As a first step of this AR flow, ST sends the received file to Partner B via the PeSIT protocol and waits for an acknowledgment with a user message (PI 91) containing the information needed for the next steps in the flow.
   The PI 91 is pre-agreed by the PeSIT partners and set by the partner that sends the acknowledgment. In this case, we set the following message on Partner B side:
   ACCOUNT@TRANSFERSITE@DESTINATIONFOLDER@RECEIVEFILEAS@DOWNLOADFOLDER@DOWNLOADPATTERN@ACCOUNT(second S2P)@TRANSFERSITE(second S2P)
3. ST receives the acknowledgment from Partner B, which triggers a Pull From Partner step utilizing the information highlighted above. It downloads the files that match the DOWNLOADPATTERN from the location ACCOUNT@TRANSFERSITE@DOWNLOADFOLDER to the target location DESTINATIONFOLDER under the name specified in the RECEIVEFILEAS.
4. The successfully downloaded files are sent to Partner D, whose "address" is specified in the second segment of the User Message sent by Partner B.

Configuration steps:

1. Create a user account *U1* on the server with the following elements configured:
   - PeSIT transfer site and transfer profile
   - HTTP transfer site (could be any protocol)
2. Go to that user's *Route* tab and create a composite route *CR* with the *Execution Rule: All Matching Routes*.
   CR will consists of three basic routes: *R1*, *R2*, and *R3*.
3. Define a route *R1* that will handle the file transmission via PeSIT to Partner B :
   - Condition: Expression Language `${transfer.trigger ne 'ack' and transfer.trigger ne 'nack'}`
   - Add Step: Send To Partner
     - Condition: `${transfer.trigger eq 'client_upload'}`
     - Account Transfer Sites: the U1's PeSIT transfer site
4. Define a route *R2* that will handle the pulling of files from Partner C, specified in the User Message from Partner B.
   - Condition: Expression Language `${transfer.trigger eq 'ack' or transfer.trigger eq 'nack'}`
   - Add Steps: Pull From Partner
     - Specify An Account: `${extract(pesit.pi.msgData, '@', 1)}`
     - Account Transfer Site: `${extract(pesit.pi.msgData, '@', 2)}`
     - Destination Folder: `${extract(pesit.pi.msgData, '@', 3)}`
       It must be the same as the subscription folder of the composite route *CR*.
     - Receive File As: `${extract(pesit.pi.msgData, '@', 4)}`

This setup is applicable when pulling a single file. When pulling multiple files, the *Receive File As* field in the step configuration should be left empty because all the files will be downloaded under the same name; hence, any subsequent file will overwrite the preceding one.

- Download Folder: `${extract(pesit.pi.msgData, '@', 5)}`
- Download Pattern: `${extract(pesit.pi.msgData, '@', 6)}`

5. Define a route *R3* that will send the file/s downloaded from Partner C to Partner D, specified in the acknowledgment sent from Partner B:
   - Condition: Expression Language `${transfer.trigger eq 'server_pull' and transfer.status eq 'success'}`
   - Add Step: Send To Partner
     - Specify An Account Name: `${extract(transfer.ackMsgData, '@', 7)}`
     - Account Transfer Sites: `${extract(transfer.ackMsgData, '@', 8)}`

       Both the assignor (Partner C) and the relay (ST) must be aware of the final destination (Partner D transfer site).

       There is no limitation on the transfer protocol. However, if you choose PeSIT, the automatic acknowledgment must be turned off on the Partner D side to prevent an infinite loop in the flow.
6. Subscribe *U1* to an Advanced Routing application, where
   - Subscription Folder: the destination folder for the files received form Partner A
   - Route: *CR*
   - **Trigger Route Execution on PeSIT Acknowledgment**: selected

Send a file specified via PeSIT acknowledgment to a partner

Use case: SecureTransport pulls a specific file based on the information received with a PeSIT acknowledgment, and forwards it to a partner.

To achieve this scenario, we will configure the following AR flow:

1. Route 1 with a Send To Partner step will use the PeSIT protocol to send the files successfully uploaded by a client to the subscription folder to a partner.
2. The arrival of a PeSIT acknowledgment (positive or negative) will trigger Route 2 with a Pull From Partner step that will use the information in the User message field of the ACK/NACK to identify the file to pull.
3. The successful pull will trigger Route 3 with a Send To Partner step that transmits the downloaded file to another partner.

Prerequisites:

- SecureTransport must have PeSIT partnership defined with all partners in the transfer flow.
- The partners must agree upon the User message structure in advance. In this case, we will the following message on the partner that sends the acknowledgment: ACCOUNT@TRANSFERSITE@RECFILEAS@DOWNFOLDER@DOWNPATTERN

Configuration steps:

1. Create a user account *U1* with the following elements configured:
   - Two PeSIT transfer sites that will be used by the Send To Partner steps.
   - One non-PeSIT transfer site that will be used by the Pull From Partner step.
2. Go to the account's Route tab and create a composite route: *CR1*. It will consist of three routes.
3. Define a route *R1* with the following characteristics:
   - Condition: Expression Language `${transfer.trigger ne 'ack' and transfer.trigger ne 'nack'}`
   - Add Step: Send To Partner with EL condition `${transfer.trigger eq 'client_upload'};`
4. Define a route *R2* with the following characteristics:
   - Condition: Expression Language `${transfer.trigger eq 'ack'}`
   - Add Step: Pull From Partner with the following configuration:
     - Specify An Account: `${extract(pesit.pi.msgData, '@', 1)}`
     - Account Transfer Sites: `${extract(pesit.pi.msgData, '@', 2)}`
     - Receive File As: `${extract(pesit.pi.msgData, '@', 3)}`
     - Download Folder: the same folder will be used as a subscription folder.
     - Download Pattern: `${extract(pesit.pi.msgData, '@', 5)}`
     - Destination Folder: The same as the subscription folder of the composite route.
5. Define a route *R3* with the following characteristics:
   - Condition: Expression Language `${transfer.trigger eq 'server_pull' and transfer.status eq 'success'}`
   - Add Step: Send To Partner with EL condition `${transfer.trigger eq 'client_upload'};` and partner details specified.
6. Subscribe *U1* to an Advanced Routing application. In the subscription settings:

- Route: *CR*
- Subscription folder: The download folder of the transfer site, specified in the Pull From Partner step.
- **Trigger Route Execution on PeSIT Acknowledgment**: selected

Configure AR flow with payload based on received acknowledgment

Use case: SecureTransport acts as a relay: It sends a file to a partner and, upon receiving a positive acknowledgment, forwards the same file over HTTP to a partner that is specified in the User message (PI 91) field of the received ACK.

To achieve this scenario, we will configure the following AR flow:

1. For SecureTransport to receive a PeSIT acknowledgment, we will create a Route 1 with a Send To Partner step
   Route 1 with a Send To Partner step that transmits the file to Partner 1 over PeSIT. It will be executed when a file arrives in the subscription folder and must NOT be triggered upon acknowledgment arrival.
2. Route 2 with a Send To Partner step that is triggered upon the arrival of a positive acknowledge from Partner 1. It forwards the file to the recipient, specified in User Message field of the ACK.

Prerequisites:

The partners must agree upon the User Message format in advance. It must be configured on the partner that receives the file and will acknowledge it.

Caution If SecureTransport acts as an intermediate relay between two PeSIT partners with auto acknowledgments enabled, the final receiver must not have auto acknowledgment enabled, as this would cause the AR process to enter an infinite loop.

If you need to configure the User Message format on you server, go to **Operations > Server Configuration** and set it in the *Pesit.Transfer.Acknowledge* configuration option.

For illustrative purposes, we will use the following message:
`ACCOUNT@TRANSFER_SITE@OVERWRITE_UP_FOLDER@ROUTE_FILE_AS.`

Configuration steps:

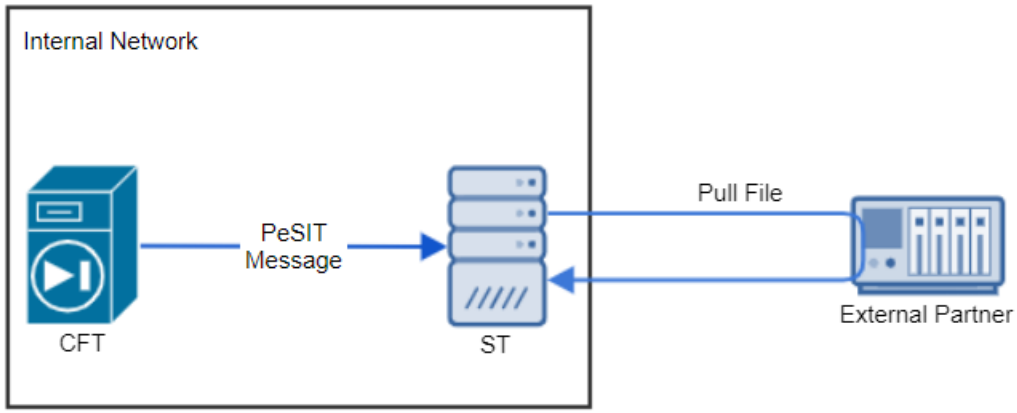
1. Create a user account *U1* on the server with the following elements configured:
   - PeSIT partnership with a user at the partner site
   - HTTP transfer site
2. Go to that user's *Route* tab and create a composite route *CR*. It will consists of two basic routes: *R1* and *R2*.
3. Define a route *R1* with the following characteristics:
   - Condition: Expression Language `${transfer.trigger ne 'ack' and transfer.trigger ne 'nack'}`
   - Add Step: Send To Partner configured to use a PeSIT Account Transfer Site.

4. Define a route *R2* with the following characteristics:
    - Condition: Expression Language `${transfer.trigger eq 'ack'}`
    - Add Step: Send To Partner
        - Specify An Account: `${extract(pesit.pi.msgData, '@', 1)}`
        - Account Transfer Site: `${extract(pesit.pi.msgData, '@', 2)}`
        - Overwrite Upload Folder: `${extract(pesit.pi.msgData, '@', 3)}`
        - Route File As: `${extract(pesit.pi.msgData, '@', 4)}`
5. Subscribe *U1* to an Advanced Routing application, where
    - Route: *CR*.
    - **Trigger Route Execution on PeSIT Acknowledgment**: selected

Configure AR flow without payload based on received PeSIT message

Use case: The arrival of a PeSIT message transfer in SecureTransport triggers an Advanced Routing flow without payload (for example, routes that contain a Pull From Partner or External Script step).

Example configuration with a Pull From Partner step:



1. Create a user account on the server with a configured PeSIT partnership with a user at the partner site.
2. Create a Route Package Template and define a route with the following characteristics:
    - Condition: Always
    - Add step: Pull From Partner
3. (Optional) Create an Advanced Routing application if you don't already have one.
4. Go to the user's *Routes* tab, select the Route Package Template created in *Step 2* and click **Assign Route**. Enter a route name and save.
5. Go to the user's *Subscriptions* tab, select the Advanced Routing application and click **Subscribe**.

a. In the **Post Transmission Settings** section, select the **Route** that will be executed (created in *Step 4*).

b. (Optional) To ensure that this subscription is triggered exclusively by PeSIT messages (and not by files), select **Trigger processing of files based on condition** and enter the following expression as the trigger condition: `${stenv.is_pesit_message eq 'true'}`.

c. Click **Add**.

6. Go to the user's *Transfer Profiles* tab and edit the transfer profile:

a. Enable the **Advanced Properties** checkbox.

b. Enable the **Receiving Message Parameters** checkbox.

c. Select the **Download in account folder and trigger processing** option.

d. In the **Receive Message Directory** field, enter the relative path to the subscription folder.

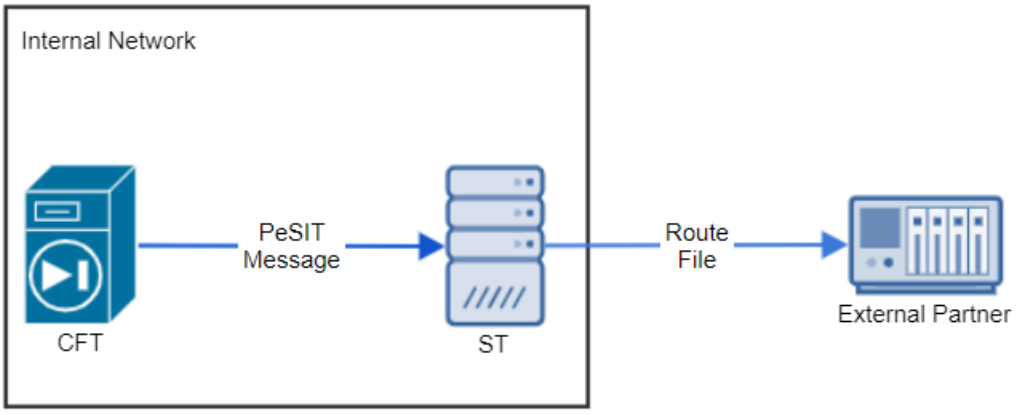
You can open the tooltip for a list of expressions that can be used. Additionally, you can use the `${extract(pesit.msgData, '@', 1)}` expression which will extract the required data directly from the received PeSIT message.

Tip If you create a Route Package Template in *Step 2* with two or more routes, you can choose which route(s) should be executed only upon the arrival of a PeSIT message. To do that, select **Expression Language** in a route's **Condition Settings** and enter the following expression: `${transfer.trigger eq 'pesit_message'}`.

Configure AR flow with payload based on received PeSIT message

Use case: The arrival of a PeSIT message transfer in SecureTransport triggers an Advanced Routing flow with payload (for example, routes that contain a Send To Partner step).

Example configuration with Send To Partner step:



1. Create a user account on the server with a configured PeSIT partnership with a user at the partner site.

2. Create a Route Package Template and define a route with the following characteristics:
    - Condition: Always
    - Add step: Send To Partner

      Instead of selecting a specific account and transfer site, you can use the `${extract (pesit.pi.msgData, '@', 1)}` expression which will extract the required data directly from the received PeSIT message. Note that `@` is the separator and `1` indicates the data's position in the message.
3. (Optional) Create an Advanced Routing application if you don't already have one.
4. Go to the user's *Routes* tab, select the Route Package Template created in *Step 2* and click **Assign Route**. Enter a route name and save.
5. Go to the user's *Subscriptions* tab, select the Advanced Routing application and click **Subscribe**.
    a. In the **Post Transmission Settings** section, select the **Route** that will be executed (created *Step 4*).
    b. Select **Trigger processing of files based on condition** and enter the following expression as the trigger condition:

    ```
    ${stenv.is_pesit_message eq 'true'}
    ```

    c. Click **Add**.
6. Go to the user's *Transfer Profiles* tab and edit the transfer profile:
    a. Enable the **Advanced Properties** checkbox.
    b. Enable the **Receiving Message Parameters** checkbox.
    c. Select the **Download in account folder and trigger processing** option.
    d. In the **Receive Message Directory** field, enter the relative path to the subscription folder.

    You can open the tooltip for a list of expressions that can be used. Additionally, you can use the `${extract(pesit.msgData, '@', 1)}` expression.

Tip If you create a Route Package Template in *Step 2* with two or more routes, you can choose which route(s) should be executed only upon the arrival of a PeSIT message. To do that, select **Expression Language** in the route's **Condition Settings** and enter the following expression:
`${transfer.trigger eq 'pesit_message'}`

Trigger a Send To Partner step on received acknowledgment

Let's take a look at a composite route consisting of two routes:

- *STP1* contains a Send To Partner step which sends a file to *Partner 2* over PeSIT
    - Route trigger condition: `${transfer.trigger ne 'ack' and transfer.trigger ne 'nack'}`

- *STP2* contains a Send To Partner step which sends a file to *Partner 3* over FTP (or any other protocol)
    - Route trigger condition: `${transfer.trigger eq 'ack' or transfer.trigger eq 'nack'}`

If you add this route to a subscription with a trigger condition `${stenv.is_pesit_message eq 'true'}`, only the first Send To Partner step will be executed successfully. The second step will start execution but will not process any files since there are none that can be used. Note that if you add a Pull From Partner step which downloads files in the subscription directory, they will not be processed until a PeSIT message is received.

Workaround: If all routes with payload in your setup use the same file(s), make the following modifications:

1. Change the subscription trigger condition to accept either a PeSIT message or a trigger file:

   ```
   ${stenv.is_pesit_message eq 'true' or stenv['target'].matches ('.*\\.trigger')?1:0}
   ```

2. Change the trigger condition for route *STP1* to:

   ```
   ${transfer.trigger eq 'pesit_message' and transfer.trigger ne 'ack' and transfer.trigger ne 'nack'}
   ```

3. Change the trigger condition for route *STP2* to:

   ```
   ${transfer.trigger ne 'pesit_message' and transfer.trigger ne 'ack' and transfer.trigger ne 'nack'}
   ```

4. Add a new route *PFP*:
    - Condition: Expression Language `${transfer.trigger eq 'ack' or transfer.trigger eq 'nack'}`
    - Add step: Pull From Partner with the following configuration:
        - Condition: Always
        - Destination Folder: must match the subscription folder
        - Receive File As pattern: `file.trigger`
5. Check that the order of the routes is as follows: *STP1*, *PFP*, *STP2*.

The configuration above will achieve the following scenario:

1. *Partner 1* receives a PeSIT message.
2. The PeSIT message triggers route *STP1*: *Partner 1* sends the file *X* to *Partner 2* (over PeSIT).
3. *Partner 2* receives the file and acknowledges it.
4. *Partner 1* receives the acknowledgment.
5. The acknowledgment triggers route *PFP*: the trigger file is downloaded in the subscription folder.
6. The trigger file starts route *STP2*: *Partner 1* sends the file *X* to *Partner 3* (over any protocol).

Route files downloaded with Pull From Partner to an external target

Use case: An event triggers the download of files from an external source. The files are then sent to an external target.

Steps to configure the flow:

1. Create an Advanced Routing application: *routing*
2. Create and configure a subscription to that Advanced Routing application that will trigger a route handling the first part of the use case (downloading files from external partners): */triggerPull*
3. Create another subscription to the same application that will trigger a route, on success, which handles the second part of the use case (sending the file to an external partner): */triggerPush*
4. Create the route that will trigger the downloads: *Downloads*
    a. Add a Pull From Partner step and configure the following:
        - The account and the transfer site to pull files from
        - Download folder
        - Download pattern
        - In the **Destination Folder** field, enter the subscription folder that triggers the route which sends the file to the partner: */triggerPush*.
5. Create a second route that handles the file uploads to the external partner: *Uploads*
    a. In the route configuration, add the following condition:
    `${transfer.trigger eq 'server_pull' and transfer.status eq 'success'}`
    This way the route will be triggered only if the preceding pull action completes successfully.
    b. Add a *Send to Partner* step and configure the account and the transfer site to which the file will be sent.
6. Navigate to the */triggerPull* subscription. In the **Post Transmission Settings** section, configure the route *Downloads* to be executed on each transfer status with a Route action.
7. Navigate to the */triggerPush* subscription. In the **Post Transmission Settings** settings, configure the route *Uploads* to be executed for each transfer status with a Route action.

Use JMS metadata values for Pull From Partner step

A Pull From Partner step can be triggered based on metadata received from external messaging systems over JMS. For more details on how to set up a JMS transfer site, see JMS Connector documentation. This topic assumes that you have deployed the JMS Connector, and focuses on the configuration steps that have to be executed to fulfill the goal of the use case.

Use Case: Pull files from a specific destination based on information (metadata) received from a JMS scheduled pull (via subscription), and send a message to JMS that contains information about the above flow.

Flow attributes environment is persisted and passed between steps (configured in the same route) and between different routes. Therefore, you can use flow attributes to trigger multiple flow actions (pull and push) based on metadata received from external messaging systems and not only.

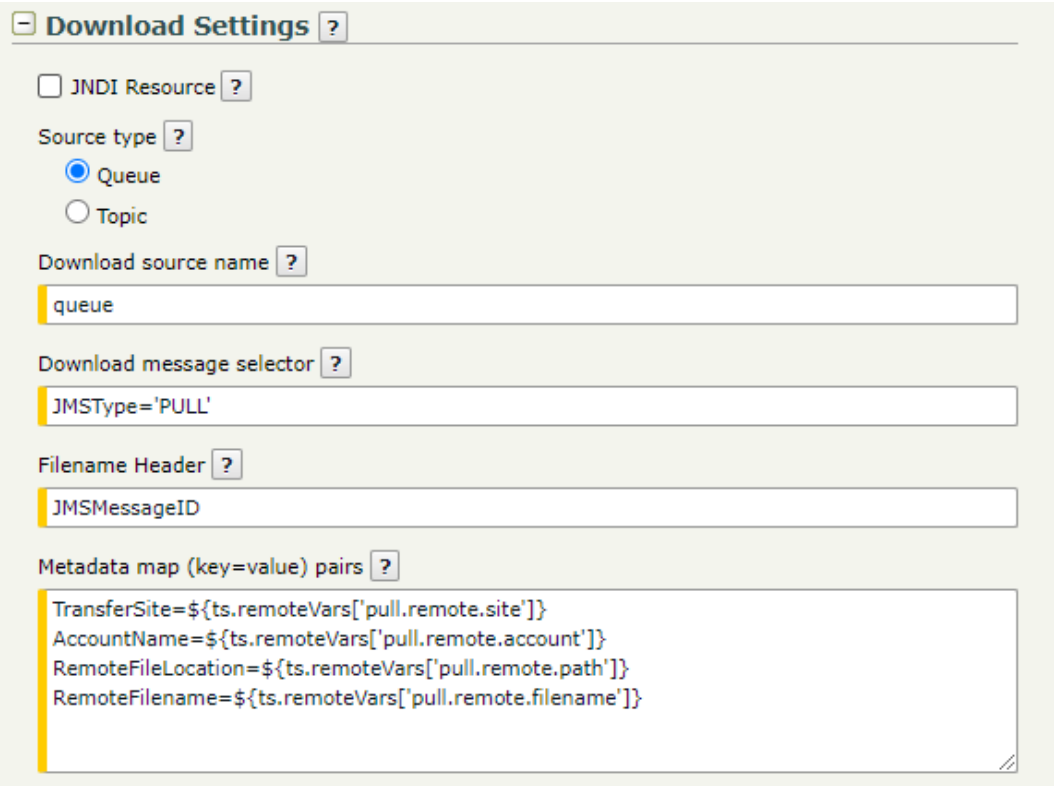
In our example, the metadata hosted by the external messaging system contains the following:

- transfer site that contains the files you want to pull: `pull.remote.site=SFTP_TS`
- partner account that holds the transfer site you want to pull files from: `pull.remote.account=user1`
- remote file location: `pull.remote.path=/dir1/dir2`
- remote filename: `pull.remote.filename=file.txt`

The result would be downloading */dir1/dir2/file.txt* from account *user1* and transfer site *SFTP_TS* dynamically using the Pull From Partner step.

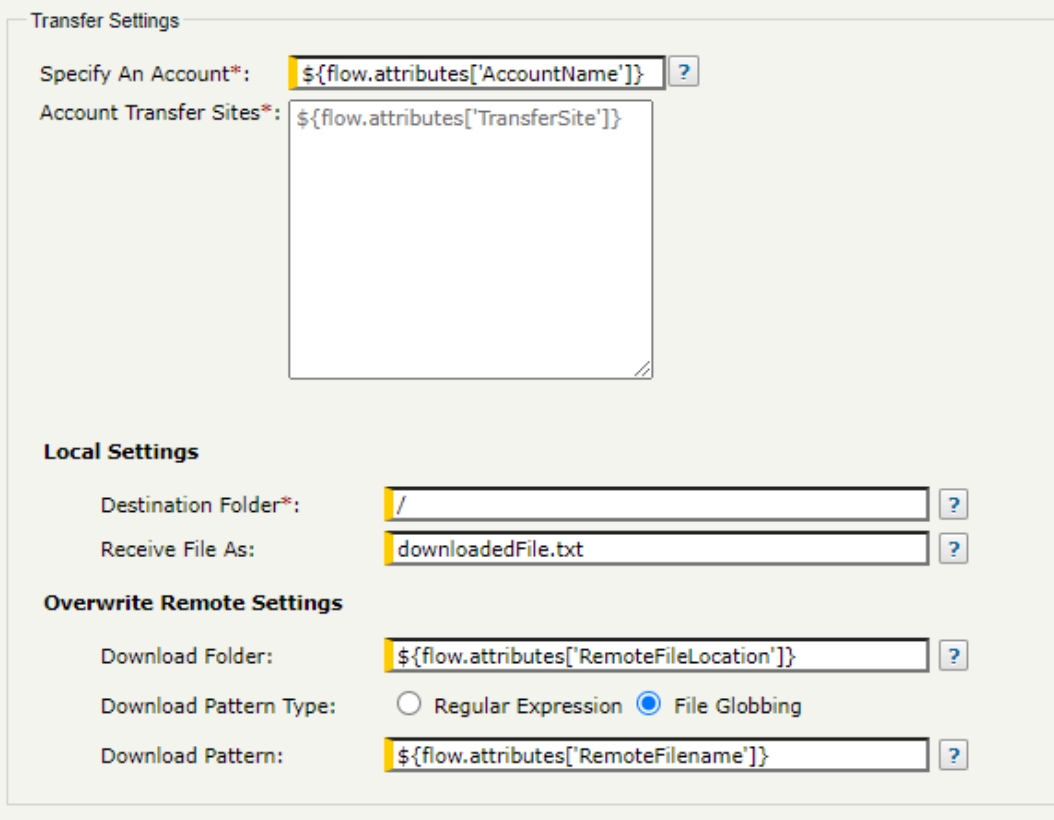
Steps to configure the flow:

1. To pull information from the messaging system, enter the metadata that will be stored in the SecureTransport's flow attributes environment in the JMS transfer site's **Metadata Map** field, as shown in the image below.

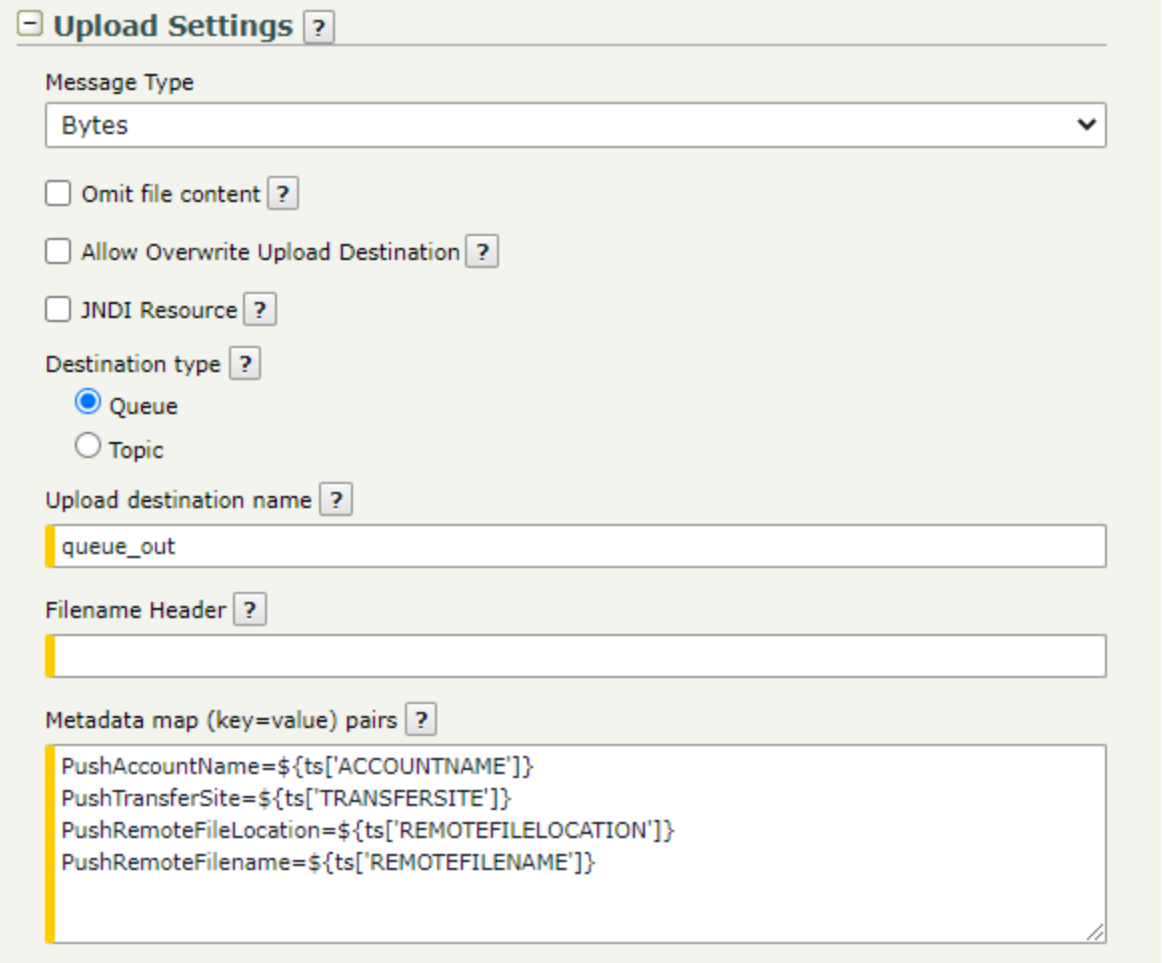


2. Create a route that is triggered by JMS pulls.

3. In that route, add a *Pull From Partner* step. The step is used to download the file that is specified in the metadata pulled from the external messaging system. Therefore, it needs to access the flow attributes environment and use the information provided there. The example below shows how to configure the step.



4. To send a message to JMS that contains information about the above flow, let's say the account name, the expression in the **Upload Metadata map** must have the following format:
`PushAccountName=${ts['ACCOUNTNAME']}`

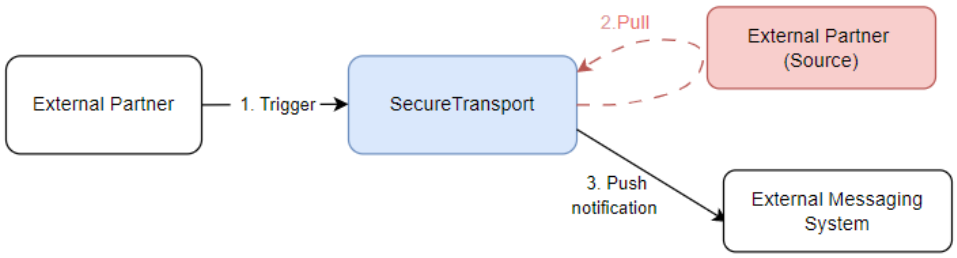


To access and send the transferID and the coreID attributes, use the following expressions:

```
TransferId=${ts.transferVars['source_status_id']}
CoreId=${ts.transferVars['source_core_id']}
```

Notify external messaging systems (JMS) of pull failures

Use case: An event triggers the download of files from an external source. If the download fails, a notification to an external messaging system (JMS) will be sent.



This topic focuses on the configuration steps that have to be performed in SecureTransport to fulfill the goal of the use case. It assumes that you have deployed the JMS Connector. For instructions on how to install and configure a JMS transfer site, see the JMS Connector for SecureTransport readme file.

Steps to configure the flow:

1. Create an Advanced Routing application.
2. Create a route that will trigger the downloads: *Downloads*.
3. In that route, add a Pull from Partner step and specify the following:
    a. The account and transfer site to pull files from
    b. Under *Local Settings*, in the **Destination Folder** field, enter the subscription folder that triggers the route that sends the notification: */triggerNotification* (Step 6).
    c. Under *Overwrite Remote Settings*, specify **Download Folder** and **Download Pattern**.
    d. Select the **Send Trigger File On Failure** checkbox and specify the name and optionally the content of the trigger file.
4. Create a second route that handles the notifications via JMS: *SendNotification*
5. In that route, add a Send To Partner step and specify the following:
    a. The account and the JMS transfer site
    b. Select **Overwrite Upload Folder** section and specify the upload queue to which the messages will be sent.

    This value overwrites the value defined in the *Upload destination name* section in the JMS transfer site configuration. To use the value defined in the transfer site, leave the field in Send to Partner step empty.
6. Create and configure a subscription to that Advanced Routing application which will trigger a route handling the first part of the use case (downloading files from external partners): */triggerPull*
7. In the */triggerPull* subscription, in the **Post Transmission Settings** section, configure the route *Downloads* to be executed on each transfer status with a Route action.
8. Create another subscription to the same application that will trigger a route, which handles the second part of the use case (sending the notification via JMS): */triggerNotification*.
9. In the */triggerNotification* subscription, configure the following:
    a. In *Post Transmission Settings*, select **Route**.
    b. In *Post Transmission Settings – On Success*, select the **Execute route when the remote server returns no files** checkbox.
    c. In *Post Transmission Settings – On Failure*, select the **Route** action, **Submit the transferred file(s) to the route for processing** and **Execute route when the remote server returns no files** .

Results:

Notification to JMS is sent in the following cases:

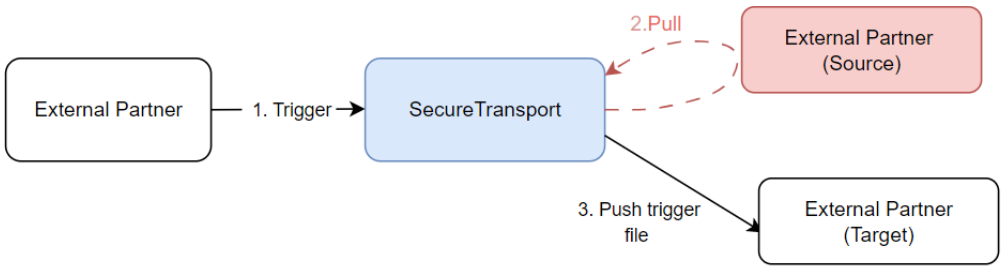
- When no file is found on attempt to pull a file with a static pattern. The notification is sent after all retry attempts are exhausted.

	RESUBMIT				ACCOUNT	LOGIN	DIRECTION	ACTION BY	FILE	BYTES TRANSFERRED	PROTOCOL
☐	Resubmit		✓		Target	Target	Outbound	Server	SendTriggerFileAfterPullKO	0.01 KB	jms
☐			✗		Target	Target	Inbound	Server	pullFileViaFTP.txt	0 KB	ftp
☐			✗		Target	Target	Inbound	Server	pullFileViaFTP.txt	0 KB	ftp
☐			✗		Target	Target	Inbound	Server	pullFileViaFTP.txt	0 KB	ftp
☐	Resubmit		✓		Target	Target	Inbound	User	file.txt	10.02 KB	ftp

- When no files are found on attempt to pull multiple files using a wildcard pattern.
- When the download fails due to a connection issue.

Send a trigger file via SFTP on pull failures

Use case: An event triggers the download of files from an external source. If the download fails (permanent failure), a trigger file will be routed via SFTP to an external partner.



Steps to configure the flow:

Follow the configuration steps described in *Notify external messaging systems (JMS) of pull failures* on page 988 but replace the JMS configuration *(Step 4)* with an SFTP configuration (the transfer Site and the account that holds it).

Results:

The trigger file defined in the Pull From Partner step is sent via SFTP in the following cases:

- When no file is found on attempt to download a file with a static pattern.

	RESUBMIT				ACCOUNT	LOGIN	DIRECTION	ACTION BY	FILE	BYTES TRANSFERRED	PROTOCOL
☐	Resubmit		✓		Target	Target	Outbound	Server	SendTriggerFileAfterPullKO	0.01 KB	ssh
☐			✗		Target	Target	Inbound	Server	pullFileViaFTP.txt	0 KB	ftp
☐			✗		Target	Target	Inbound	Server	pullFileViaFTP.txt	0 KB	ftp
☐			✗		Target	Target	Inbound	Server	pullFileViaFTP.txt	0 KB	ftp
☐	Resubmit		✓		Target	Target	Inbound	User	file.txt	10.02 KB	ftp

- When no files are found on attempt to pull multiple files using a wildcard pattern.
- When the download fails due to a connection issue.

Route files based on file name extension

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the route files based on file name extension use case:

Overview

Identify the Routing Destination based on filename following the convention `<routing-destination>_<filename>`. The custom Expression Language function `extract('some_string', '_', 2)` is utilized since it splits the given string to tokens based on a delimiter.

Prerequisites

- Create a Route Package Template. For Route Package Template creation details, refer to *Add Route Package Template* on page 876.
- Create an Advanced Routing application instance. For Advanced Routing application instance creation details, refer to *Create Advanced Routing application* on page 874.
- Create a SecureTransport user account. For user account creation details, refer to *User accounts* on page 501.
- Create two remote transfer sites (for example, named **partner1** and **partner2**) which are used as routing destinations. For remote transfer site creation details, refer to *Create a transfer site* on page 628.

Step to configure the flow

1. Create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.
    a. Configure the subscription folder.
    b. (Optional) Configure the rest of the settings.
    c. Click **Add** when done.
2. Navigate to the *Routes* tab of the created account and assign a new route package to the account by choosing the created Route Package Template and clicking the **Assign Route** button. For assigning a route configuration details, refer to *Assign Route Package Template* on page 887.
3. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 1.

4. Configure the *Execution Rule* of the route package to be **First Matching Route**.
5. Create a new route by clicking the **New Route** button in the *Specific Settings* pane. For route configuration details, refer to *New Route* on page 881.
    a. Configure the new route's name (for example, **Partner 1**) and (optionally) description.
    b. Configure the *Route Condition* to be **Expression Language** and enter the following in the expression field:

    ```
    ${extract(basename(transfer.target),'_',1) eq 'partner1'}
    ```

    c. Add and configure a Send To Partner step by selecting it from the *-- Select Step --* drop-down menu and clicking the **Add Step** button. For Send To Partner configuration details, refer to *Send To Partner* on page 948.
        i. Select the account which contains the target transfer site (or select **Use current account**).
        ii. Select the transfer site from the selected account to send the file to (for example, **partner1**).
        iii. Select **Route file as** and enter the following expression:

        ```
        ${extract(basename(transfer.target),'_',1)}
        ```

        This expression extracts the *<routing-destination>* from the filename (for example, `partner1_incoming.txt` returns `partner1`).
        iv. Click **Save** when done.
6. Create another route by clicking the **New Route** button in the *Specific Settings* pane.
    a. Configure the new route's name (for example **Partner 2**) and (optionally) description.
    b. Configure the *Route Condition* to be **Expression Language** and enter the following in the expression field:

    ```
    ${extract(basename(transfer.target),'_',1) eq 'partner2'}
    ```

    c. Add and configure a Send To Partner step by selecting it from the *-- Select Step --* drop-down menu and clicking the **Add Step** button.
        i. Select the account which contains the target transfer site (or select **Use current account**).
        ii. Select the transfer site from the selected account to send the file to (for example, **partner2**).
        iii. Select Route file as and enter the following expression:

        ```
        ${extract(basename(transfer.target),'_',2)}
        ```

        This expression extracts the *<filename>* from the filename (for example, `partner2_incoming.txt` returns `incoming`).
        iv. Click **Save** when done.
7. Save the route package.

Flow of events

1. A file named `partner1_filename.txt` is uploaded to the Advanced Routing subscription folder.
2. Advanced Routing application is triggered and `partner1` is extracted from the filename.
3. Route **Partner 1** is triggered. Route **Partner 2** is skipped.
4. File is routed to the transfer site **partner1**.
5. A file named `partner2_filename.txt` is uploaded to the Advanced Routing subscription folder.
6. The Advanced Routing application is triggered and `partner2` is extracted from the filename.
7. Route **Partner 1** is skipped. Route **Partner 2** is triggered.
8. File is routed to the transfer site **partner2**.

PGP Encryption (partner's certificate) and send to multiple partners

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the PGP Encryption and send to multiple partners use case:

Overview

Pull files from multiple sources and route the incoming files to multiple partners after PGP encrypting the files with the partner's PGP certificate.

Prerequisites

- Create an Advanced Routing application instance. For Advanced Routing application instance creation details, refer to *Create Advanced Routing application* on page 874.
- Create two partner accounts (for example, accounts with names **partner1** and **partner2**). For user account creation details, refer to *User accounts* on page 501.
    - Create transfer site in each account to be used as a routing destination to the respective partner.

      Note Modify the access level of the transfer sites to be **Public**, so these transfer sites can be used in routes defined outside of this account.

- Generate or import a partner PGP certificate in each account that is used for encrypting the files before routing them to the partner

  Note Modify the access level of the certificates to be **Public**, so these certificates can be used in routes defined outside of this account.

- Create two local accounts (for example, accounts with name **local1** and **local2**) and transfer sites (for example, named **target1** and **target2**) which are used as source for pulling. For user account creation details, refer to *User accounts* on page 501.

Steps to configure the flow

1. Create a Route Package Template by navigating to **Routes** and clicking **New Route Package Template**. For Route Package Template creation details, refer to *Add Route Package Template* on page 876.
   a. Configure the new Route Package Template's name and (optionally) description.
   b. Create a new route by clicking the **New Route** button. For route configuration details, refer to *New Route* on page 881.
      i. Configure the new route's name (for example, **Partner 1**) and (optionally) description.
      ii. Add and configure a PGP Encryption step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For PGP Encryption configuration details, refer to *PGP Encryption* on page 896.
         I. Select the **Encrypt only** option.
         II. Select the first partner account.
         III. Select the PGP certificate for encryption from that account.
         IV. Click **Save** when done.
      iii. Add and configure a Send To Partner step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For Send To Partner configuration details, refer to *Send To Partner* on page 948.
         I. Uncheck **Proceed with route execution on step failure**.
         II. Select the first partner account which contains the target transfer site.
         III. Select the transfer site from the selected account to send the file to.
         IV. Click **Save** when done.

c. Save the route and create a new one by clicking the **New Route** button.

i. Configure the new route's name (for example, **Partner 2**) and (optionally) description.

ii. Add and configure a PGP Encryption step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button.

I. Select the **Encrypt only** option.

II. Select the second partner account.

III. Select the PGP certificate for encryption from that account.

IV. Click **Save** when done.

iii. Add and configure a Send To Partner step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button.

I. Uncheck **Proceed with route execution on step failure**.

II. Select the second partner account which contains the target transfer site.

III. Select the transfer site from the selected account to send the file to.

IV. Click **Save** when done.

d. Save the Route Package Template.

2. Go to the first local account (**local1**) and create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.

a. Configure the subscription folder.

b. Select **Automatically Retrieve Files From** checkbox and choose the transfer site to pull files from (the one created as a prerequisite).

c. Configure a schedule for pulling the files.

d. (Optional) Configure the rest of the settings.

e. Click **Add** when done.

3. Navigate to the *Routes* tab of the same account and assign a new route package to the account by choosing the Route Package Template created in Step 1 and clicking the **Assign Route** button.

a. Configure the new route package's name and (optionally) description.

b. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 2.

c. Click **Save** when done.

Note There's no need to create routes within the route package as the routes inherited from the Route Package Template are sufficient for the current use case.

4. Go to the second local account (**local2**) and create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button.

a. Configure the subscription folder.

b. Select **Automatically Retrieve Files From** checkbox and choose the transfer site to pull files from (the one created as a prerequisite).

c. Configure a schedule for pulling the files.

d. (Optional) Configure the rest of the settings.

e. Click **Add** when done.

5. Navigate to the *Routes* tab of the same account and assign a new route package to the account by choosing the route template package created in Step 1 and clicking the **Assign Route** button.

a. Configure the new route package's name and (optionally) description.

b. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 2.

c. Click **Save** when done.

Note There's no need to create routes within the route package as the routes inherited from the Route Package Template are sufficient for the current use case.

Flow of events

1. A scheduled SIT pull is triggered as configured in account **local1's** subscription.
2. The files from the remote site are downloaded in the local subscription folder and the Advanced Routing application is triggered.
3. Files are processed by the two routes configured in the Route Package Template:
   - The first route PGP encrypts the file with the certificate imported in **partner1** account and routes the file to **partner1**.
   - The second route PGP encrypts the file with the certificate imported in **partner2** account and routes the file to **partner2**.
4. A scheduled SIT pull is triggered as configured in account **local2's** subscription.
5. The files from the remote site are downloaded in the local subscription folder and the Advanced Routing application is triggered.
6. Files are processed by the two routes configured in the Route Package Template:
   - The first route PGP encrypts the file with the certificate imported in **partner1** account and routes the file to **partner1**.
   - The second route PGP encrypts the file with the certificate imported in **partner2** account and routes the file to **partner2**.

Decompress and Send to Partner (trigger file output)

The following topics provide the overview, prerequisites, configuration steps, and flow of events for the Decompress and Send To Partner with trigger file output use case:

Overview

Decompress incoming archives, send the result files to two remote transfer sites and send a trigger file to one of them.

Prerequisites

- Create a Route Package Template. For Route Package Template creation details, refer to *Add Route Package Template* on page 876.
- Create an Advanced Routing application instance. For Advanced Routing application instance creation details, refer to *Create Advanced Routing application* on page 874.
- Create an SecureTransport user account. For user account creation details, refer to *User accounts* on page 501.
- Create remote transfer sites which are used as routing destinations (for example, **Partner 1 (FTP)** and **Partner 2 (HTTP)**). For remote transfer site creation details, refer to *Create a transfer site* on page 628.

Steps to configure the flow:

1. Create and configure a subscription to the Advanced Routing application by navigating to the account's *Subscriptions* tab and clicking the **Subscribe...** button. For Advanced Routing subscription configuration details, refer to *Subscribe to Advanced Routing application* on page 888.
    a. Configure the subscription folder.
    b. (Optional) Configure the rest of the settings.
    c. Click **Add** when done.
2. Navigate to the *Routes* tab of the created account and assign a new route package to the account by choosing the created Route Package Template and clicking the **Assign Route** button. For assigning a route configuration details, refer to *Assign Route Package Template* on page 887.
3. Assign a subscription to the new route package by selecting **Assign** in the *Subscriptions* pane and selecting the subscription created in Step 1.
4. Create a new route by clicking the **New Route** button in the *Specific Settings* pane. For route configuration details, refer to *New Route* on page 881.
    a. Configure the new route's name and (optionally) description.
    b. Add and configure a Decompress step by selecting it from the *-- Select Step --* drop-down menu and clicking the **Add Step** button. For Decompress configuration details, refer to *Decompress* on page 908.

i. (Optional) Configure the available options.

ii. Click **Save** when done.

Note The Decompress step automatically detects the archive type.

c. Add and configure a Send To Partner step by selecting it from the -- *Select Step* -- drop-down menu and clicking the **Add Step** button. For Send To Partner configuration details, refer to *Send To Partner* on page 948.

i. Uncheck **Proceed with route execution on step failure**.

ii. Select the account which contains the target transfer sites (or select **Use current account**).

iii. Select the transfer sites from the selected account to send the files to (for example, **Partner 1 (FTP)** and **Partner 2 (HTTP)**).

iv. Select **Send trigger file**.

v. Specify trigger file name (for example, `${timestamp}.trigger`).

vi. (Optional) Specify **Trigger file content** if the remote system expects trigger file with content.

Note You can use the `${transferredfilenames}` environment variable to specify the trigger file contents. This way the trigger file contains all files transferred by the step and each file is on a new line. The trigger file contains the original file names of the transferred files even if rename have been configured in the transfer site. `\r\n` (CRLF) is used as a line separator for the content of the trigger file.

vii. Select an account and a transfer site to send the trigger file to. Usually these are the same as the ones configured in Step ii and Step iii.

viii. Click **Save** when done.

5. Save the route and the route package.

Flow of events

1. An archive file is uploaded to the Advanced Routing subscription folder. The archive contains two files named `text1.txt` and `text2.txt`.
2. The Advanced Routing application is triggered and the archive is processed by the route.
3. The archive is decompressed – `text1.txt` and `text2.txt` are the resulting files.
4. The files are routed to the selected transfer sites (**Partner 1 (FTP)** and **Partner 2 (HTTP)**).
5. A trigger file is generated with name `<current timestamp>.trigger` (for example, `1334851799648.trigger`) with contents:
   - `text1.txt`
   - `text2.txt`
6. The trigger file is routed to **Partner 1 (FTP)** transfer site only.

Advanced Routing best practices

This topic emphasizes a number of important Advanced Routing usage tips by describing the process flow of some complex scenarios.

The following Advanced Routing best practices are described:

- *Chain of route execution* on page 999 - Describes the chain of route execution Advanced Routing best practice.
- *Inherited settings versus Specific settings* on page 1000 - Describes the inherited settings versus specific settings Advanced Routing best practice.
- *Skipped transformation* on page 1000 - Describes the skipped transformation Advanced Routing best practice.
- *Transformation on multiple files* on page 1001 - Describes the transformation on multiple files Advanced Routing best practice.
- *Route failure* on page 1002 - Describes the Route failure Advanced Routing best practice.
- *Transformed file as the input to the next step* on page 1002 - Describes the transformed file as the input to the next step Advanced Routing best practice.
- *Routing to multiple transfer sites* on page 1003 - Describes the routing to multiple transfer sites Advanced Routing best practice.

Chain of route execution

The following Advanced Routing steps are configured:

- Route 1:
  - Transformation 1: Compression
  - Routing 1: Publish To Account
- Route 2
  - Transformation 1: Compression
  - Transformation 2: PGP Encryption
  - Routing 1: Send To Partner

The uploaded file triggers Route 1 and Route 2 in the specified order. The execution of Route 1 compresses the original file and routes the transformed (compressed) file to an account for publishing. The execution of Route 2 compresses the original file, PGP encrypts the compressed file, and routes the transformed (compressed and PGP encrypted) file to a partner transfer site.

Each route works with a copy of the original file. During the route execution the copy of the original file is passed from step to step. See *Transformed file as the input to the next step* on page 1002 for more information.

Note Execution of the routes is sequential, not simultaneous.

Note It is highly recommended that the last step of every route is a routing step (Send To Partner or Publish To Account). The reason is that only routing steps can move (send or publish) the transformed file or files outside of the sandbox folder. Having only transformation steps properly transforms the file in the sandbox folder, but when the route terminates the transformed files is deleted along with the sandbox folder.

Related topics:

- *Inherited settings versus Specific settings* on page 1000
- *Skipped transformation* on page 1000
- *Transformation on multiple files* on page 1001
- *Route failure* on page 1002
- *Transformed file as the input to the next step* on page 1002
- *Routing to multiple transfer sites* on page 1003

Inherited settings versus Specific settings

The following Advanced Routing steps are configured:

- Inherited Settings:
  - Route 1:
    - Transformation 1: Compression
    - Routing 1: Publish To Account
- Specific Settings:
  - Route 2:
    - Transformation 1: PGP Encryption
    - Routing 1: Send To Partner

The uploaded file triggers Route 1 (Inherited Settings) and Route 2 (Specific Settings). The Inherited Settings compress the original file and routes the transformed (compress) file to an account for publishing. The Specific Settings PGP encrypt the original file and routes the transformed (PGP encrypted) file to a partner transfer site.

The Inherited Settings have a higher priority than the Specific Settings.

Note It is highly recommended that the last step of every route is a routing step (Send To Partner or Publish To Account). The reason is that only routing steps can move (send or publish) the transformed file or files outside of the sandbox folder. Having only transformation steps properly transforms the file in the sandbox folder, but when the route terminates the transformed files are deleted along with the sandbox folder.

Skipped transformation

The Advanced Routing routes are configured:

- Route 1:
    - Transformation 1: PGP Decryption where **Require Encryption** is not selected
    - Route 1: Publish To Account
- Route 2:
    - Transformation 1: Line Ending where **Process files based on a filename pattern** is selected
    - Route 1: Send To Partner

A clear text file is uploaded. Route 1 is triggered. The PGP Decryption transformation is skipped since **Require Encryption** is not selected. The clear text file is published successfully. A file is uploaded that does not match the configured filename pattern. Route 2 is triggered. The Line Ending transformation is skipped since the filename pattern was not recognized and routes the original file to partner transfer site.

Note PGP Decryption, Decompression, and Line Ending transformations have optional behaviors that enables them to pass along files that are not eligible for transformation.

Note Even though transformation steps could be skipped, it is highly recommended that the last step of every route will be a routing step (Send To Partner or Publish To Account). Only routing steps can route the transformed file outside of the sandbox folder.

Transformation on multiple files

The following Advanced Routing routes are configured:

- Route 1:
    - Transformation 1: Decompress
    - Transformation 2: PGP Decryption where **Require Encryption** is not selected
    - Routing 1: Publish To Account
- Route 2:
    - Routing 2: Send To Partner

A compressed file, containing both PGP encrypted and clean files, is uploaded. Route 1 is triggered. Transformation 1 decompresses the file and multiple files are passed on to Transformation 2. The PGP encrypted files are decrypted and the clean files are left as they are. The PGP decrypted and clean files are routed to the Publish to Account destination. Route 2 is triggered and originally uploaded compressed file is pushed to the partner transfer site.

Note If **Proceed with route execution on step failure** is not selected for the PGP Decryption (Route 1, Transformation 2) and PGP Decryption fails on any file, the whole transformation chain is considered failed. Any further operations within the route are not executed. Any successfully transformed files (until the failure point) are left on the file system. Any files that are in the sandbox folder are lost. Only files published by Send To Partner or Publish To Account route steps are available.

Route failure

The following Advanced Routing routes are configured:

- Route 1:
    - Transformation 1: Decompress
    - Transformation 2: PGP Decryption where **Require Encryption** is selected and **Proceed with route execution on step failure** is not selected
    - Routing 1: Publish To Account
- Route 2:
    - Routing 1: Send To Partner

A compressed file, containing both PGP encrypted and clean files, is uploaded. Route 1 is triggered. Transformation 1 decompresses the file and multiple files are passed on to Transformation 2. The PGP encrypted files are decrypted but since some of the files are in plain text the step execution fails (because **Require Encryption** is selected). Since **Proceed with route execution on step failure** is not selected the whole transformation chain is considered failed and any further processing within Route 1 is not executed, which means that Routing 1 (Publish To Account) is not executed (even though there are successfully decrypted files). Although other routes (if any) defined in the chain are executed. That's why Route 2 is triggered and the originally uploaded compressed file is pushed to the partner site.

Note By default the Send To Partner and Publish To Account steps are configured to be processed even though one or more files have failed to be sent or published, though all transformation steps (PGP Encryption, PGP Decryption, and so forth) are configured to fail even if only one file transformation has failed.

Note If route execution fails, any successfully transformed files (until the failure point) are left in the sandbox folder. Any files that are in the sandbox folder are lost. Only files published by the Send To Partner or Publish To Account route steps are available.

Transformed file as the input to the next step

The following Advanced Routing routes are configured:

- Route 1:
    - Transformation 1: Line Ending where **Rename output files** is not selected
    - Routing 1: Publish To Account

A text file is uploaded. Route 1 is triggered. Transformation 1 performs a Line Ending transformation over the file. The name of the file is not modified because **Rename output files** is not selected and the Line Ending transformation does not automatically modify the file name. The transformed file is the input for the next step, that's why the transformed file is routed to the Publish To Account destination.

Note When **Rename output files** is selected, the transformed file is renamed in the sandbox folder only. The original file in the subscription folder is not transformed.

Routing to multiple transfer sites

The following Advanced Routing route is configured:

- Route 1:
    - Routing 1: Send To Partner where multiple transfer sites belonging to a single account are selected

A file is uploaded. Route 1 is triggered. The file is sent to all transfer sites. If any of the transfers temporarily fail, only the failed transfers are automatically retried. There are no retries if the transfer permanently fails.

Note When routing to multiple transfer sites, it is recommended the **Route file as** be set using an Expression Language expression, containing `${timestamp}` or `${random()}` so it's transformed to an unique value for each of the transfers.

Note Unlike automatic retries when an ordinary subscription folder is used to send files directly to a transfer site, with the Advanced Routing feature the administrator is not able to control the automatic retries using *Cancel* or *Resubmit* buttons.

Custom Expression Language functions and variables

The following topics provide detailed lists of the custom Expression Language functions and variables that can be used within the Advanced Routing file processing.

Session related EL for AR

The following table provides the session related EL expressions:

Agent Env Variable	Routing EL expression	Example
DXAGENT_ PROTOCOL	session.protocol	${session.protocol eq 'http'} - returns true
DXAGENT_PWD=/	session.workDir	${concat (transfer.targetDir.substring (0,1), leadingFolder (session.workDir)) eq transfer.targetDir} - returns true
DXAGENT_PWD_ RESOLVED	session.workDirFull	${session.workDirFull.substring(13,15) eq account.businessUnit.name} - returns true
DXAGENT_ REMOTEADDR	session.remoteAddress	${session.remoteAddress eq session.remoteHost}
DXAGENT_ REMOTEHOST	session.remoteHost	${session.remoteHost.matches ('10.*')}
DXAGENT_ CLIENT	session.streamingClient	${session.streamingClient eq 'httpd'} ${extract (session.streamingClient,'d', 1) eq session.protocol}
DXAGENT_ SECURE_DATA	session.isSSL	${session.isSSL} ${!session.isSSL}
DXAGENT_TYPE	session.transferDirection	The direction of the transfer configuration. Values: • 0 indicates a transfer from an account to the application. • 1 indicates a transfer from the application to an account.
DXAGENT_ TIMESTAMP_ OUTGOING_END	session.timestampOutgoingEnd	${session.timestampOutgoingEnd} - the timestamp for events with outgoing type and trigger end.

Agent Env Variable	Routing EL expression	Example
DXAGENT_LOGFILENAME	session.logFileName	${session.logFileName} - the log file name. This will be used by runas utility on Unix to redirect stderr.
DXAGENT_EDGEID	session.edgeId	${session.edgeId} - the identifier of the current SecureTransport Edge. The Edge identification string is set in the protocol server's configuration file(s).
DXAGENT_SUBSCRIPTION_FOLDER	session.subscriptionFolder	${session.subscriptionFolder} - the subscription folder in the form of a POSIX-style path relative to the user home directory. This value represents the client path.
DXAGENT_APPLICATION_TYPE	session.applicationType	${session.applicationType} - a string that identifies application type.
DXAGENT_APPLICATION_NAME	session.applicationName	${session.applicationName} - the name of the application instance.
DXAGENT_APPLICATION_NOTES	session.applicationNotes	${session.applicationNotes} - notes associated with the application instance.
DXAGENT_SITE_ATTR_DOWNLOAD_FOLDER	session.siteDownloadFolder	${session.siteDownloadFolder}- directory location on the remote server to check for files to download. The files to be downloaded is determined by DXAGENT_SITE_ATTR_DOWNLOAD_PATTERN.
DXAGENT_SITE_ATTR_DOWNLOAD_PATTERN	session.siteDownloadPattern	${session.siteDownloadPattern} - the download pattern used to determine which files should be downloaded from the remote server. It's applied to the names of the files found in Download Folder.
DXAGENT_SITE_ATTR_UPLOAD_FOLDER	session.siteUploadFolder	${session.siteUploadFolder} - the upload folder specifies the directory on the remote server where the uploaded files are placed.

Agent Env Variable	Routing EL expression	Example
DXAGENT_SITE_ATTR_USERNAME	session.siteUsername	${session.siteUsername} - the username presented to the remote server for authentication; optional Site attribute.
DXAGENT_SITE_ATTR_HOST	session.siteHost	${session.siteHost} - the remote host represented by the site. If absent, the site does not establish a connection to the remote host. An example of that is the Folder Monitor site.

Predefined EL functions for AR

The following table lists examples of predefined EL functions and descriptions. To check additional available methods for string manipulation, refer to the *Open JDK Documentation*.

Syntax	Description
${variable.trim()}	Removes whitespace from both ends of a string and returns a new string.
${variable.length()}	Returns the number of code units in the string.
${variable.matches()}	Returns `true` if the string matches the given regular expression, otherwise returns `false`.
${variable.replaceAll()}	Replaces each substring of the string that matches the given regular expression with the given replacement.
${variable.toUpperCase()}	Converts a given string to upper case only symbols.
${variable.toLowerCase()}	Converts a given string to lower case only symbols.
${variable.substring (beginIndex,endIndex)}	Returns a substring for a given string. The substring begins at the specified `beginIndex` and extends to the character at index `endIndex` - 1. Thus the length of the substring is `endIndex-beginIndex`.
${extract(variable, delimiter, position)}	Splits a given string to tokens based on a delimiter.
${leadingFolder(path)}	For a given directory/file path returns only the leading one.

Syntax	Description
`${parentFolder(path)}`	For a given directory/file path returns the parent folder path.
`${dayOffset(format, offset)}`	Returns a date representing today's date with the offset of the days parameter.

The following table lists predefined EL variable functions and Advanced Routing examples.

Syntax	Advanced Routing Usage
`${variable.trim()}`	`${transfer.target.trim()}` - returns '`utf8-all.txt`' if the original file name was '*utf8-all.txt*'
`${variable.length()}`	`${transfer.target.length()}` - returns `12` if the original file name was *filename.txt*
`${variable.matches()}`	`${transfer.target.matches("(.*json)")}` - returns `true` if the original file name was *example.json*
`${variable.replaceAll()}`	`${transfer.target.replaceAll("e","3")}` - - returns `3xampl3.txt` if the original file name was *example.txt*
`${variable.toUpperCase()}`	`${transfer.target.toUpperCase()}` - returns `EXAMPLE.TXT` if the original file name was *ExamplE.txt*
`${variable.toLowerCase()}`	`${transfer.target.toLowerCase()}` - returns `example.txt` if the original file name was *EXAMPLE.txt*
`${variable.substring(beginIndex,endIndex)}`	`${transfer.target.substring(0,5)}` - returns `examp` if the original file name was *example.txt*
`${extract(variable, delimiter, position)}`	`${extract('payroll_Axway_21457584375.txt', '_', 2)}` returns `Axway`

`${leadingFolder(path)}`	`${leadingFolder('/opt/TMWD/st51')}` - returns `'opt'`
	`${leadingFolder('/opt')}` - returns `'opt'`
	`${leadingFolder('/')}` - returns `'/'`

`${parentFolder(path)}`	`${parentFolder('/opt/TMWD/st51')}` - returns `'/opt/TMWD'`
	`${parentFolder('/')}` - returns `'/'`
	`${parentFolder('/usr/file.txt')}` - returns `'/usr'`

Syntax	Advanced Routing Usage
`${dayOffset(format, offset)}`	`${dayOffset('yyMMdd', '-5')}` - returns 10th if today is 15th of August formatted as per the specified format parameter - `120810`. `${dayOffset('yyMMdd', '+7')}` - returns 22nd if today is 15th of August formatted as per the specified format parameter - `120822`. `${dayOffset('ddMMyy', '+1')} ge '090414'}`

Account related EL for AR

The following table provides the account related EL expressions.

Agent Env Variable	Routing EL expression	Example
`DXAGENT_ACCOUNT_ATTR_*`	`account.attributes['ATTRIBUTE_NAME']`	`${account.attributes['templateClass'].toUpperCase() eq 'AdClass'.toUpperCase()}` - returns true `${account.attributes['transfersWebServiceAllowed']}` - returns either false or true
`DXAGENT_ACCOUNT_DELIVERY_METHOD`	`account.deliveryMethod`	`${account.deliveryMethod.toLowerCase() == 'custom' }`
`DXAGENT_ACCOUNT_DISABLED`	`account.disabled`	`${account.disabled != '0'}` - returns true
`DXAGENT_ACCOUNT_EMAIL`	`account.email`	`${!empty account.email}`
`DXAGENT_ACCOUNT_ENROLLMENT`	`account.enrollment`	`${account.enrollment.toLowerCase() eq 'existing_account'}`
`DXAGENT_ACCOUNT_HTMLTEMPLATE`	`account.htmlTemplate`	`${account.htmlTemplate.substring(11,14) eq 'sm6'}`

Agent Env Variable	Routing EL expression	Example
DXAGENT_ACCOUNT_ID	account.id	${ !empty account.id}
DXAGENT_BUSINESS_UNIT_NAME	account.businessUnit.name	${account.businessUnit.name eq 'bu'} - returns true
DXAGENT_BUSINESS_UNIT_ID	account.businessUnit.id	${ !empty account.businessUnit.id}
DXAGENT_ACCOUNT_NAME	account.name	${account.name eq 'template-routes'} - returns true
DXAGENT_ACCOUNT_NOTES	account.notes	${ !empty account.notes}
DXAGENT_ACCOUNT_PHONE	account.phone	${ !empty account.phone}
DXAGENT_ACCOUNT_TYPE	account.type	${account.type eq 'template'} ${account.type != 'service'}
DXAGENT_HOMEDIR	account.home	${parentFolder (transfer.targetDirFull) eq account.home}
DXAGENT_ACCOUNT_IMPLICIT_ENROLLMENT	account.implicitEnrollment	${account.implicitEnrollment.toLowerCase().matches('.*account')}
	account.attributes ['userVars.xxx']	Access additional attribute of an account with name xxx.

LDAP related EL for AR

The following table provides the LDAP related EL expressions.

Agent Env Variable	Routing EL expression	Example
STSESSION_LDAP_DOMAIN_ID	ldap.domainId	${ldap.domainId == '<domain id>'}
STSESSION_LDAP_DOMAIN_NAME	ldap.domainName	${ldap.domainName eq 'ad'} - will return true
STSESSION_LDAP_DN	ldap.dn	${ldap.dn.toLowerCase().matches('.*dc=st1.*')} - will return true
STSESSION_LDAP_AUTH_BY_EMAIL	ldap.authByEmail	${ldap.authByEmail gt 0}
STSESSION_LDAP_DIR.*	ldap.attributes.*	
STSESSION_LDAP_DIR_mail	ldap.attributes.mail	${ldap.attributes.mail.matches('usert.*') ? 1 : 0} - will return 1

PeSIT related EL for AR

The following table provides the PeSIT related EL expressions.

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_FILE_DESTINATION	pesit.file.destination	Provides the file destination.
DXAGENT_PESIT_FILE_FILENAME	pesit.file.filename	Provides the file name.
DXAGENT_PESIT_FILE_ORIGINATOR	pesit.file.originator	Provides the originator of the file.
DXAGENT_PESIT_FILE_RECEIVER	pesit.file.receiver	Provides the receiver for the file.
DXAGENT_PESIT_FILE_SENDER	pesit.file.sender	Provides the sender of the file.
DXAGENT_PESIT_FILE_FILETYPE	pesit.file.filetype	Provides a description of the file type.

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_FILE_TRANSFERID	pesit.file.transferID	Provides the transfer ID of the file.
DXAGENT_PESIT_PI_CRC	pesit.pi.crc	Provides the cyclic redundancy check (CRC) parameters in the message body. Example: Outgoing ${!pesit.pi.crc}
DXAGENT_PESIT_PI_DIAGCODE	pesit.pi.diagCode	Provides the diagnostic code parameters in the message body. Example: Outgoing ${pesit.pi.callerID.toLowerCase() eq account.user.loginName}
DXAGENT_PESIT_PI_senderID	pesit.pi.senderID	Provides the sender identification parameters in the message body. Example: ${pesit.pi.senderID.toLowerCase() eq 'target'}
DXAGENT_PESIT_PI_receiverID	pesit.pi.receiverID	Provides the receiver identification parameters in the message body. Example: ${pesit.pi.receiverID.toLowerCase() eq account.name}
DXAGENT_PESIT_PI_callerPassword	pesit.pi.callerPassword	Provides the caller password parameters in the message body.
DXAGENT_PESIT_PI_serverPassword	pesit.pi.serverPassword	Provides the server password parameters in the message body.
DXAGENT_PESIT_PI_version	pesit.pi.version	Provides the version parametes in the message body. Example: Outgoing ${pesit.pi.version == 2}

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_PI_exchangeBufferSize	pesit.pi.exchangeBufferSize	Provides the exchange buffer size parameters in the message body. Example: ${!empty pesit.pi.exchangeBufferSize}
DXAGENT_PESIT_PI_totalRecords	pesit.pi.totalRecords	Provides the number of total records parameter in the message body.
DXAGENT_PESIT_PI_file Organization	pesit.pi.fileOrganization	Provides the file organization parameter in the message body. Example: ${pesit.pi.fileOrganization == 0}
DXAGENT_PESIT_PI_recordLength	pesit.pi.recordLength	Provides the record length parameter in the message body. Example: ${pesit.pi.recordLength == 2048}
DXAGENT_PESIT_PI_keyLength	pesit.pi.keyLength	Provides the key length parameter in the message body
DXAGENT_PESIT_PI_allocationUnit	pesit.pi.allocationUnit	Provides the number of allocation units parameter in the message body. Example: ${pesit.pi.allocationUnit == 0}
DXAGENT_PESIT_PI_creationDateTime	pesit.pi.creationDateTime	Provides the creation date and time parameter in the message body. Example: ${!empty pesit.pi.creationDateTime}
DXAGENT_PESIT_PI_originalSenderID	pesit.pi.originalSenderID	Provides the original sender identification parameter in the message body. *Example:* Store ${pesit.pi.originalSenderID eq 'CFT1'}

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_PI_msgData	pesit.pi.msgData	Provides the message data parameter in the message body. If used in an External Script step, it must not contain parentheses () or double quotes ".
DXAGENT_PESIT_PI_checkPointInterval	pesit.pi.checkPointInterval	Provides the check point interval parameter in the message body. *Example:* `${pesit.pi.checkPointInterval == 1024}`
DXAGENT_PESIT_PI_checkPointWindow	pesit.pi.checkPointWindow	Provides the check point window parameter in the message body. *Example:* `${pesit.pi.checkPointWindow == 4}`
DXAGENT_PESIT_PI_fileType	pesit.pi.fileType	Provides the file type parameter in the message body. Example: `${pesit.pi.fileType == 0}`
DXAGENT_PESIT_PI_fileName	pesit.pi.fileName	Provides the file name parameter in the message body. Example: `${pesit.pi.fileName.toLowerCase() eq 'idf'}`
DXAGENT_PESIT_PI_transferID	pesit.pi.transferID	Provides the transfer identification parameter in the message body. Example: `${!empty pesit.pi.transferID}`
DXAGENT_PESIT_PI_fileAttributes	pesit.pi.fileAttributes	Provides the file attributes parameter in the message body.
DXAGENT_PESIT_PI_restart	pesit.pi.restart	Provides the restart parameter in the message body. *Example:* Outgoing `${!pesit.pi.restart}`

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_PI_dataEncoding	pesit.pi.dataEncoding	Provides the data encoding parameter in the message body. *Example:* ${pesit.pi.dataEncoding lt 3}
DXAGENT_PESIT_PI_priority	pesit.pi.priority	Provides the priority parameter in the message body. *Example:* ${pesit.pi.priority == 1}
DXAGENT_PESIT_PI_restartCheckPoint	pesit.pi.restartCheckPoint	Provides the restart check point parameter in the message body.
DXAGENT_PESIT_PI_cancelCode	pesit.pi.cancelCode	Provides the cancel code parameter in the message body.
DXAGENT_PESIT_PI_checkPointNumber	pesit.pi.checkPointNumber	Provides the check point number parameter in the message body.
DXAGENT_PESIT_PI_compressed	pesit.pi.compressed	Provides the compression parameter in the message body. *Example:* Outgoing ${!pesit.pi.compressed}
DXAGENT_PESIT_PI_compressionType	pesit.pi.compressionType	Provides the compression type parameter in the message body. Example: ${pesit.pi.compressionType eq 3}
DXAGENT_PESIT_PI_accessType	pesit.pi.accessType	Provides the access type parameter in the message body.
DXAGENT_PESIT_PI_resyncAllowed	pesit.pi.resyncAllowed	Provides the resync allowed parameter in the message body. *Example:* Outgoing ${pesit.pi.resyncAllowed == 0}
DXAGENT_PESIT_PI_totalBytes	pesit.pi.totalBytes	Provides the total bytes parameter in the message body.

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_PI_diagnosticText	pesit.pi.diagnosticText	Provides the diagnostic text parameter in the message body.
DXAGENT_PESIT_PI_recordFormat	pesit.pi.recordFormat	Provides the record format parameter in the message body. *Example:* `${pesit.pi.recordFormat eq 128}`
DXAGENT_PESIT_PI_fileLabel	pesit.pi.fileLabel	Provides the file label parameter in the message body. *Example:* `${pesit.pi.fileLabel eq transfer.target}`
DXAGENT_PESIT_PI_keyOffset	pesit.pi.keyOffset	Provides the key offset parameter in the message body.
DXAGENT_PESIT_PI_allocationSize	pesit.pi.allocationSize	Provides the allocation size parameter in the message body. *Example:* `${pesit.pi.allocationSize == 195}`
DXAGENT_PESIT_PI_extractionDateTime	pesit.pi.extractionDateTime	Provides the extraction date and time parameter in the message body.
DXAGENT_PESIT_PI_finalDestinationID	pesit.pi.finalDestinationID	Provides the final destination identification parameter in the message body. *Example:* Store and Forward `${pesit.pi.finalDestinationID eq 'CFT2'}`
DXAGENT_PESIT_PI_serviceParam	pesit.pi.serviceParam	Provides the service parameter in the message body. *Example:* `${pesit.pi.serviceParam eq 'X'}`

Routing related EL for AR

The following table provides the routing related EL expressions.

Routing EL expression	Description / example
`routing.routePackageId`	The ID of current Route Package. *Example:* `${routing.routePackageId.matches ('<id>')}`
`routing.routePackage SandboxFolder`	The master working directory of current execution of a Route Package; located in `${user_home_ dir}/.stfs/objects/${routePackageId (0,2)}/${routePackageId(3)}/;` . The directory is same during route recovery and it is persisted with the event. *Example:* `${!routing.routePackageSandboxFolder}`
`routing.executeRoute SandboxFolder`	The working directory of current Execute Route (it is a subdirectory of `${routing.routePackageSandboxFolder}`). *Example:* `${parentFolder (routing.executeRouteSandboxFolder) eq routing.routePackageSandboxFolder}`
`${routing.originalFiles}`	List of file paths being selected for processing from the Subscription folder.
`${routing.targetFiles}`	List of file paths to process in current Execute Route sandbox (these are copies of `${routing.originalFiles};` located in `${routing.executeRouteSandboxFolder}).`
`routing.triggeredWithoutPayload`	Determines if the Route was triggered without any file(s) available for processing. This is determined by the *Submit the transferred file(s) to the route for processing.* checkboxes on the Advanced Routing subscription page. Values: • `true` • `false`

Special routing EL variables for AR

The following table provides the special routing variables.

Variable	Description
`${currentfulltarget}`	Contains the path to the file in sandbox folder that is currently processed by a transformation or routing step. Used in PGP Encryption transformation steps and in Send to Partner and Publish To Account routing steps. Also, used in the Rename output file of Transformation steps. The expression is evaluated to the absolute file path of the file being processed.
`${transformedfilename}`	Denotes the name of current transformed file. Used mainly in PGP and Compression steps (Transformation steps). Used in the *Rename output file to* pane.
`${transferredfilename}`	Denotes the name of last transferred file. It can be the same as ${currentfulltarget} or changed by using "Route file as"/"Publish file as" settings. Used in Routing steps (Publish/SendToPartner) in *Post Routing Action Rename* pane.
`${transferredfilenames}`	Contains the list of names of currently transferred files by the Send To Partner step. Files which were not processed by the Send To Partner step (because of the File Filter criteria) are not on this list. This variable should be used only in the *Trigger file content* field.

STFS PeSIT related EL for AR

The following table provides the STFS PeSIT related EL expressions.

Agent Env Variable	Routing EL expression	Description
`pesitPIEnvVariables`		
`DXAGENT_PESIT_PI_senderID`	`${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_senderID']}` `${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_senderID']}`	Provides the sender identification parameter in the message body. • Example: `${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_senderID'] eq 'PARTNER-PESIT'}`

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_PI_recordLength	`${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_recordLength']}` `${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_recordLength']}`	Provides the record length parameter in the message body. • Example: `${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_recordLength'] eq '2048'}`
DXAGENT_PESIT_PI_allocationUnit	`${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_allocationUnit']}` `${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_allocationUnit']}`	Provides the allocation unit parameter in the message body. • Example: `${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_allocationUnit'] eq '0'}`
DXAGENT_PESIT_PI_fileName	`${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_fileName']}` `${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_fileName']}`	Provides the file name parameter in the message body. • Example: `${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_fileName'].matches('IDF.*')}`
DXAGENT_PESIT_PI_serviceParam	`${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_serviceParam']}` `${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_serviceParam']}`	Provides the service parameter in the message body. *Examples:* • *No PI99* `${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_serviceParam'] eq null}` • With PI99 `${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_serviceParam'].matches('AloAlo')}`

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_PI_receiverID	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_receiverID']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_receiverID']}	Provides the receiver identification parameter in the message body. *Examples:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_receiverID'] eq 'U1'} ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_receiverID'].matches ('U.*')}
DXAGENT_PESIT_PI_priority	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_priority']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_priority']}	Provides the priority parameter in the message body. *Examples:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_priority']} - will be evaluated to {1} which is the priority value ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_priority']==1} - will be evaluated to true
DXAGENT_PESIT_PI_dataEncoding	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_dataEncoding']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_dataEncoding']}	Provides the data encoding parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_dataEncoding'] eq '1'}
DXAGENT_PESIT_PI_recordFormat	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_recordFormat']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_recordFormat']}	Provides the record format parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_recordFormat']==0}

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_PI_fileLabel	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_fileLabel']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_fileLabel']}	Provides the file label parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_fileLabel'].matches('file.*')}
DXAGENT_PESIT_PI_originalSenderID	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_originalSenderID']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_originalSenderID']}	Provides the original sender identification parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_originalSenderID'] eq null}
DXAGENT_PESIT_PI_finalDestinationID	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_finalDestinationID']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_finalDestinationID']}	Provides the final destination identification parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_finalDestinationID'] eq null}
DXAGENT_PESIT_PI_creationDateTime	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_creationDateTime']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_creationDateTime']}	Provides the creation date and time parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_creationDateTime'] gt '140611093656'}

Agent Env Variable	Routing EL expression	Description
DXAGENT_PESIT_PI_transferID	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_transferID']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_transferID']}	Provides the transfer identification parameter in the message body. *Example:* ${!empty stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_transferID']}
DXAGENT_PESIT_PI_compressionType	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_compressionType']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_compressionType']}	Provides the compression type parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_compressionType']==3}
DXAGENT_PESIT_PI_exchangeBufferSize	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_exchangeBufferSize']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_exchangeBufferSize']}	Provides the exchange buffer size parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_exchangeBufferSize'] lt '8192'}
DXAGENT_PESIT_PI_recordLength	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_recordLength']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_recordLength']}	Provides the record length parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_recordLength'] < 512}
DXAGENT_PESIT_PI_fileOrganization	${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_recordLength']} ${stfs.attributes['pesitPIEnvVariables']['DXAGENT_PESIT_PI_recordLength']}	Provides the file organization parameter in the message body. *Example:* ${stfs.attributes.pesitPIEnvVariables['DXAGENT_PESIT_PI_fileOrganization'] eq '0'}

Agent Env Variable	Routing EL expression	Description
recordsLength		
	${stfs.attributes.recordsLength} ${stfs.attributes['recordsLength']}	Provides the record length parameter in the message body. *Example:* ${stfs.attributes['recordsLength'].matches('[I@.*')}
endOfLineSymbol		
	${stfs.attributes.endOfLineSymbol} ${stfs.attributes['endOfLineSymbol']}	Provides the end of line symbol parameter in the message body. *Example:* ${stfs.attributes['endOfLineSymbol'].matches('n.*')}

Transfer related EL for AR

The following table provides transfer related EL expressions:

Agent Env Variable	Routing EL expression	Example
DXAGENT_CORE_ID	transfer.coreId	${transfer.coreId eq "390bedec-c82e-45aa-afc2-1b78d846732d"}
DXAGENT_TARGETPATH	transfer.targetDirFull	${parentFolder(transfer.targetDirFull) eq account.home}
DXAGENT_TRANSFERRED_BYTES	transfer.transferredBytes	${transfer.transferredBytes ge 20}
DXAGENT_TRANSFER_STATUS_START_TIME	transfer.startTime	${transfer.startTime lt transfer.endTime}
DXAGENT_TIMESTAMP_INCOMING_END	transfer.endTime	${transfer.endTime gt transfer.startTime}

Agent Env Variable	Routing EL expression	Example
DXAGENT_XFERTYPE	transfer.xferType	${transfer.xferType eq "A"} ${transfer.xferType eq "I"}
DXAGENT_TARGETDIR	transfer.targetDir	${concat (transfer.targetDir.substring (0,1), leadingFolder (session.workDir)) eq transfer.targetDir} - returns true
DXAGENT_FULLTARGET	transfer.targetFull	${filename (transfer.targetFull).matches ('part.*\\.crt')} - returns true ${extension(transfer.target) eq extension(filename (transfer.targetFull))} - returns true
DXAGENT_TARGET	transfer.target	${transfer.target.matches ('.*\\.crt') ? 1 : 0} - will return 1 ${extract(basename (transfer.target),'_',1) eq 'partner1'} - will return true ${basename (transfer.target).replace(' (.*)_(.*)', '$2_$1') eq 'certificate_partner')

Additionally, the following transfer related EL expressions can be used for Advanced Routing:

Routing EL expressions	Description
`transfer.trigger`	Determines the transfer trigger. Values: • `server_pull` - If the routing was started by a server-initiated pull. • `client_download` - If the routing was started by a client download. • `client_upload` - If the routing was started by a client upload. • `wildcard_pull` - If the routing was started by a wildcard pull. • `ack` - If the routing should start upon receiving a positive acknowledgment (automatic or manual). • `nack` - If the routing should start upon receiving a manual negative acknowledgment. • `pesit_message` – If the routing should start upon receiving a PeSIT message.
`transfer.status`	Determines the transfer status. Values: • `success` - If the routing was started by a success. • `failure` - If the routing was started by a failure. • `temporary_failure` - If the routing was started by a temporary failure. • `empty` - If the routing was started by a wildcard pull and no files matching the download pattern were found.

Transfer related expressions can be used together.
Example: The routing is triggered when the listing is successful but does not return any files, on days that are NOT Monday or Tuesday.

```
${transfer.trigger eq 'wildcard_pull' and transfer.status eq
'empty' and (date("EEE") ne 'Mon' and date("EEE") ne 'Tue')}
```

Trigger related EL for AR

The following table provides the trigger related EL expressions.

Routing EL expression	Example
`string getFileContent (string filePath, int beginIndex, int length, string charset)`	Returns a string that is a sub-string of the file's content. *Example:* When the file contains `1234567890` the `getFileContent (file, 1, 3, "UTF8")` will return `234`.
`string getFileContentTail (string filePath, int beginIndex, int length, string charset)`	Returns a string that is a sub-string of the file's content. The reading of the file begins at the end. *Example:* When the file contains `1234567890` the `getFileContentTail (file, 1, 3, "UTF8")` will return `789`.
`byte[] getFileContentBytes (string filePath, int offset, int length)`	Reads up to the specified number of bytes of data starting from a specified offset into an array of bytes from the beginning of the file. An attempt is made to read as many bytes as possible, but a smaller number may be read.
`byte[] getFileContentBytesTail (string filePath, int offset, int length)`	Reads up to the specified number of bytes of data from a specified offset into an array of bytes starting from the end of the file. An attempt is made to read as many bytes as possible, but a smaller number may be read.

The functions which return a string can be used in Expressions and Predicates for route triggering, as well as in other fields in which the expression language is supported.

The functions which return bytes cannot be used in Expressions and Predicates for route triggering. The extraction of file is used for file content composition (for example, trigger file content in a Send To Partner route step).

Note When functions which return bytes are used in trigger file content, they cannot be combined with string functions, because bytes will be written.

Note The `filePath` parameter must contain an absolute path and should be written under home folder of the user who is triggering the corresponding route.

User related EL for AR

The following table provides the user related EL expressions.

Agent Env Variable	Routing EL expression	Example
DXAGENT_LOGINNAME	account.user.loginName	${account.user.loginName.matches('.*usert.*') ? 1 : 0} - will return 1
DXAGENT_USERTYPE	account.user.type	${account.user.type.matches ('virtual') } - will return true
DXAGENT_USERCLASS	account.user.className	${account.user.class eq 'AdClass'}
DXAGENT_ NATIVEUSER	account.user. nativeUserName	
DXAGENT_USERGID	account.user.uid	${account.user.uid gt '1000'}

HTTP headers related EL for AR

The following table provides the HTTP header EL expressions.

Agent Env Variable	Routing EL expression	Example
DXAGENT_HTTP_*	http.headers. HEADER_NAME	
DXAGENT_HTTP_ HOST	http.headers. HOST	${http.headers.HOST.matches('.*')}
DXAGENT_HTTP_ USER_AGENT	http.headers. USER_AGENT	${extract(http.headers.USER_ AGENT,'/',1) eq 'Mozilla'}
DXAGENT_HTTP_ REFERER	http.headers. REFERER	${concat(concat ('https://',http.headers.HOST),':444'). matches(http.headers.REFERER)}
DXAGENT_HTTP_ CONTENT_LENGTH	http.headers. CONTENT_LENGTH	${http.headers.CONTENT_LENGTH eq transfer.transferredBytes}
DXAGENT_HTTP_ CONTENT_TYPE	http.headers. CONTENT_TYPE	

Agent Env Variable	Routing EL expression	Example
DXAGENT_HTTP_ACCEPT_LANGUAGE	http.headers.ACCEPT_LANGUAGE	
DXAGENT_HTTP_FEATURES	http.headers.FEATURES	
DXAGENT_HTTP_CONTENT_RANGE	http.headers.CONTENT_RANGE	
DXAGENT_HTTP_ACCEPT	http.headers.ACCEPT	
DXAGENT_HTTP_ORIGIN	http.headers.ORIGIN	
DXAGENT_HTTP_CONNECTION	http.headers.CONNECTION	
DXAGENT_HTTP_ACCEPT_ENCODING	http.headers.ACCEPT_ENCODING	
DXAGENT_HTTP_CONTENT_DISPOSITION	http.headers.CONTENT_DISPOSITION	

Troubleshoot Advanced Routing

This topic outlines the specific troubleshooting steps for Advanced Routing as well as some general troubleshooting approach recommendations.

The following troubleshooting procedures are provided:

- *General troubleshooting steps* on page 1028
- *Debug logging* on page 1028
- *Logging into a file* on page 1
- *Exceptional case: absolute path to sandbox folder in EL expressions* on page 1030
- *Advanced Routing fails with the sandbox and user home folders on the same CIFS share* on page 1028

General troubleshooting steps

Should you encounter any problems with Advanced Routing, take the following steps:

1. Check if the observed behavior is listed in the *Advanced Routing best practices* on page 999 topic.
2. Check if the observed issue is listed in the *Known Issues* topic of the *SecureTransport Release Notes*.
3. Restart the failing sever(s) or client(s).
4. Reproduce the issue with Debug Logging enabled if applicable. Refer to the *Debug logging* on page 1028 topic for instructions.
5. Collect the debug log files, screen shots, or any other data related to the issue and contact Axway Support at support.axway.com.

Debug logging

SecureTransport stores log messages in the SecureTransport database. The contents of log messages can be viewed on the *Server Log* page. To access and view the *Server Log* page, select **Operations > Server Log**. For complete information on viewing, searching, and exporting logs, refer to *Server log* on page 322.

Advanced Routing fails with the sandbox and user home folders on the same CIFS share

Problem summary: With SecureTransport deployments on Linux, Advanced Routing fails when the sandbox is located on the same CIFS share as the user's home folder.

Problem details:

1. You set the user's home folder to be on a CIFS share.
2. You set an absolute path value for the sandbox location in the '`AdvancedRouting.sandboxFolderLocation`' server configuration option to point to the same CIFS share.
3. When you attempt to execute an Advanced Routing configuration, you get errors in the Server Log.

Solution:

1. Make sure symbolic links are enabled.
2. Mount the CIFS share following the example:

   Enter the following command to run the Axway Installer:

   ```
   mount -t cifs -o
   ```

```
username=<Administrator>,password=<password>,file_
mode=0777,dir_mode=0777,mfsymlinks //<IP_
address>/Shared/<user>/<home_folder>
```

AR fails while copying input files to sandbox

Problem: In a cluster environment where one node uploads files and another node processes Advanced Routing, a route could fail with a transient error.

Possible cause: Once a file is uploaded to an AR subscription folder, it is copied from the subscription folder to the sandbox folder. If the shared storage synchronization does not happen on time, the second node does not "see" the file in the subscription folder uploaded on another node. As a result, a "no such file" error occurs and the route fails while copying the file to the sandbox folder.

Solution: In this case, retrying the failed operation more times or/and for longer intervals may solve the issue. SecureTransport uses an exponential backoff algorithm for retries: it increases the waiting time between the consecutive retries after each retry failure following the logic:

The first retry is after `(RetryNumber -1)` x `retryTime`, the second retry is after `RetryNumber` x `retryTime`, the third after 2 x `retryTime` and so on until the configured number of `retries` is reached.

where

- `retries` is the maximum number of retry attempts for failed AR file processing operations configured in the `com.axway.st.server.fs.ar.file.processing.retries` parameter (10 by default).
- `retryTime` is the delay of the first retry and the growth rate between attempts configured in the `com.axway.st.server.fs.ar.file.processing.retryTime` parameter. (100 ms by default).

With the default values, SecureTransport retries 10 times: the first retry happens immediately, the second after 100 ms, the third after 200 ms, and so on. This sets a deadline of 4.5 seconds for SecureTransport to retry a failed AR file processing operation.
The AR retry logic is enabled by default. When active, it prints WARN messages in the Server Log like that look like this one:

```
NoSuchFileException: (will retry in : 0 ms; retries: 1/10)
srcPath:/<AccountSubscriptionFolder>/<FileName>
```

If it fails with "Error while copying input file(s) to sandbox environment" after all the retries are exhausted, you may need to increase the AR retry defaults.
To do so, edit the `<FILEDRIVEHOME>/conf/STStartScriptsConfig` file to add the following lines:

```
TM_JAVA_OPTS="-Dcom.axway.st.server.fs.ar.file.processing.retries=20 $TM_JAVA_OPTS"
TM_JAVA_OPTS="-Dcom.axway.st.server.fs.ar.file.processing.retryTime=200 $TM_JAVA_
OPTS"
```

Edit the values to fit your needs. In the example, we limit the number of retries to 20 and set the delay to 200 ms. Therefore, the retries happen in 0, 200, 400, 600,...., 3800 ms., and the total amount of time for retrying is 38 seconds. Note that the benefits of changing these parameters are dependent on the shared storage type and its performance.

Exceptional case: absolute path to sandbox folder in EL expressions

Problem summary: With SecureTransport deployments on Windows Server, attempts to use a configured absolute path to sandbox location in an EL expression may fail and return an error.

Problem details:

1. You set an absolute path value for the sandbox location in the '`AdvancedRouting.sandboxFolderLocation`' server configuration option.
2. You configure use of EL expressions: set the '`AdvancedRouting.sandboxFolderLocation.expressionLanguage`' configuration option to '`true`'.
3. When you attempt to use an expression (for example'`\\<IP_address>\Shared\sandbox\${env['DXAGENT_ACCOUNT_TYPE']}`'), you might get errors in the Server Log.

Possible cause: The issue is configuration based and relates to the allowed remote storage for 'Users' to create symbolic links (configurable in 'secpol.msc' where 'Users' must be added in - *Security Settings > Local Policies > User Rights Assignment > 'Create symbolic links'*). The problem occurs only in the case when the user's home folder and the sandbox custom folders are both set to one or more remote storage machines.

Solution: This problem might occur because by default remote to remote symbolic links are disabled. You can enable it with `fsutil`.

C:\Windows\system32>fsutil behavior query SymlinkEvaluation

Local to local symbolic links are enabled.

Local to remote symbolic links are enabled.

Remote to local symbolic links are disabled.

Remote to remote symbolic links are disabled.

C:\Windows\system32>fsutil behavior set SymlinkEvaluation R2R:1 C:\Windows\system32>fsutil behavior query SymlinkEvaluation

Local to local symbolic links are enabled.

Local to remote symbolic links are enabled.

Remote to local symbolic links are disabled.

Remote to remote symbolic links are enabled.

Be sure to run fsutil from elevated command prompt.

Configuring asynchronous MDN receipts with AS2 transfers

MDN (Message Disposition Notification) receipts serve to warrant data integrity and non-repudiation in the underlying protocol, in this case AS2. Asynchronous AS2 MDN receipts are communicated in separate request from transfers, and this behavior requires specific setup for correct processing in advanced routes. It is possible to use Advanced Routing (AR) for AS2 file transfers using asynchronous MDN receipts but the available options require special out-of-the-box approaches which serve as *workarounds*. To better understand the logic of each workaround, it is important to understand where the concepts in AR and asynchronous exchange of MDN receipts differ and where they intercept.

AS2 MDN receipts overview

The AS2 protocol requires a bi-directional partnership to be created: you must define the partner's server on your side, and the partner defines your server on theirs. This creates several use cases with this partnership and the AS2 MDN receipts configuration differs depending on the actual usage.

It is possible to use the partnership only in one direction - to send files to a partner or receive files from them (files that partner sends you). In this case the AS2-MDN is configured only for one side of the partnership.

For two-way, bi-directional communication between partners, the asynchronous AS2-MDN receipts have to be configured on both ends, for both inbound and outbound directions.

The asynchronous AS2-MDN receipt is sent in a separate HTTP or HTTPS TCP/IP connection and is similar to an inbound transfer from a partner.

SecureTransport Applications for AS2 transfers

In SecureTransport, the setup that handles AS2 transfers can be an instance of Site Mailbox, the Basic Application or the Advanced Routing application types. A user account must be subscribed to either of these applications to be able to send / receive files over the AS2 protocol.

AR uses Route setup for outbound transfers

With Advanced Routing, the configuration for outbound transfers is part of the Routes, not the Subscription. This presents a problem with AS2 transfers and the asynchronous MDN receipts due to the way SecureTransport relates the received receipts to the transfers: via the Subscription.

In the cases with e.g. Site Mailbox and Basic Application, the Subscription holds the configuration for the outbound transfers, so when SecureTransport receives a MDN, it can relate to the proper subscription and validate the MDN. With Advanced Routing, however, when a file is sent to partner via AS2 in a Route, and the MDN is returned by the partner, SecureTransport is unable to locate the Subscription the MDN belongs to and throws an error:

```
org.openas2.partner.PartnershipNotFoundException: AS2 site <AS2-sitename>
is not used for sending in any subscription.
```

This is a direct consequence of the design of the Advanced Routing. However, this behavior can be fixed using special scenarios in which you can combine AR with other applications to achieve AS2 transfers when asynchronous MDN receipts are required.

- Scenario 1: Combine AR with Basic application (outbound transfers only)
- Scenario 2: Combine AR with two Basic application instances (one for inbound and one for outbound transfers)
- Scenario 3: Combine AR with both SiteMailbox application (for both inbound and outbound transfers)

Scenario 1: Combine AR and Basic application for outbound transfers

This approach uses a combination of an Advanced Routing instance and one Subscription to the Basic Application. Configure Advanced Routing to receive files and send them to an AS2 Site. A dummy Basic Application will be added (and subscribed to a different folder) to facilitate correct MDN processing for outbound transfers in AR. In essence, the Advanced Routing application will be used to send the files, while the Basic Application will only be used for receiving the asynchronous MDN receipts.

Setup specifics

With this setup, you must configure:

- One AS2 Transfer Site (in the user account).
- An Advanced Routing instance with a Route that uses an AS2 Transfer Site to send the files in a Send To Partner step. In the subscription to this application, you must set that AS2 Transfer Site in the **Automatically retrieve files from** option. The Subscription must use a dedicated Subscription folder.
- A Basic application with a the subscription that contains the following settings: **Send files directly to** the same AS2 Transfer Site. The Subscription must use another Subscription folder (different from the one above).

Note Even though the AS2 Site is selected in the **Send files directly to** Subscription option, it is not practically used to send files.

Use case

This setup is good only for outbound transfers with asynchronous AS2-MDN. If your partner sends you an AS2 message (i.e. a file) you will return it back to them as the Route will be triggered from the AR subscription. This problem can be avoided if the route in the subscription sends the file to an internal server instead of to the partner AS2 Transfer Site (in the Advanced Routing Send To Partner step). If you wish to be able to send files outbound with this use case, you need another Advanced Routing instance with a route that uses an AS2 Transfer Site to send the files in a Send To Partner step, but without specifying **Automatically retrieve files from** Subscription option.

Scenario 2: Combine AR and Basic application for both inbound and outbound transfers

This approach is similar to the one from above with outbound transfers. This time you must also configure another Basic application for inbound transfers.

Basically, here you use Advanced Routing in one Subscription, and two (dummy) Subscriptions with Basic Application. Each of the three is subscribed to a separate folder once again. The AR is used to do the actual file transfers outbound, while one of the BA Subscriptions is used to receive the files, while the other one is taking care of receiving the asynchronous MDN receipts and "bridging the gap" between the MDN receipts and the Subscriptions.

Setup specifics

With this setup, you must configure:

- One AS2 Transfer Site (in the user account).
- An Advanced Routing instance with a Route that uses an AS2 Transfer Site to send the files in a Send To Partner step. The Subscription must use a dedicated Subscription folder.
- One "outbound" Basic application that uses the same AS2 Transfer Site. In the subscription to this application, you must set that AS2 Transfer Site in the **Send files directly to** option. The Subscription must use a second, different Subscription Folder.

- One "inbound" Basic application, that uses the same AS2 Transfer Site. In the subscription to this application, you must set that AS2 Transfer Site in the **Automatically retrieve files from** option. The Subscription must use a third, different Subscription folder.

Use case

This setup is good for both inbound and outbound transfers with asynchronous AS2-MDN.

Scenario 3: Combine AR for outbound and SiteMail for inbound transfers

This approach combines the use of Advanced Routing with Site Mailbox application to achieve for both inbound and outbound transfers.

Basically, here you use Advanced Routing in one Subscription. A dummy Site Mailbox application will be added (and subscribed to a different folder) to facilitate correct MDN processing in AR. In essence, the Advanced Routing will be used to send files and the Site Mailbox application will only be receiving the files and the asynchronous MDN receipts.

Setup specifics

- One AS2 Transfer Site (in the user account).
- An Advanced Routing instance with a Route that uses an AS2 Transfer Site to send the files in a Send To Partner step. The Subscription must use a dedicated Subscription folder.
- A Site Mailbox application, that uses the same AS2 Transfer Site. In the subscription to this application, you must select that AS2 Transfer Site in both the **Send files directly to** and **Automatically receive files from** Subscription options. The Subscription must use a second, different Subscription Folder.

Use case

This setup is good for both inbound and outbound transfers with asynchronous AS2-MDN.

AS2 transfers 16

Axway SecureTransport supports AS2 (Applicability Statement 2) as the industry standard for Internet-based data exchange.

Use the SecureTransport AS2-certified solution to exchange data with any trading partner using an AS2-interoperable solution over the Internet. AS2 simplifies communication by reducing the number of technologies an organization must support and manage. It would be cost prohibitive for small business partners to exchange data electronically with large organizations that use a different data transport standard. The AS2 standard allows organizations, both large and small, to implement one solution for data exchange with all business partners using an AS2 solution.

The AS2 standard secures data with S/MIME (Secure/Multipurpose Internet Mail Extensions) over HTTP (Hypertext Transfer Protocol) or HTTPS (secure HTTP over SSL), also using MDN (Message Disposition Notification). AS2 provides synchronous, real-time transmission of data with immediate message delivery notice.

Note In Windows, the names of files transferred using the AS2 protocol are limited to 255 characters.

For more information, see *AS2 implementation on page 1035*.

AS2 implementation

Companies that conduct Business-to-Business (**B2B**) Electronic Commerce can use SecureTransport Server to send business documents to their business partners using the AS2 protocol.

The SecureTransport AS2 server provides security by utilizing encryption and signing. The server sends a document that is signed and encrypted to the partner using digital certificates that are agreed upon between the server and the target partner.

The partner checks the signature of the document and sends a signed MDN receipt to acknowledge that the transfer succeeded. MDN is an Internet messaging format used to transmit a receipt. MDN is used interchangeably with receipt. MDN is a receipt. This MDN signature is used to prove that the original document was sent from the SecureTransport AS2 server to the partner in a process called non-repudiation.

Note The use of signing, encryption, and MDN receipts are optional AS2 features that are decided when businesses enter into a partnership.

The following topics describe the SecureTransport AS2 implementation:

- *Synchronous and asynchronous receipts* on page 1036 - Describes synchronous and asynchronous receipts.
- *AS2 and application framework: Architecture and workflow* on page 1036 - Describes the architecture and workflow of AS2 and the application framework.

- *SecureTransport AS2 server: Setup overview* on page 1037 - Provides an overview of the SecureTransport AS2 server setup.

Synchronous and asynchronous receipts

A synchronous receipt is returned to the sender during the same HTTP session as the sender's original message. An asynchronous receipt is returned to the sender on a different communication session than the sender's original message session.

The synchronous receipt is sent as an HTTP response to an HTTP POST or as an HTTPS response to an HTTPS POST. This form of AS2-MDN is called synchronous because the AS2-MDN is returned to the originator of the POST on the same TCP/IP connection.

The asynchronous AS2-MDN is sent on a separate HTTP or HTTPS TCP/IP connection. Logically, the asynchronous AS2- MDN is a response to an AS2 message. However, at the transfer-protocol layer, assuming that no HTTP pipe lining is used, the asynchronous AS2-MDN is delivered on a unique TCP/IP connection, distinct from that used to deliver the original AS2 message. When handling an asynchronous request, the HTTP response MUST be sent back before the MDN is processed and sent on the separate connection.

When an asynchronous AS2-MDN is requested by the sender of an AS2 message, the synchronous HTTP or HTTPS response returned to the sender prior to terminating the connection *must* be a transfer-layer response indicating the success or failure of the data transfer. The format of such a synchronous response *may* be the same as that response returned when no AS2-MDN is requested.

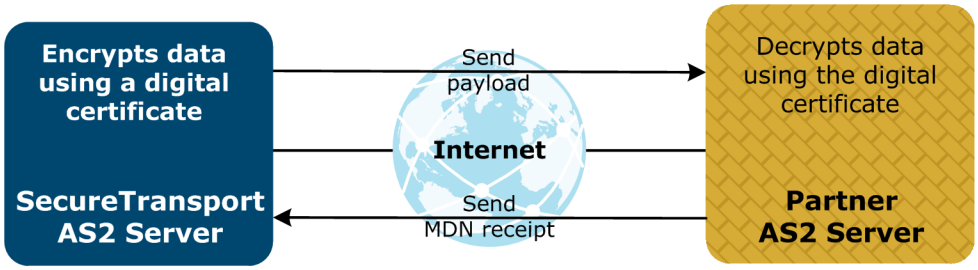


Figure 7. AS2 receipt

AS2 and application framework: Architecture and workflow

AS2 transfers can be handled by an AS2 application, which is an instance of the generic Site Mailbox application type. The AS2 Site Mailbox application can perform inbound and outbound transfers from and to the same AS2 site. To this end, the respective account is subscribed to the AS2 Site Mailbox application and the subscription is associated with the AS2 site that is set up during the definition of the transfer site of the account. The application outgoing and incoming subfolders are called by default `outbox` and `inbox`.

Some basic application framework concepts, relating to the AS2 implementation in SecureTransport, are outlined here.

The following topics describe the architecture and workflow of AS2 and the application framework:

- *Application for AS2* on page 1037
- *AS2 site* on page 1037

Application for AS2

The particular AS2 application that handles AS2 transfers can be an instance of the Site Mailbox application type or the Basic or Advanced Routing application types. To be able to transfer files over the AS2 protocol, the user account must be subscribed to the AS2 application and the subscriptions must have the AS2 site defined.

The AS2 application must have an outgoing subfolder `outbox` and an incoming subfolder `inbox`.

AS2 site

AS2 partnership holds a description of local and remote parties which participate in AS2 transfers. Besides identification information (local and remote IDs and remote address), the partnership holds information about data transformation of payload and MDN requirements.

The SecureTransport architecture, implemented by the application framework, introduces the site concept. A SecureTransport site has all the necessary attributes to connect to the AS2 site and transfer the contents. The only partnerships items that are not a part of the SecureTransport site specification are the following:

- **Subfolder** – an attribute of subscription between the AS2 user account and the AS2 Site Mailbox application.
- **Username** – an attribute of the user account for AS2.

Note AS2 site does not support server-initiated inbound transfers (server pull).

For details, see *Manage applications* on page 1.

SecureTransport AS2 server: Setup overview

Use the following procedure to set up a SecureTransport Server for AS2 transfers:

1. Configure the AS2 server control settings. See *Manage an AS2 server on page 272*.
2. Configure the AS2 settings. See *Configure the AS2 server settings on page 80*.
3. Configure an HTTP proxy (optional). See *Specify proxy settings in a network zone on page 234*.
4. Start the Transaction Manager and AS2 servers. See *Server control on page 259*.
5. Create a SiteMailbox application, or a Basic or Advanced Routing application. See *Manage applications*.
6. Create a user account. See *Create a user account on page 503*.

7. In the user account, create an AS2 transfer site using the AS2 protocol. During this step, specify the AS2 local and remote sites in the transfer site definition. See *AS2 transfers on page 1035*.
8. In the user account, create a subscription to the SiteMailbox application or to the Basic or Advanced Routing application, and specify the AS2 transfer site. See *Subscribe an account to an application on page 664*.

File services interface transfers 17

A developer can use the Axway SecureTransport file services interface feature to implement a protocol that can use a shared file system and a transferred metadata file.

The following topics describe file services interface transfers:

- *File services interface overview* on page 1039 - Provides an overview of the file services interface.
- *Receive files using a file services interface protocol* on page 1040 - Provides how-to instructions for receiving files using a file services interface protocol.
- *Send files using a file services interface protocol* on page 1044 - Provides how-to instructions for sending files using a file services interface protocol.

File services interface overview

For SecureTransport to receive a file using the file services interface, the external system must copy the file into the shared file system. The external system then sends the parameters for the transfer in the metadata file using another protocol.

To perform a file services interface transfer, the external server:

1. Copies the file to transfer to a location in the shared folder.
2. Construct a metadata file that specifies the parameters of the transfer.
3. Upload the metadata file into the subscription folder of an application of type File Transfer via File Services Interface.

The SecureTransport Server:

1. Reads the specification of the transfer from the metadata file.
2. Copies the transferred file from the shared directory to the locations specified in the metadata file.
3. Deletes the metadata file.
4. Deletes the transferred file from the shared directory, if specified in the metadata file.

When SecureTransport sends a file using the file services interface, it calls a connector process configured by the developer which pushes the file to the remote server.

Receive files using a file services interface protocol

For SecureTransport to receive files sent by another system using a file services interface protocol, you must set up access to a shared directory, create an application, and subscribe an account to that application.

1. Determine which shared directory the systems are to use to transfer the files and make it accessible from both systems.
2. Perform the procedure in *File Transfer via File Services Interface application* on page 837.
3. Create or identify an account to receive the metadata files from the other system. For details, see *Create a user account* on page 503.
4. Subscribe that account to the file services interface application. For details, see *Subscribe an account to an application* on page 664.

When the remote system transfers the metadata file to the subscription folder, the file services interface application reads it and processes the transferred file based on its contents.

The following topics describe the metadata file and the location of the transferred file:

- *Metadata file* on page 1040 - Lists and describes the elements in the XML metadata file.
- *Location of the transferred file* on page 1043 - Describes the location of the transferred file.

Metadata file

The metadata file is an XML file that contains a `Transfer` element. The elements in the following table must be included in the `Transfer` element unless noted as optional:

Element	Description	Valid values	Notes
`SourceFileLocation`	Path to the transferred file	Full path name or path relative to `RemoteSharePath`	See *Location of the transferred file* on page 1043.
`RemoteSharePath`	Path to the shared directory on the remote system	Full path name	Optional. SecureTransport uses this to determine the location of the transferred file. See *Location of the transferred file* on page 1043.

Element	Description	Valid values	Notes
`LocalSharePath`	Path to the shared directory on the SecureTransport system	Full path name	Optional. Optional for transfers to SecureTransport. SecureTransport uses this to determine the location of the transferred file. See *Location of the transferred file* on page 1043.
`CycleID`	Processing cycle identifier for the file transfer	Any valid cycle ID	SecureTransport uses this cycle ID in events reported to Axway Sentinel.
`Protocol`	Name of a file services interface protocol	Any protocol defined in the file services interface protocol registry	SecureTransport displays the corresponding display name in Protocol column of the *File Tracking* page and in events it sends to Axway Sentinel.
`Mode`	File transfer mode	`A` for ASCII or `I` for binary	
`Recipients`	Container for `Recipient` elements	N/A	Optional. Contains any number of `Recipient` elements.
`Recipient`	A recipient for the file	N/A	Optional. Contains a `Name` element and, optionally, a `Path` element

Element	Description	Valid values	Notes
`Name`	Login name of a SecureTransport account to receive the file	Any existing account login name	
`Path`	Path to the directory where SecureTransport copies the file, relative to the account home folder	Path to any directory in the home folder of the named account	Optional. The default is the home folder of the account.
`Parameters`	Container for `Parameter` elements	N/A	Optional. Contains any number of `Parameter` elements.
`Parameter`	A parameter	N/A	Optional. Contains a `Key` element and, optionally, a `Value` element.
`Key`	Key ID for the parameter	Any string	
`Value`	Value of the parameter	Any string	Optional.

The following is an example of a metadata file:

```
<?xml version="1.0" encoding="UTF-8"?>
<Transfer>
   <SourceFileLocation>/opt/shared/incoming/report-20110704
      </SourceFileLocation>
   <CycleId>2162164</CycleId>
   <Protocol>T3Direct</Protocol>
   <Mode>I</Mode>
   <Recipients>
      <Recipient>
         <Name>acctng</Name>
         <Path>/incoming/reports</Path>
      </Recipient>
```

```
         <Recipient>
         <Name>audit</Name>
      <Path>/incoming/check</Path>
      </Recipient>
   </Recipients>
   <Parameters>
      <Parameter>
         <Key>status</Key>
         <Value>complete</Value>
      </Parameter>
   </Parameters>
</Transfer>
```

Location of the transferred file

For files pushed to SecureTransport from a remote system, use `SourceFileLocation` and optionally `RemoteSharePath` and `LocalSharePath` to specify the location of the transferred file:

- If `RemoteSharePath` or `LocalSharePath` is not specified, `SourceFileLocation` is the location of the transferred file on the SecureTransport system.
- If `RemoteSharePath` and `LocalSharePath` are specified and `SourceFileLocation` starts with `RemoteSharePath`, then SecureTransport replaces `RemoteSharePath` with `LocalSharePath` to determine the location of the transferred file on the SecureTransport system.
- If `RemoteSharePath` and `LocalSharePath` are specified and `SourceFileLocation` does not start with `RemoteSharePath`, then SecureTransport concatenates `LocalSharePath` and `SourceFileLocation` to determine the location of the transferred file on the SecureTransport system.

For files pushed to a remote system from SecureTransport, `LocalSharePath` must be specified. Use `SourceFileLocation` and optionally `RemoteSharePath` to specify the location of the transferred file:

If `RemoteSharePath` is specified, SecureTransport concatenates `LocalSharePath`, a unique directory name, `RemoteSharePath`, a unique file prefix, and `SourceFileLocation` to determine the location of the transferred file on the SecureTransport system.

If `RemoteSharePath` is not specified, SecureTransport concatenates `LocalSharePath`, a unique directory name, a unique file prefix, and `SourceFileLocation` to determine the location of the transferred file on the SecureTransport system.

Send files using a file services interface protocol

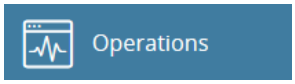
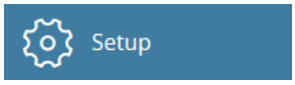
For SecureTransport to send files to another system using a file services interface protocol, you must create a transfer site for a file services interface protocol, and subscribe to an application that sends files to that site.

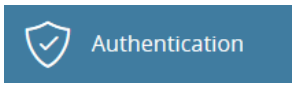
1. Create or identify an account to send files to the other system. For details, see *Create a user account* on page 503.
2. Create a transfer site in that account that references the file services interface protocol. For details, see *File services interface transfer sites* on page 554.
3. Subscribe that account to an application configured to send files to that transfer site. For details, see *Subscribe an account to an application* on page 664.

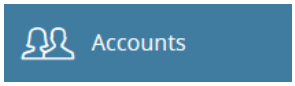
When the application sends a file to the transfer site, SecureTransport finds the information for the protocol in the protocol registry and calls a program configured for the protocol to perform the transfer.

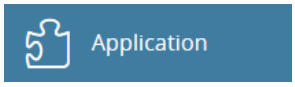
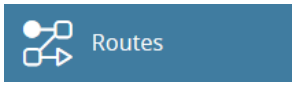
Appendix A: Administration Tool features checklist

Most Administration Tool features are present in Axway SecureTransport Server, but not all of those are available in the Administration Tool for SecureTransport Edge. The following table lists all available features within the Administration Tool for both SecureTransport Edge and SecureTransport Server.

Administration Tool features	SecureTransport Edge	SecureTransport Server
Operations Menu 	P	P
Server Control	P	P
Cluster Management	P	P
Server Usage Monitor	P	P
File Tracking	—	P
Server Log	P	P
Audit Log	P	P
Server Configuration	P	P
Support Tool	P	P
Setup Menu 	P	P
Certificates	P	P
FTP Settings	P	P
AS2 Settings	P	P
SSH Settings	P	P

Administration Tool features	SecureTransport Edge	SecureTransport Server
Admin Settings	P	P
PeSIT Settings	P	P
AdHoc Settings	—	P
Database Settings	P	P
Axway Sentinel/DI	—	P
Server License	P	P
Allowed ST Servers	P	—
Command Logging	P	P
Transfer Logging	P	P
Holiday Schedule	—	P
Mail Templates	—	P
Miscellaneous	P	P
ICAP Settings	—	P
TM Settings	—	P
File Archiving	—	P
Network Zones	P	P
Address Books	—	P (If enabled on the *Server Configuration* page.)
Authentication Menu 	P	P
Login Settings	P	P
LDAP Domains	—	P

Administration Tool features	SecureTransport Edge	SecureTransport Server
SiteMinder Settings	—	P
User Type Ranges	—	P (UNIX-based servers only)
Home Folders	—	P
Accounts Menu 	P	P
User Accounts	—	P
Unlicensed Accounts	—	P
Service Accounts	—	P
Import/Export	—	P
Administrators	P	P
Change Password	P	P
Manage Roles	—	P
Account Templates	—	P
System	—	P
Business Units	—	P
Active Users	—	P
Access Menu 	P	P
User Classes	—	P
Secure Socket Layer	P	P
Virtual Groups	—	P

Administration Tool features	SecureTransport Edge	SecureTransport Server
Restrictions	—	P
FTP Commands	P	P
Admin Access Control	P	P
Server Access Control	—	P
Access Rules	P	P
Login Restrictions	—	P
Application 	—	P
Application	—	P
Routes 	—	P
Route Packages	—	P

Appendix B: Troubleshoot common problems

The following topics discuss the most common issues that can occur when using SecureTransport. If you are still having issues after following the procedures in this topic, contact Axway Global Support for further assistance. For more information, see *Get more help* on page 27.

Communication problems

When the SecureTransport Server and the SecureTransport Edge are unable to communicate with each other, you can follow the procedures below. The main symptom of a communication problem is the inability to log into the server from the client. Each topic explains how to troubleshoot the system to determine if you are experiencing one of these common problems. Make sure that you check each item in the order listed.

1. **Clock Settings** – Verify that the clock settings for the servers are correct.
2. **Trust Establishment** – Check the log files for errors that can occur between SecureTransport Server and SecureTransport Edge
3. **Connectivity** – Make sure that the correct IP addresses are listed.

The following topics provide lists of what to check for communication problems:

- *Clocks out of sync* on page 1050 - Provides a list of what to check for clocks out of sync issues.
- *Trust establishment issues* on page 1050 - Provides a list of what to check for trust establishment issues.
- *Connectivity* on page 1051 - Provides a list of what to check for connectivity issues.

Clocks out of sync

If the clocks on both the SecureTransport Server and SecureTransport Edge are out of sync, the two servers might not be able to communicate correctly. Verify that both systems have the same date and time set.

If you are using the Windows platform, the SecureTransport Edge might not be in the same domain, but instead in a workgroup. If this is the case, the SecureTransport Edge does not use the Windows Time Service and you must manually verify that the clocks are in sync.

When using a UNIX-based platform, make sure that the time formats and time zones are the same. If they are not the same, the imported CA certificate might have a different "valid from" date and time, and are considered invalid until the server reaches the listed date and time.

For an Enterprise Cluster (EC), the clocks of all servers must be synchronized.

Trust establishment issues

If the certificates are not configured correctly for both the SecureTransport Server and the SecureTransport Edge, trust might not be properly established between the two systems. Try the following procedures to verify your trust settings.

- To establish trust between SecureTransport Server and SecureTransport Edge, you need to exchange Trusted CA certificates. Exchanging certificates consists of:
    - Saving the CA certificate for SecureTransport Server to a file on a local system and importing the file to the SecureTransport Edge using the SecureTransport Administration Tool.
    - Saving the CA certificate for SecureTransport Edge to a file on a local system and importing the file to the SecureTransport Server using the SecureTransport Administration Tool.

To export or import a CA certificate, see *Manage trusted CAs* on page 55.

Common certificate errors

If certificates are not correctly imported or the certificate has expired, you might see ERROR-level entries regarding SSL handshaking in the server log on the SecureTransport Server such as:

```
com.valicert.brules.eventmonitor - SSL handshake failed
```

Specific issues that can cause an error include:

- SecureTransport Server CA certificate is not imported into Edge.
- SecureTransport Edge CA certificate is not imported into Server.
- SecureTransport Server CA certificate is expired.
- SecureTransport Edge CA certificate is expired.

- SecureTransport Server certificate is expired.
- SecureTransport Edge certificate is expired.

Use **Setup > Certificates** to view the certificates and monitor expiration dates on a regular basis.

Connectivity

Communication issues can arise when IP addresses are not recognized properly. Use the following procedures to check your connectivity settings.

- Make sure that the correct IP addresses are listed for the SecureTransport Server and SecureTransport Edge in the `hosts` file. For details, see *Incorrect host name and IP address in the host file* on page 1052.
- Make sure that the network zones on the SecureTransport Server and SecureTransport Edge are correct and consistent.
- Make sure that the SecureTransport Server is listed as an allowed server on the SecureTransport Edge.
- Make sure that the `Streaming.TrustedAliases` server configuration parameter is set correctly. For more information, see *Secure the communication between the TM server and the protocol servers* on page 237.
- Make sure that the firewall is configured correctly. For more information on configuring the firewall, see *Firewall settings* on page 1089.

Servers do not start

If a SecureTransport protocol or TM server does not start, check the following topics to troubleshoot the problem. Make sure you check each item in the order listed.

1. **SSL Certificate** – Verify that the SSL certificate is properly configured for the server.
2. **Conflicting Port Numbers** – Verify that the port number assigned to the server is not in use elsewhere.
3. **Host Name** – Look in the server host file for an entry for each host name with the correct IP address.

The following topics provide lists of what to check for when servers do not start:

- *No SSL certificate configured for the server* on page 1052 - Provides how-to instructions for verifying that an SSL certificate is configured for the server.
- *Conflicting port numbers* on page 1052 - Provides how-to instructions for verifying port assignments and eliminating conflicting port assignments.
- *Incorrect host name and IP address in the host file* on page 1052 - Provides how-to instructions for verifying IP address and host name assignments.

No SSL certificate configured for the server

Verify that the SSL certificate configured for the server has not expired. If the certificate is valid, use the following procedure to make sure that the SSL certificate is properly configured for the server that is not starting.

1. Select **Operations > Server Control**.
2. For each server, select an appropriate certificate alias from the **SSL Key Alias** list.
3. At the bottom of the page, click **Update**.
4. Restart each server for which you changed the certificate.

In a synchronized configuration, the certificate aliases must match on the primary and the secondary nodes, or after synchronizing, the link between the server and the certificate is broken.

Conflicting port numbers

Make sure the port number assigned to the server is not in use elsewhere, such as the operating system SSH server using port 22. To check the port, open the SecureTransport Administration Tool and select **Operations > Server Control**. Verify that the ports assigned to the FTP Server, the HTTP server, the AS2 Server, the SSH server, and the PeSIT Server are not in use by other servers.

Incorrect host name and IP address in the host file

Verify that the correct IP addresses are in the `hosts` file for each system running SecureTransport. Look in the hosts file for an entry for each host name with the correct IP address. This file is located in the `/etc` directory on UNIX-based systems. The `hosts` file can be found in the `WINNT\System32\drivers` directory on Windows Server.

Entries in the hosts file have the following format:

```
127.0.0.1 localhost.localdomain localhost
```

If the entry is not present, create it. If there is an entry, make sure that it has the correct IP address listed. Entries should have the primary alias for each IP address first, followed by the aliases for all other interfaces. You should also verify that your DNS is set up correctly. For more information, see *DNS settings* on page 1060.

Cannot log in as a client

If you attempt to login to SecureTransport and the login fails despite using a correct user name and password, try the following topics to troubleshoot the problem. Make sure that you check each item in the order listed.

1. **License Issues** – Verify that the required licenses are installed and current.
2. **Connection to Server** – Verify that the client system can communicate with the server and that no other client is experiencing a problem.
3. **Authentication** – Check the LDAP or SiteMinder settings.
4. **LDAP** – Verify that the LDAP configuration is set correctly.
5. **Unable to use file system commands such as ls** – make sure that you have plenty of hard drive space available.
6. **Unable to log in to SecureTransport Edge, but can log in to SecureTransport Server** – Make sure that the certificates for Edge and Server match.
7. **Client certificate authentication fails** – Make sure that you do not have two Root CA certificates in the "Trusted Certificates" keystore with an identical DN specified.
8. **Logged in to client with reduced functionality** – Make sure that you are using the required browser and that the required features are installed and enabled.
9. **Session terminates due to CSRF protection** – Add the client to the white list.

The following topics provide procedures for troubleshooting client log in problems:

- *License issues* on page 1053 - Provides how-to procedures for troubleshooting license issues.
- *Connectivity to server failed* on page 1054 - Provides how-to procedures for troubleshooting connectivity issues.
- *SiteMinder issues* on page 1054 - Provides how-to procedures for troubleshooting SiteMinder issues.
- *LDAP issues* on page 1055 - Provides how-to procedures for troubleshooting LDAP issues.
- *File system commands not functional* on page 1055 - Provides how-to procedures for troubleshooting file system command functionality issues.
- *Cannot log in to SecureTransport Edge* on page 1055 - Provides how-to procedures for troubleshooting SecureTransport Edge log in issues.
- *Client certificate authentication fails* on page 1056 - Provides how-to procedures for troubleshooting client certificate authentication issues.
- *Session terminates due to CSRF protection* on page 1056 - Provides how-to procedures for troubleshooting session termination issues due to CSRF protection.

License issues

If you get an error indicating that the license is expired or that there is no license available for ad hoc users, verify that the license is installed and within the validity period. Use the following procedure to verify that your license is still valid.

To view server licenses:

1. Open the SecureTransport Administration Tool and select **Setup > Server License**.
2. Verify that the license is installed and not expired by checking the **Core Server License** for the FTP and HTTP servers, user accounts, and ad hoc users and the **Features License** for AS2, SSH, and Connect:Direct protocols and the SiteMinder feature.
3. If your license has passed the expiration date, contact Axway Global Support to renew it. For details, see *Get more help* on page 27.

For each license type, the validity period is given in the **Valid from** and **Valid to** fields. Check the following items if your license is valid.

- Make sure that you are using the correct IP address to connect to the server from the client.
- Make sure that your server's clock is set correctly.
- If you experience a license error, log in to the SecureTransport Administration Tool, select **Operations > Server Log**, and look in the following log entries for the exact error code and description.
    - AS2: AS2D
    - FTP or FTPS logins (secure or nonsecure): FTPD
    - HTTP or HTTPS logins: HTTPD
    - PeSIT transfer: TM
    - SiteMinder: TM
    - SSH: SSHD
    - Transaction Manager: TM

Connectivity to server failed

SecureTransport Server and SecureTransport Edge must be able to communicate with each other properly. Make sure that the following settings are configured correctly.

- **DNS Settings** – For details, see *DNS settings* on page 1060.
- **Firewall** – For details, see *Firewall settings* on page 1089.

SiteMinder issues

Check the SiteMinder settings to make sure that authentication is configured correctly.

To view the SiteMinder settings:

1. Open the SecureTransport Administration Tool on the SecureTransport Server and select **Setup**.
2. Select **SiteMinder Settings**. Verify that the following settings are correct: IP Address, Authorization Port, Authentication Port, Accounting Port, Agent Name, Shared Secret, Maximum Connections, Connection Timeout, File Storage Root Path, Default Home Directory, Default Local User ID, and Default Local Group ID.

LDAP issues

Common LDAP problems are incorrect LDAP configuration, incorrect service account credentials, and an inability to access LDAP through the firewall. Check the following setting in the SecureTransport Administration Tool to troubleshoot problems:

- Make sure that the credentials LDAP can search through are correct for each user unable to login from the client. If the users are connecting to an Active Directory (AD) LDAP server, the Bind DN and Password need to be specified. AD does not allow anonymous binds. Verify that the user has the correct user name and password and is entering the correct information into the client.
- For Windows-based systems, make sure that the LDAP SysUser is set correctly to a local or domain user that has the necessary permissions on the directory. The LDAP SysUser must be present in the password vault.
- Make sure that the correct LDAP port, generally 389 or 3268, is available to SecureTransport in the firewall.
- Check that for each LDAP-based user, the settings allow the user to obtain the UID/GID (UNIX-based systems only), user type, and home directory either from the LDAP directory (using attribute mapping) or through the user defaults. Make sure that the user home directory exists and has the correct permissions. You can set up a login agent for this purpose.
- Make sure that you use the correct LDAP Protocol version, and the LDAP server is the same version as the one selected in the SecureTransport Administration Tool on the *LDAP Server* page.
- Try using the `ldapsearch` command from the command line. If the command fails, you might have an incorrect search query or insufficient credentials.
- If you are not using attribute mapping, make sure that the home directory exists in the LDAP User Default and that the entry is enabled. For more information, see *LDAP logins* on page 476.

File system commands not functional

If you are unable to log in or file system commands such as `ls` or `dir` do not work correctly, the server might be low or out of disk drive space on the installation volume. Free up additional disk space and try logging in again.

Cannot log in to SecureTransport Edge

If you are unable to log in to a SecureTransport Edge, but you can log into the SecureTransport Server directly, make sure that the Internal CA certificate matches the local certificates created for the protocol servers.

Also make sure that your SecureTransport Server and SecureTransport Edge have exchanged certificates as appropriate.

Client certificate authentication fails

If you have two Root CA certificates in the "Trusted Certificates" keystore with an identical DN specified, you might not be able to log into SecureTransport. SecureTransport searches for the first certificate that matches the DN of the Certificate Authority. If a match is found, then SecureTransport does not continue to search. When two or more CA certificates with the same DN exist, the correct CA might never be selected.

Make sure that all CA certificates have unique DN values.

Session terminates due to CSRF protection

SecureTransport implements token-based cross-site request forgery (CSRF) protection. This prevents a user who is logged in to a SecureTransport web client from causing unwanted modification of user data, such as uploading or deleting a file, by opening a malicious web page or clicking a crafted link in another browser tab or window. When the CSRF protection detects a violation, SecureTransport marks the session as expired and logs out the user.

If an HTTP client identifies itself with a User-Agent string, SecureTransport matches the string against a white list of clients represented by a Perl-compatible regular expression. If the User-Agent string matches, SecureTransport does not perform CSRF protection. By default, the white list is `(^SecureTransport|^Axway/SecureClient|^Axway/EndPoint|Java|^curl/|^Jakarta Commons\-HttpClient)`.

To configure a different white list, save the regular expression in the `Http.UserAgentWhiteList` server configuration parameter. To exclude another client from CSRF protection, add a pattern that matches the User-Agent string for that client to the regular expression. For example, to exclude Wget in addition to the clients in default, set the value of the `Http.UserAgentWhiteList` server configuration parameter to `(^SecureTransport|^Axway/SecureClient|^Axway/EndPoint|Java|^curl/|^Jakarta Commons\-HttpClient|^Wget)`. If the value of the `Http.UserAgentWhiteList` server configuration parameter is empty, SecureTransport uses the default white list.

FTP does not work through the firewall

If you are having a problem using FTP through a firewall, use the following topics to help you troubleshoot the problem. Make sure you check each item in the order listed.

1. **Firewall Rules** – Verify that the firewall rules open the ports specified for using FTP.
2. **Passive Port Range** – Make sure the passive port range is configured correctly.
3. **Check Point Firewalls** – Make sure you set up the Check Point firewall to use bidirectional transfers.

The following topics provide procedures for troubleshooting FTP protocol firewall issues:

- *Firewall rules prevent the port from opening* on page 1057 - Provides procedures for troubleshooting port opening firewall rule issues.
- *Passive port range is not defined in the firewall* on page 1057 - Provides procedures for troubleshooting port definition firewall issues.
- *Check Point firewall is not configured for bidirectional transfers* on page 1057 - Provides procedures for troubleshooting Check Point firewall bidirectional transfer issues.

Firewall rules prevent the port from opening

Make sure the rules for your firewall open the ports specified for using FTP. For more information, see *Firewall settings* on page 1089.

Passive port range is not defined in the firewall

Configure the passive mode port range in your firewall if you are planning on using passive FTP. The passive range is defined on the SecureTransport Server and must be made available on the firewall. To set the passive range on the SecureTransport Server, open the SecureTransport Administration Tool and select **Setup > FTP Settings**. Configure the port range in the *FTP Passive Mode* pane of the page.

Firewalls with no stateful inspection must make this port range explicitly available. Firewalls that support stateful inspection can define the port range dynamically. To ensure proper operation, the FTP control channel must not be encrypted. If you are using a third-party secure FTP client, use Clear Command Channel (CCC).

If you are using Axway Secure Client, you can use firewall-friendly Tunnel mode.

Note Because the Axway Secure Client firewall-friendly Tunnel mode uses SSL v3, you cannot use it for FTPS in FIPS transfer mode.

Check Point firewall is not configured for bidirectional transfers

The symptoms of incorrectly configured bidirectional FTP transfers include:

- The error message `message_info violated unidirectional connection` in the Check Point log viewer
- Bidirectional FTP data connections getting dropped

These symptoms can occur because bidirectional FTP data connections are not allowed by default. Bidirectional FTP data connections are not considered as safe since the data connection is interactive and the connection changes the basic way FTP works.

Check Point firewalls need to be configured to use bidirectional transfers. In the Check Point NG firewall (AI R55 and higher), set the FTP connection to FTP_BASIC. This allows bidirectional communications and sets the firewall to allow commands not terminated with a new line.

PeSIT file transfers fail over TLSv1 Legacy for certain ciphers

A change to CBC ciphers in SecureTransport made in response to CVE-2011-3389 causes older version of Transfer CFT and other PeSIT clients to fail to transfer files from and to a SecureTransport server over TLSv1 Legacy. These clients fail because they do have the update or have other deficiencies in their SSL implementations.

Note The following workarounds enable the file transfer between these clients and SecureTransport but both are considered insecure and using any is at your own risk.

Failed PeSIT file transfers to SecureTransport

To enable transfer of files to SecureTransport, disable the fix in SecureTransport by adding the following Java option in `<FILEDRIVEHOME>/bin/start_pesitd`:

```
JAVA_OPTS="-Djsse.enableCBCProtection=false $JAVA_OPTS"
```

Failed PeSIT file transfers from SecureTransport

To enable SecureTransport to transfer files to such clients, disable the fix by adding the following Java option in `<FILEDRIVEHOME>/bin/start_tm_console`:

```
JAVA_OPTS="-Djsse.enableCBCProtection=false $JAVA_OPTS"
```

SIT transfers fail when using DSA certificates

Server-initiated transfers might fail if specific DSA certificates are used by the remote server. To prevent this issue, add the following Java option in `<FILEDRIVEHOME>/bin/start_tm_console`:

```
-Dorg.bouncycastle.jsse.client.dh.unrestrictedGroups=true
```

Save your change and restart the TM Server.

Debug SSH issues

If you are trying to determine an issue with SSH, changing the `mchange` and `ehcache` logger parameters in the `sshd-log4j.xml` file to `Debug` can prevent SSH from receiving connections.

To work around this issue, you can reset `sshd-log4j.xml` to send log messages to a file instead of the database. Follow the steps in *Redirect log4j output from the database* on page 1082 to change the log.

Note When log messages are stored in the database, they are displayed in the *Server Log* page. When you store the log messages in a file, they are not displayed in the *Server Log* page.

Performance issues

When SecureTransport performance is reduced from previous levels or is not consistent your expectations, use the checklist below to troubleshoot the issue and determine if performance can be improved. Make sure you check each item in the order listed.

1. Evaluate performance issues – Investigate factors that might result in reduced performance.
2. DNS Settings – Verify that the DNS settings for the servers are correct by using `nslookup` in Windows or UNIX-based systems.
3. Firewall Settings – Verify that the firewall is configured properly for SecureTransport and that no other application is also experiencing a firewall problem.
4. System Resources – Look at the memory and CPU usage for any other services running on the same computer.
5. Installation Drive – To avoid performance problems, always install SecureTransport on a local disk drive.
6. Log Level – Check the log level.

The following topics provide how-to procedures for evaluating and troubleshooting performance issues:

- *Evaluate performance issues* on page 1059 - Provides how-to procedures for evaluating performance issues.
- *DNS settings* on page 1060 - Provides how-to procedures for troubleshooting DNS settings issues.
- *Firewall issues* on page 1061 - Provides how-to procedures for troubleshooting firewall issues.
- *Other services using to much CPU or memory* on page 1061 - Provides how-to procedures for troubleshooting CPU and memory usage issues.
- *Installation on network drive* on page 1062 - Provides how-to procedures for troubleshooting drive issues.
- *Debug log output slows computer* on page 1062 - Provides how-to procedures for troubleshooting debug log output issues.

Evaluate performance issues

Before considering configuration changes to improve performance, investigate factors that might result in reduced performance. Consider the following factors:

- How many transfers have occurred during recent typical and peak transfer periods? Has the number of transfers increased from the levels before the performance problems were observed?
- How many concurrent users or partner sessions are transferring files during recent typical and peak transfer periods? Has the number of user or sessions increased from the levels before the performance problems were observed?
- Is your organization no longer meeting service level agreements with your customers?

If performance issues are due to increased workload, perhaps increasing slowly over time, evaluate implementing a cluster or adding a server to an existing cluster to accommodate the additional workload.

DNS settings

Try the following procedures to verify that your DNS settings are not causing performance to deteriorate.

- Verify that the DNS settings for the servers are correct by using `nslookup` in Windows or UNIX-based systems. If the `nslookup` returns an error you need to reconfigure your DNS settings. If `nslookup` times out, make sure that the firewall allows port 53 UDP/TCP.
- Reverse DNS lookups are used to resolve an IP address into a fully qualified domain name. The domain name is used for logging purposes and for applying access rules that specify a host name instead of an IP address. Reverse DNS lookups might cause server performance to deteriorate since a series of requests through the DNS name server tree is made each time.

  To disable reverse DNS lookups for the FTP, HTTP, and SSH servers:

  1. Open the SecureTransport Administration Tool in a web browser and log in as the administrator.
  2. Select **Setup > Miscellaneous**.
  3. For **Reverse DNS Lookups**, choose `Reverse DNS lookups disabled`.
  4. Click **Apply**.

  To disable DNS lookups for Administration Tool server:

  1. Open the SecureTransport Administration Tool in a web browser and log in as the administrator.
  2. Select **Operations > Server Configuration**.
  3. Search for the `Admin.ReverseDNSLookup` system configuration parameter.
  4. Click the Edit icon () in the **Edit** column, type `Off` in the **Value** column, and click the Save icon () in the **Edit** column.
  5. Restart the server.

- If you need to use reverse DNS lookups, make sure that the DNS path is not blocked by a firewall. Firewalls can block the DNS path when SecureTransport is in the peripheral network (DMZ). Try one of the following workarounds to allow access through the firewall:

    - Allow SecureTransport access to port 53 TCP/UDP.
    - Include a hosts file entry for every computer using the Administration Tool.
    - Connect to the Administration Tool through a proxy, and put a host file entry for just the proxy.
- In some cases, SecureTransport performs NIS lookups for users and groups, even when real users are disabled in SecureTransport. If there is a large number of users defined in NIS, server performance can deteriorate. To work around the issue, take the computer out of NIS.

Firewall issues

To eliminate any possible firewall settings that can affect performance, check the following:

- Make sure the ports SecureTransport uses are set correctly. If your firewall uses passive mode, make sure that you are using Clear Command Channel (CCC) or Firewall Friendly mode.
- Verify that no other applications are experiencing problems with the firewall.
- The Transaction Manager has a timeout value that allows connections to close before the firewall can close them. Make sure the timeout setting in `<FILEDRIVEHOME>/brules/conf/brules.xml` on the SecureTransport Server is a lower value than the timeout setting for the firewall. Look for the following setting in the Event Monitor element in the file:

  ```
  <!-- single simple value, timeout in seconds -->
  <client timeout="900"/>
  ```

  Change the setting to be smaller than the value of the firewall timeout.

Other services using to much CPU or memory

One possible reason performance deteriorates can be caused by other services using too many system resources. Perform the following procedures to fine tune performance.

- Look at the memory and CPU usage for any other services running on the same computer. If you turn off all the SecureTransport services, but you still see high memory usage, you might need to add more memory or assign a dedicated memory amount to SecureTransport. If the CPU usage is still too high, you need to move some of the services to another computer or turn them off. If you can configure the services to use less CPU resources, do so.

Because SecureTransport is a CPU-intensive application, during peak demand times, it can consume most or all of the CPU resources. This can reduce the performance of other services. To provide sufficient processor resources for SecureTransport and other services, allocate a computer with higher processing power to SecureTransport. For example, employ a computer with multiple processors.

Installation on network drive

The SecureTransport Server and Edge must access disk files and the database on disk for all file transfer and processing actions. If SecureTransport is installed on a network or shared drive, application performance depends on network throughput and processes from other computers accessing the drive.

SecureTransport installation on a network drive, regardless of its performance characteristics, is not supported. To avoid performance problems, always install SecureTransport on a local disk drive.

Debug log output slows computer

You might need to set the log level for one or more of the SecureTransport logs configured in files in the `<FILEDRIVEHOME>/conf/` directory to `debug` to produce a detailed log for problem isolation. If you leave a log set to `debug`, it can produce a very large volume of log output. Because this keeps the database and disk busy, it can affect the performance of all processes running on the computer.

- Always reset the log level after using `debug` to gather log information for problem isolation.
- Never set a log level to `debug` for routine operation.

Troubleshooting I/O problems

This section provides solutions to common I/O issues:

Problem: I/O error when a large number of write or read operations are happening simultaneously

Problem summary: SecureTransport may fail to retrieve file attributes or decrypt repository encrypted files when sending them outbound, for example. The issue is observed mainly in cluster environments with GlusterFS.

Possible cause: Read/write operations may require more time.

Solution: In this case, retrying failed read/write operations more times or/and for longer intervals may solve the issue. SecureTransport uses an exponential backoff algorithm for retries: it increases the waiting time between the consecutive retries after each retry failure following the logic:

The first retry is after 1 x `retryTime`, the second retry is after 2 x `retryTime`, the third after 3 x `retryTime` and so on until the configured number of `retries` is reached.
where

- `retries` is the maximum number of retry attempts for failed read/write operations configured in the `com.axway.st.server.fs.attributes.read.retries` parameter (10 by default).
- `retryTime` is the delay of the first retry and the growth rate between attempts configured in the `com.axway.st.server.fs.attributes.read.retryTime` parameter. (100 ms by default).

With the default values, SecureTransport retries 10 times: the first retry happens in 100 ms, the second after 200, the third after 300, and so on. This sets a deadline of 5.5 seconds for SecureTransport to retry a failed read/write operation.

To customize the read/write retry logic, add the following in the `<FILEDRIVEHOME>/conf/STStartScriptsConfig` file and edit the values to fit your needs:

```
TM_JAVA_OPTS="-Dcom.axway.st.server.fs.attributes.read.retries=20 $TM_JAVA_OPTS"
TM_JAVA_OPTS="-Dcom.axway.st.server.fs.attributes.read.retryTime=200 $TM_JAVA_OPTS"
```

In this example, we limit the number of retries to 20 and set the delay to 200 ms. The retries happen in 200, 400, 600, 800,...., 4000 ms., and the total amount of time for retrying is 42 seconds.
The benefits of changing these parameters are dependent on the shared storage type and its performance.

Note The option values set in the STStartScriptsConfig file overwrite the ones in the *start_tm_console* file.

Problem: No alert for slow file listing

Problem Summary: When performing end user listing operations via HTTP, FTP or SSH on folders containing a large number of files, administrators cannot easily determine which account owns these files and the volume of data they occupy.

Solution: As of Update 5.5-20240829, administrators are notified via a WARN server log message when file listing takes longer than a specified time interval. This interval can be set using the `TransactionManager.LogSlowFileListingThreshold` option, which defines the maximum acceptable time (in milliseconds) for listing files within a user account. The default value is 10000 (10 seconds).
If the time exceeds this threshold, SecureTransport records a message in the Server log that looks like this:

```
Warn: [<LISTENER>] Slow file listing operation: Folder <DIR_NAME> listed by
account <ACCOUNT_NAME> with login name <LOGIN_NAME>, listed files: <N>,
duration: <MS> milliseconds.
```

Calculate the week number

In certain fields of the SecureTransport Administration tool, you can use the `date()` function to calculate the week number of the current month (WW) or the week number of the year (ww). For example, if you want to use a "week-year" or "week-month-year" string in a PTA, you can use an expression like

`${date("ww-yyyy")}` or `${date("WW-MM-yyyy")}`

When using the `date()` function for week-related calculations, be aware that week number is calculated differently based on the SecureTransport locale settings. The locale in use determines the week numbering system – the starting day of the week, when does the first week of a year start and whether partial weeks are allowed or not.

There are two main numbering systems:

- European week numbering system

  The week number is calculated according to ISO-8601 atop of the Gregorian calendar, where a week starts on Monday and a year has 52 or 53 full weeks.

- United States, Canada, Australia, and New Zealand

  The week starts on Sunday and the first week of the year starts on January 1st, which could create partial weeks at the start and end of the year. A year always has 53 weeks.

For example, January 1st, 2022 is a Saturday. The US calendar (en_US locale) will return week 1 (of 2022), but the week number of the year according to ISO-8601 will be 52 (of 2021).

To change the locale settings, add the following Java option in the `<FILEDRIVEHOME>/bin/start_tm_console` file:

```
JAVA_OPTS="-Duser.language=<language> -Duser.region=<region>
$JAVA_OPTS"
```

Troubleshoot "CMS parsing has failed"

As part of our ongoing effort to improve the performance of SecureTransport, we introduced the `AdvancedRouting.DontCopyPayload` server configuration option. Enabling it helps speed up the performance of the Advanced Routing application by not copying the file into the sandbox folder when there are no file transformations.

When this option is set to `true`, concurrent I/O operations performed on a file which is in transit may lead to specific error messages in the Server Log, which have the following format:

```
ERROR com.tumbleweed.st.server.tm.agents.SizeAgent SessionID-*** TransferID-*** -
Stream read/write error. Exception message is: CMS parsing has failed
```

These errors do not affect the integrity of the files and do not cause transfer failures.

To avoid such messages, disable the server configuration option.

Appendix C: FIPS transfer mode

For client-initiated file transfers using the AS2 (SSL), FTPS, HTTPS, PeSIT (SSL, legacy SSL), or SSH (SFTP and SCP) protocols, you can restrict the Axway SecureTransport Server to use only FIPS certified cryptographic libraries. This requires the sender and the recipient (clients and partner servers) to use only approved algorithms, ciphers, and cipher suites and assures that the entire transfer is secure at FIPS 140-2 Level 1.

For a complete list of supported ciphers and algorithms that be used in FIPS mode, see FIPS-compliant ciphers and cipher suites (login required).

Note Because Axway Secure Client firewall-friendly Tunnel Mode uses SSL v3, you cannot use it for FTPS in FIPS transfer mode.

For the relevant protocols, you can select **Enable FIPS Transfer Mode** in the *Server Control* page or the *Add Transfer Site* or *Edit Transfer Site* page.

As an administrator, you can customize the list of allowed TLS cipher suites or SSH ciphers and algorithms at the following levels:

- per protocol server (in the server settings). For more information about client-initiated transfers, see *Server control* on page 259.

  Note Enabling FIPS transfer mode for a protocol server causes transfers to fail if the client that uses that server does not provide the required FIPS cipher or cipher suite.

- for server-initiated transfers through a specific transfer site (in the transfer site configuration)
- for server-initiated transfers over a specific protocol (via dedicated server configuration options)

  For more information about server-initiated transfers, see *Manage transfer sites* on page 628.

  Note Enabling FIPS Transfer Mode for an existing transfer site causes transfers to fail if the other server does not provide the required cipher or cipher suite.

FIPS-certified cryptographic libraries

SecureTransport5.5 uses the following cryptographic library in FIPS transfer mode:

- BC-FJA (Bouncy Castle FIPS Java API) 1.0.2, FIPS 140-2 Certificate No. 3514

Appendix D: Command line utilities

The Axway SecureTransport Server includes several utility files that allow users to manually perform functions like installing a license file, stopping a server, and starting a server. These files are scripts on UNIX and batch files (`.bat`) on Windows. The utility files are located in the `<FILEDRIVEHOME>/bin` directory.

Utility files on page 1066 provides a summary of the utilities and their functions.

Note If an alias is used to install SecureTransport, use the `-A` option for any other commands for the server on UNIX. For example, to stop all servers for a SecureTransport installation with an alias as `myserver` on UNIX, use the following command:

```
./stop_all -A myserver
```

On Windows, you do not need to use the `-A` option. Instead, use the following command:

```
stop_all
```

The following topics describe how to control servers from the command line and list the command line utility files:

- *Control the servers* on page 1066 - Describes how to control servers from the command line.
- *Utility files* on page 1066 - Lists the command line utility files.

Control the servers

If a private key is specified for a server during installation and the server needs to be stopped or restarted, you must type the password for the private key. If the password is entered incorrectly during the boot process, use the Ctrl+C keyboard combination to stop the boot process and then restart the server.

Utility files

SecureTransport includes the utility files listed in the following table:

Utility file	Description
`bounce`	Bounces the SecureTransport protocol servers. If run on a primary server of a cluster, it will also bounce all the secondary servers.

Utility file	Description
`collect_support_information`	Collects information for Axway Global Support as specified on the *Support Tool Configuration* page.
`log_export`	Export server log or File Tracking (transfer log) entries from the database in CSV or DBF format.
`mkadmin`	Adds a new administrator account or changes the password of an existing administrator account. Changes are automatically synchronized across all servers in a Standard Cluster or Enterprise Cluster, unless the `-sync=n` option is present.
`repconv`	Updates repository encryption by decrypting files encrypted in a previous version of SecureTransport and encrypting them for the current version. Can also change the cipher algorithm and certificate SecureTransport uses to encrypt files and decrypt files and folders. **Note** On Windows systems, the repconv tool cannot process files with special characters in their names (such as Japanese, Chinese, or Cyrillic characters).
`show_ports`	Displays the ports on which the servers are configured to run.
Start scripts	
`start_admin`	Starts the Administration Tool server.
`start_all`	Starts all SecureTransport servers.
`start_as2d`	Starts the AS2 server.
`start_db`	Starts the embedded database server.
`start_ftpd`	Starts the FTP server.
`start_httpd`	Starts the HTTP server.
`start_pesitd`	Starts the PeSIT server.
`start_sshd`	Starts the SSH server.
`start_tm`	Starts Transaction Manager.
Stop scripts	
`stop_admin`	Stops the Administration Tool server.

Utility file	Description
`stop_all`	Stops all SecureTransport servers.
`stop_as2d`	Stops the AS2 server.
`stop_db`	Stops the embedded database server.
`stop_ftpd`	Stops the FTP server.
`stop_httpd`	Stops the HTTP server.
`stop_tm`	Stops Transaction Manager.
`stop_pesitd`	Stops the PeSIT server.
`stop_sshd`	Stops the SSH server.
Status scripts In order for the status scripts to work, both the admin server and the database must be up and running. If either of them is not functional, the status script will report the last recorded status in the service's "out" file, along with the statuses of all dependent services.	
`status_ftpd`	Checks the status of the FTP server.
`status_socks`	Checks the status of the SOCKS5 proxy. Only available on Edge.
`status_httpd`	Checks the status of the HTTP server. The following example is applicable to `status_ftpd` as well: you can expect the same messages with either in the place of `status_httpd`. • `status_httpd` returns `'httpd is disabled'`, when all http listeners are disabled. • `status_httpd` returns `'httpd is functional'`, when at least one http listener is enabled, and it is functional. • `status_httpd` returns `'httpd is down'`, when at least one http listener is enabled, and it is stopped or it is not functional.
`status_as2d`	Checks the status of the AS2 server.
`status_pesitd`	Checks the status of the PeSIT server.

Utility file	Description
`status_sshd`	Checks the status of the SSH proxy. The following example is applicable to `status_as2d` and `status_pesitd` as well: you can expect the same messages with either in the place of `status_sshd`. • `status_sshd` returns `'sshd is disabled'`, when all ssh listeners are disabled, and there are no functional listeners. • `status_sshd` returns `'sshd is functional'`, when at least one SSH listener is functional. • `status_sshd` returns `'sshd is down'`, when at least one SSH listener is enabled, and there are no functional SSH listeners.
`status_admin`	Checks the status of the Admin server.
`status_tm`	Checks the status of the Transactional manager.
`status_db`	Checks the status of the database.

Note SecureTransport includes several utilities that are used internally. All the utility files are stored in the `<FILEDRIVEHOME>/bin` or `<FILEDRIVEHOME>/bin/utils` directory.

Note In case of low disk space do not start servers.

Command line directory or file listing

This subsection discusses file listing specifics in SecureTransport and also supplies examples and usage details. Refer to the `ls` manual for the full list of command options and their usage.

The `ls -l` command is the standard file/directory listing command that returns all files and directories contained in the directory where you run it. You can also specify a *subdirectory listing*, using `ls -l <directory_name>` or a *file listing*, using `ls -l <file_name>`.

The returned output lists all contained files and directories in the following format:

```
[Permissions] [ST_nlink] [UID] [GID] [File_Size] [Date] [File_Name]
```

The following table includes the descriptions and specifics of each output parameter:

Utility file	Description
`Permissions`	The read-write-execute permissions for the current file.
`ST_nlink`	Represents the number of hard links to the file. Output of `ls -l <file_name>`, returns either as '`0`' or '`?`' for the `ST_nlink` value. This behavior applies to deployments on any supported OS.
`UID`	The User ID. When listing (file and folder) performed over SFTP on Windows, this value is always presented as a 0.
`GID`	The Group ID. When listing (file and folder) performed over SFTP on Windows, this value is always presented as a 0.
`File_Size`	The size of a file. Use the `Stfs.Encryption.ListDecryptedSize` server configuration option to specify which file size, the original (decrypted) or the encrypted file size, to be reported for encrypted files. The option is available in both SecureTransport Server and Edge, and should be set to the same value on both. The default value is `false`, meaning the encrypted file size is displayed. When the option is set to `true`, the original file size (taken from the STFS attribute) is reported. Due to the STFS attribute checks, performance degradation may occur.
`Date`	For a file, the returned value when that file was last modified; for a directory, it is the returned value for when that directory was created.
`File_Name`	The name and extension of the respective file.

A valid return line would look like this:

```
rw-rr--r-- 1 100 1001 1024 Aug 19 08:46 samplelog.txt
```

Appendix E: Server logs

The following topics lists the log files maintained by Axway SecureTransport and provide details for each log file:

- *Log file list* on page 1071 - Lists the log files maintained by SecureTransport.
- *Log output details* on page 1074 - Provides detailed descriptions of each log file maintained by SecureTransport.

Log file list

SecureTransport writes to several log files in multiple locations. These files can be used to monitor SecureTransport processes and identify any issues that can occur. Some messages are logged directly to the database and are visible on the *Server Log* page. For more information, see *Server log* on page 322.

You can open log files in a text editor to review them and to find specific messages.

The following table describes the log files used by SecureTransport and provides the location and a description of each file. Several log files share the same name. When viewing the table, make sure that you note the location of the log file you want to find.

File name	Directory	Description
`audit.log`	`<FILEDRIVEHOME>/var/logs/admin`	Records failed administrator login attempts, embedded database restart from Administrative tool, and database configuration change events for either the embedded or external database.
`catalina.out`	`<FILEDRIVEHOME>/tomcat/admin/logs`	Records unhandled exceptions in Java components in the Administration Tool and information about the servlets, including information about errors.
`catalina.out`	`<FILEDRIVEHOME>/tomcat/as2/logs`	Records AS2 protocol connection unhandled exceptions and information about the servlets, including information about errors.

File name	Directory	Description
cmdlog	<FILEDRIVEHOME>/var/logs	Records FTP commands and arguments that are sent by FTP clients after they connect successfully.
migration.log	<FILEDRIVEHOME>/var/logs	Records information generated during database migration from the embedded database to an Oracle database.
monitor_*.out	<FILEDRIVEHOME>/var/logs	Records monitor server output in a separate file for each monitored server.
mariadb_error.log mysql_error.log	<FILEDRIVEHOME>/var/logs	Records information about the embedded database for SecureTransport Edge and SecureTransport Server deployments that use an embedded database. Not used in deployments that use an external database.
mariadb_slow_query.log mysql_slow_query.log	<FILEDRIVEHOME>/var/logs	Lists queries that took longer than a given time to execute for SecureTransport Edge and SecureTransport Server deployments that use an embedded database. The file name and time limit are configurable in internaldb.conf for MariaDB and MySQL. Not used in deployments that use an external database.
serverlog-fallback.log	<FILEDRIVEHOME>/var/logs/admin	Records server log messages when the server log database fails for either the embedded or external database.
tm.stdout.log	<FILEDRIVEHOME>/var/logs	Records standard output from Transaction Manager processes.
tm_agent_error.log	<FILEDRIVEHOME>/var/logs	Records errors about Transaction Manager operations and in-process agents. On Unix-like systems, it also records external script's standard error (stderr).

File name	Directory	Description
tools.log	<FILEDRIVEHOME>/var/logs	Records warnings and errors from internal SecureTransport components.
AxwaySecureTransport Admin_ SecureTransport.log	<FILEDRIVEHOME>/../cygwin/var/log	A Windows-specific log file where information about the AxwaySecureTranport Admin_5.3.0 service is recorded.
AxwaySecureTransport AS2d_ SecureTransport.log	<FILEDRIVEHOME>/../cygwin/var/log	A Windows-specific log file where information about the AxwaySecureTranport AS2d service is recorded.
AxwaySecureTransport FTPd_ SecureTransport.log	<FILEDRIVEHOME>/../cygwin/var/log	A Windows-specific log file where information about the AxwaySecureTranport FTPd service is recorded.
AxwaySecureTransport HTTPd_ SecureTransport.log	<FILEDRIVEHOME>/../cygwin/var/log	A Windows-specific log file where information about the AxwaySecureTranport HTTPd service is recorded.
AxwaySecureTransport SSHd_ SecureTransport.log	<FILEDRIVEHOME>/../cygwin/var/log	A Windows-specific log file where information about the AxwaySecureTranport SSHd service is recorded.
AxwaySecureTransport PeSITd_ SecureTransport.log	<FILEDRIVEHOME>/../cygwin/var/log	A Windows-specific log file where information about the AxwaySecureTransport PeSITd service is recorded.
AxwaySecureTransport TM_ SecureTransport.log	<FILEDRIVEHOME>/../cygwin/var/log	A Windows-specific log file where information about the AxwaySecureTransportTM (Transaction Manager) service is recorded.

File name	Directory	Description
AxwaySecureTransport Database_ SecureTransport.log	<FILEDRIVEHOME>/../cygwin/var/log	A Windows-specific log file where information about the AxwaySecureTransportMYSQL or AxwaySecureTransportMARIADB(embedded database) service is recorded when it is used.
xferlog	<FILEDRIVEHOME>/var/logs	Records information about FTP(S), HTTP(S), SFTP, AS2, Connect:Direct, Folder Monitor, and PeSIT transfers. For more information, see *File Tracking* on page 302 and *Configure transfer log* on page 192.

Log output details

The following topics provide the log file output details:

- *Log4j files* on page 1074 - Provides the details of the log4j files.
- *Database log files* on page 1076 - Provides the details of database log files.
- *FTPD log file* on page 1077 - Provides the details of the FTPD log file.
- *Admin log file* on page 1077 - Provides the details of the admin log file.
- *General log files* on page 1077 - Provides the details of the general log files.
- *Log rotation and filtering* on page 1083 - Provides how-to instructions for changing the log4j files.
- *Redirect log4j output from the database* on page 1082 - Provides how-to instructions for redirecting the log4j output from the database.
- *Control log fallback from database to file* on page 1085 - Provides how-to instructions to control log fallback from the database to a file.
- *Server log rotation and monitor scheduling* on page 1086 - Provides how-to instructions for scheduling server log rotation.

Log4j files

A number of the log files and some log output to the database use the log4j format. For more information, refer to the log4j documentation on the Apache web site.

Unless otherwise specified, by default, the logs are in the following format:

```
%d %p [%t] %c - %m
```

where:

- `%d` is the date
- `%p` is the error level
- `%t` is the thread ID
- `%c` is the Java class name
- `%m` is the log message

You might also find Java stack traces in these logs. Axway Global Support can use these to determine the cause of a particular error condition.

The following table log4j configuration files in `<FILEDRIVEHOME>/conf`, the log output they control, and the default destinations.

Configuration file name	Log output	Destinations

`admin-log4j.xml`	Administration Tool server	Database
		`audit.log`
		`migration.log`

`as2d-log4j.xml`	AS2 Server	Database `xferlog`
`ftpd-log4j.xml`	FTP Server	Database `xferlog`
`httpd-log4j.xml`	HTTP Server	Database `xferlog`
`pesitd-log4j.xml`	PeSIT Server	Database `xferlog`
`socks-log4j.xml` (Only on SecureTransport Edge)	SOCKS5 proxy	Database
`sshd-log4j.xml`	SSH Server	Database `xferlog`
`tm-log4j.xml` (Only on SecureTransport Server)	TM Server internal agents	Database `xferlog`
`tools-log4j.xml`	Data migration Export/import Various components	Database Java console `tools.log`

In an Enterprise Cluster (EC), when the database does not accept log messages fast enough and the queue becomes full, the servers listed in the table store their log messages in a buffer file until the database can accept them. See *Control log fallback from database to file* on page 1085.

The behavior for each server is controlled by the following parameters in their respective log4j files:

- `queueAwaitDefaultTimeout`: Time in milliseconds to wait for the queue to free up when full (default 5000 milliseconds)
- `queueAwaitMinTimeout`: Minimum time in milliseconds of the queue wait period (default 50 milliseconds)
- `queueAwaitFactor`: Factor used to adjust queue wait time (default 1000). This value is divided by the number of events that have not been saved in the database, and the result is subtracted from the current timeout to get the time to wait until the next event is sent to database. If the result is less than the `queueAwaitMinTimeout` value, `queueAwaitMinTimeout` is used instead.

With a larger value of `queueAwaitFactor`, future events do not wait as long and the system is more responsive. With a smaller value, future events wait longer before they are sent to the database so the load on the database is reduced and the system response might be reduced.

next-event-await-period = maximum(`queueAwaitMinTimeout`, *last-event-await-period* - `queueAwaitFactor` / *number-of-events*)

With the default values for the parameters, the initial value of *next-event-await-period* is 5000 milliseconds. When there are 2 events that have not been saved to the database, the time to wait is reduced by 1000/2 = 500 milliseconds until it reaches 50 milliseconds.

When database communication returns to normal and the database starts to accept log messages again, *next-event-await-period* is reset to `queueAwaitDefaultTimeout` and *number-of-events* is reset to zero.

```
date time [process ID]: username: command
```

Database log files

These log files exist on SecureTransport Edge systems and on SecureTransport Server systems that use an embedded (MySQL or MariaDB) database.

mariadb_error.log / mysql_error.log – This log file contains the messages from the embedded database server. The log file contains information indicating when `mariadbd / mysqld` was started and stopped and also any critical errors that occur while the server is running. For more information on MySQL, refer to the documentation of the error log on the MySQL Developer Zone website.
For more information on MariaDB, refer to the official documentation of the error log on the MariaDB website.

mariadb_slow_query.log / mysql_slow_query.log – This log file contains SQL statements that took more than `long_query_time` seconds to execute. For more information on MySQL, refer to the documentation of the slow query log on the MySQL Developer Zone website. For more information on MariaDB, refer to the official documentation of the error log on the MariaDB website.

FTPD log file

cmdlog – This log file contains the FTP commands and arguments that are sent by FTP clients after they connect successfully. (Thus, it does not show the USER and PASS commands.) Note that this log is not enabled by default. For more information, see *Configure FTP command log* on page 188.

The format of this file is:

```
<calendar_time> <PID> <user_name> <command>
```

Where:

`<current_time>` - The current local time, for example Tue Feb 24 14:28:01

`<PID>` - The process ID

`<user_name>` - Account name

`<command>` - FTP command and (optional) target file name

The possible FTP commands are listed under `Ftp.Commands` configuration option.

The default path of the **cmdlog** file is `<filedrivehome>/var/logs` and is configurable trough the `Ftp.CommandLogging.File` configuration option.

Admin log file

audit.log – This log file contains information on configuration changes made by Java components of the SecureTransport Administration Tool and database starts and stops. When the configuration changes are logged to the database (the default), failed administrator login attempts and database starts, stops, and configuration changes are still logged to this file.

The format and content of this file is controlled by the `<FILEDRIVEHOME>/conf/admin-log4j.xml` file. This file uses the log4j format. By default, the logs are in the following format:

```
%d %s %m
```

Where:

- `%d` is the date
- `%s` is the subcomponent
- `%m` is the log message

admin_tomcat<date>.log – This log file contains Tomcat-specific error messages. The file contains the Java stack traces for the errors.

General log files

This section presents the `xferlog`, `xferlog-advanced` and `tools.log` files.

xferlog

This log file contains information about uploads and downloads for all protocols. It records information about all the AS2, FTP(S), HTTP(S), PeSIT, SFTP, Connect:Direct, and Folder Monitor transfers. In a deployment consisting of both SecureTransport Server and Edge, information for client-initiated transfers is logged in the Edge `xferlog` file, and information for server-initiated transfers is logged in the Server `xferlog` file. In a deployment consisting of only SecureTransport Servers, all transfer information is recorded in the Server `xferlog` file, as well as in the database.

The arrival of a PeSIT message does not create a new entry in the `xferlog` file.

SecureTransport Update 5.5-20230928 adds a new server configuration option, `Xferlog.Tm.Advanced.Enabled`, with a default value of **false**. If changed to **true**, information for all client- and server-initiated transfers will be logged in the Server `xferlog` file on both types of deployments. This option also introduces a new logging format with three additional parameters. Since the Transaction Manager uses the new format but the protocols do not, we recommend that you have two separate `xferlog` files: one for the protocols, and one for the Transaction Manager. For more details, see *xferlog-advanced* on page 1081.

Note On Windows, it is not possible to modify existing cron jobs or to use cron to run any jobs other than those SecureTransport jobs documented in these topics. Instead, use the Windows Task Scheduler.

By default, the separator character in the `xferlog` file is a " " (space). To avoid breaking the external parsers when the name of a transferred file contains spaces, the delimiter can be changed to a custom character of your choice.

To configure your own delimiter:

1. Add `delimiter=<value>` in the `XferLogLayout` in all log4j files (`ftpd-log4j.xml`, `as2d-log4j.xml`, `sshd-log4j.xml`, `pesitd-log4j.xml`, `httpd-log4j.xml`, and `tm-log4j.xml`).
2. Restart the services after saving your changes.

In the following example, the delimiter is changed to " ; ":

```
<XferLogAppender name="XferLogAppender
fileName="FILEDRIVEHOME/var/logs/xferlog" append="true">
      <XferLogLayout dateFormat="yyyy-MM-dd HH:mm:ss,SSS"/>
</XferLogAppender>

<XferLogAppender name="XferLogAppender"
fileName="FILEDRIVEHOME/var/logs/xferlog" append="true">
     <XferLogLayout delimiter=" ; " dateFormat="yyyy-MM-dd HH:mm:ss,SSS"/>
</XferLogAppender>
```

Each server entry is composed of a single line as shown below. All fields are separated by spaces.

```
<current_time> <transfer_time> <remote_host> <file_size> <file_name> <transfer_
```

```
mode> <transfer_security> <transfer_status> <access_mode> <user_name> <server_
name> 0 *
```

The fields represent the following:

- <current_time> – Current local time in the format *yyyy-MM-dd HH:mm:ss,SSS*, where
  - *yyyy* – Year
  - *MM* – Month
  - *dd* – Day of the month
  - *HH* – Hour
  - *mm* – Minutes
  - *ss* – Seconds
  - *SSS* – Milliseconds

    Note As of SecureTransport Update 5.5-20230928, the time zone can be included as either ISO 8601 (X), RFC 882 (Z), or General Time Zone (z). For example, to display the time zone as 2023-09-18 13:08:18,931 +03:00, add XXX to the date format:

    ```
    <XferLogAppender name="XferLogAppender
    fileName="FILEDRIVEHOME/var/logs/xferlog" append="true">
    <XferLogLayout dateFormat="yyyy-MM-dd HH:mm:ss,SSS XXX"/>
    </XferLogAppender>
    ```

- <transfer_time> – Total time for the transfer, rounded off to seconds.
- <remote_host> – Remote host name or IP address of the client.

  For HTTP client-initiated transfers performed behind a proxy or a load balancer with a configured X-Forwarded-For header, the <remote_host> displays the IP address of the proxy/load balancer, while the client IP address is positioned at the end of the record (see <X-Forwarded-For>).
- <file_size> – Number of bytes transferred.
- <file_name> – (for example: /home/jdoe/somefile): For virtual and anonymous users, the path given is relative to their home directory. For real users, it's relative to the filesystem root.

  Note If you use a File Download or File Upload agent (such as the streaming agents) to handle the file transfer, the path will be preceded by STOR: (for uploads) or RETR: (for downloads).

- <transfer_mode> – A single character indicating the type of transfer:
  - a – ASCII transfer
  - b – Binary transfer

- `<transfer_security>` – A single character indicating the level of security:
  - `s` – Secure (SSL-based)
  - `n` – Non-secure

    Note In the 2.x versions of SecureTransport (and the 2.x and 3.x versions of SecureTransport for Windows), the value of this field is always _ regardless of whether security was used.
- `<transfer_status>` – A single character indicating the upload or download status:
  - Uploads – `i` for OK, `j` for error, and `k` for aborted
  - Downloads – `o` for OK, `p` for error, and `q` for aborted

    Note Under HTTP/S and SSH , user aborts are treated as errors. The reason is that for FTP, the abort condition is indicated by an explicitly received ABOR command. Any other data socket reset is considered a failure. Since SSH and HTTP don't have a control connection, they resort to interpreting a socket reset as a failure. There isn't any trace of an abort being present for SSH and HTTP protocol daemons. This also includes the guaranteed delivery extension for HTTP. Therefore, the client side aborts are displayed in the `xferlog` (and FileTracking) as inbound and outbound errors rather than inbound and outbound aborts. For SSH uploads there is an additional peculiarity: With a native SFTP Linux client, unless a kill ABRT of the child SSH process is completed, a transfer interrupt on the client side with Ctrl +C would cause `SSH_FXP_CLOSE` to be sent to SecureTransport, indicating FULL "successful" transfer. In this case, an "interrupted " SSH client transfer will be shown in FileTracking and the `xferlog` as successful.
- `<access_mode>` – A single character indicating the type of user access:
  - `a` - Anonymous
  - `r` – Real or virtual user
- `<user_name>` – E-mail address, as given at the password prompt, for anonymous users (for example: `jdoe@foo.com`); username for real or virtual users (for example: `jdoe`).
- `<server_name>` – Type of server used to make the transfer (transport protocol):
  - `ad-hoc`
  - `as2`
  - `c:d` (Connect:Direct)
  - `folder` (Folder Monitor)
  - `ftp`
  - `http`
  - `pesit`
  - `ssh`
- A zero (`0`)

- An asterisk (`*`)
- `<X-Forwarded-For>` (when the header is set) - The client IP address. It is displayed in the log record when performing HTTP client-initiated transfers behind a proxy/load balancer on both SecureTransport Edge and Server. This parameter is not available in the *xferlog-advanced* on page 1081 file.

Note The zero and asterisk are inserted as authentication method and authenticated user id respectively for compatibility with the `wu-ftp` log format.

The following is an example of a typical log entry:

```
Wed Oct 18 10:55:13 2023 3 10.191.2.33 5873
/drives/c/home/Virtual/vuser/avatar.jpg b s i r vuser http 0 *
```

Note On Windows, it is not possible to modify existing cron jobs or to use cron to run any jobs other than those SecureTransport jobs documented in these topics. Instead, use the Windows Task Scheduler.

xferlog-advanced

As of SecureTransport Update 5.5-20230928, a separate Transaction Manager `xferlog` file can be used to log three more parameters in addition to the ones listed above:

- `<Core_ID>` - The Core ID of the transfer.
- `<Transfer_ID>` - The Transfer ID of the transfer.
- `<transfer_initiator>` - The initiator of the transfer, either client (`c`) or server (`s`).

Use the following procedure to configure an `xferlog` file for the Transaction Manager:

1. In the `<FILEDRIVEHOME>/conf/tm-log4j.xml>` file (in a cluster, on every server), enter a new `fileName`. For example, *xferlog-advanced*:

   ```
   <XferLogAppender name="XferLogAppender"
   fileName="/opt/TMWD/SecureTransport/var/logs/xferlog-advanced"
   append="true">
   <XferLogLayout dateFormat="yyyy-MM-dd HH:mm:ss,SSS"/>
   </XferLogAppender>
   ```

2. In the SecureTransport Administration Tool, go to **Operations > Server Configuration** and change the `Xferlog.Tm.Advanced.Enabled` server configuration option to **true**.
3. Restart all SecureTransport services. In a cluster environment, they must be restarted on every node.

The following is an example of a typical log entry enhanced with the additional parameters:

```
Wed Oct 18 10:55:13 2023 3 10.191.2.33 5873
/drives/c/home/Virtual/vuser/avatar.jpg b s i r vuser http 0 * d836a9e9-ea44-
4764-a7d5-77a3ce7a2cbf d05b3d49-f5ca-466b-adc0-c97c35ee46eb c
```

Note The list of additional parameters in the `xferlog-advanced` file may be expanded with future SecureTransport Updates.

tools.log

This log4j-format log file records warnings and errors from internal SecureTransport components. The format and content of this file is controlled by the `<FILEDRIVEHOME>/conf/tools-log4j.xml` file.

For example, importing accounts using a command-line tool can produce the following message:

```
2010-11-16 00:56:14,505 PST WARN [main]
com.tumbleweed.st.server.appframework.sql.SessionFactoryManagerImpl - Component
type '' is unknown. Using TOOLS configuration.
```

Redirect log4j output from the database

As listed in *Log4j files* on page 1074, some log output is directed to both the `xferlog` file and the database. It's not advisable to log to the database for log level `debug` or `all` for an extensive period of time because SecureTransport will produce too many log messages which can overload the database.

Note The *Server Log* page shows only the server log messages that are stored in the database. When you store the log messages in a file, they are not displayed on the page.

To direct the log messages being stored in the database to a file, modify the `ServerLog` appender for each relevant file as shown in the example below. In the XML configuration file, replace `FILEDRIVEHOME` with the actual installation path to the SecureTransport home folder.

Change:

```
<STDBAppender name="ServerLog" locationInfo="true"
maxLoggingEventQueueSize="10000" fallbackLogger="ServerLogFallback"
databaseStatusCheckupDelay="60" databaseCheckupTimeout="30" idAreaBegin="-
210000000000" idAreaEnd="-110000000000" driverClass="org.mariadb.jdbc.Driver"
initialPoolSize="30"
maxPoolSize="100" minPoolSize="30" acquireIncrement="7" maxStatements="4000">
      <Filters>
         <STLog4JNDCFilter componentName="TM" onMatch="NEUTRAL"
onMismatch="DENY"/>
      </Filters>
</STDBAppender>
```

Note The `dataSource` settings are different for each configuration file.

To:

```
<DailyRollingFileAppender name="ServerLog"
fileName="FILEDRIVEHOME/var/logs/tm.log" append="true"
rotateDirectory="FILEDRIVEHOME/var/db/hist/logs/" datePattern="'.'yyyy-MM-dd">
      <PatternLayout pattern="%d{ISO8601} [%t] %p %c %equals{%x}{[]}{} -
%m%n%ex"/> </DailyRollingFileAppender>
```

Note If the Server log messages are stored in a file (and it is using the DailyRollingFileAppender), rotation will be handled by the log4j files. See sample configuration.

For example, if the File option is set to /foo/bar.log and the DatePattern set to '.'yyyy-MM-dd, on 2020-02-16 at midnight, the logging file /foo/bar.log will be copied to /foo/bar.log.2020-02-16 and logging for 2020-02-17 will continue in /foo/bar.log until it rolls over the next day.

In the `admin-log4j.xml` file there is an additional appender called `AuditLogAppender`. Change appender from:

```
<STDBAppender name="AuditLogAppender" locationInfo="true"
maxLoggingEventQueueSize="10000" fallbackLogger="ServerLogFallback"
databaseStatusCheckupDelay="60" databaseCheckupTimeout="30" idAreaBegin="-
40000000000" idAreaEnd="-30000000000" driverClass="org.mariadb.jdbc.Driver"
initialPoolSize="5"
maxPoolSize="25" minPoolSize="5" acquireIncrement="5" maxStatements="4000"
      <Filters>
         <STLog4JNDCFilter componentName="AUDIT" onMatch="NEUTRAL"
onMismatch="DENY"/>
      </Filters>
</STDBAppender>
```

To:

```
<DailyRollingFileAppender name="AuditLogAppender"
fileName="FILEDRIVEHOME/var/logs/admin/audit-menu.log" append="true"
rotateDirectory="FILEDRIVEHOME/var/db/hist/logs/admin" datePattern="'.'yyyy-MM-
dd">
      <PatternLayout pattern="%d{ISO8601} [%t] %p %c %equals{%x}{[]}{} -
%m%n%ex"/>
</DailyRollingFileAppender>
```

Log rotation and filtering

This section illustrates how to modify log4j files to configure log rotation or add filters.

Configure log file rotation based on size and time

When logging information is redirected to a file, you can fine-tune the logging mechanism by modifying the log4j.xml configuration files. In the *Redirect log4j output from the database* on page 1082 topic, we demonstrate how to use SecureTransport's custom *DailyRollingFileAppender* which

rolls over log files based on time.
Here's the snippet of the TM log file (`<FILEDRIVEHOME>/conf/tm-log4j.xml`) that is configured for daily rolling log files:

```
<DailyRollingFileAppender
name="ServerLog" fileName="/actual/path/to/logs/tm.log" append="true" datePattern="'.'yyyy-
MM-dd">
    <PatternLayout pattern="%d{ISO8601} [%t] %p %c %equals{%x}{[]}{} - %m%n%ex"/>
</DailyRollingFileAppender>
```

If you want to roll over log files based on file size and date time, instead of *DailyRollingFileAppender*, you need to use *RollingFileAppender* with both time and size based triggering policies. The code snippet bellow is a *RollingFileAppender* that writes log messages to the *tm.log* file, rolls over the file at midnight every day and when the file size reaches 1000 MB, and deletes roll over log files older than 5 days.

Note SecureTransport environment variables, such as `sessionId` or `transferId`, cannot be used in the Pattern Layout part of *RollingFileAppender*.

```
<RollingFile
name="ServerLog"
fileName="FILEDRIVEHOME/var/logs/tm.log"
append="true"
filePattern="FILEDRIVEHOME/var/db/hist/logs/%d{yyyy-MM-dd}.%i.tm.log">
  <PatternLayout pattern="%d{ISO8601} [%t] %p %c %equals{%x}{[]}{} - %m%n%ex"/>
  <DefaultRolloverStrategy max="1000">
    <Delete basePath="FILEDRIVEHOME/var/db/hist/logs/" maxDepth="2">
      <IfFileName glob="*.tm.log" />
      <IfLastModified age="5d" />
    </Delete>
  </DefaultRolloverStrategy>
  <Policies>
     <TimeBasedTriggeringPolicy />
     <SizeBasedTriggeringPolicy size="1000 MB"/>
  </Policies>
</RollingFile>
```

You can use any of the log4j file appenders. For more information, refer to the Log4j documentation.

For another way to manage log file rotation, see *Log Entry Maintenance application* on page 839.

Attach logger to a non-blocking asynchronous appender

In some cases when the log messages are directed to a flat file, the rotation of the file may take a considerable amount of time during which certain SecureTransport processes (including user logins) are unavailable. This can happen, for example, when rotating a large log to a file hosted on a remote location. In this case, consider attaching the rotating log to an asynchronous appender which is explicitly defined as non-blocking.

In the example below, the *ServerLog* asynchronous appender points to the *DailyRollingFileAppender* via the *DailyAppender* reference.

```
<Async name="ServerLog" bufferSize="20000" blocking="false" includeLocation="false">
    <AppenderRef ref="DailyAppender"/>
</Async>

<DailyRollingFileAppender
name
="DailyAppender"

fileName
="FILEDRIVEHOME/var/logs/tm.log"
 append="true" rotateDirectory="FILEDRIVEHOME/var/db/hist/logs/" datePattern="'.'yyyy-MM-dd">
    <PatternLayout pattern="%d{ISO8601} [%t] %p %c %equals{%x}{[]}{} - %m%n%ex"/>
</DailyRollingFileAppender>
```

Note The `bufferSize` parameter indicates the maximum number of log entries stored in memory during log file rotation. In case the limit is reached, new log messages will be discarded.
Set the buffer size according to the size of your log - it is recommended that you multiply the number of expected messages by 20. For example, if the number of log messages received by the server during rotation is 1000, set the buffer size to 20000.

Configure Filters

You can use the log4j2 filters to specify if or how a log event should be published. A filter can be configured for the entire configuration or at logger or appender level. For more information, see Apache Log4j 2 documentation.

The example below shows how to filter out unwanted plugin messages that contain the string `DONOTSHOWME`. In it, we add a RegexFilter filter in the Logger element in `<FILEDRIVEHOME>/conf/admin-log4j.xml` file.

```
<Logger name="com.axway.st.server.plugins" level="INFO" additivity="false">
    <AppenderRef ref="ServerLog"/>
    <Filters>
        <RegexFilter regex=".DONOTSHOWME." onMatch="DENY" onMismatch="ACCEPT"/>
    </Filters>
    <!-<AppenderRef ref="Stdout" />->
    <!-<AppenderRef ref="Stderr" />->
</Logger>
```

Control log fallback from database to file

If the embedded or external database stops accepting log messages, SecureTransport directs the messages to `<FILEDRIVEHOME>/var/logs/admin/serverlog-fallback.log`.

You can control this behavior by setting the following parameters for each server in `<FILEDRIVEHOME>/conf/configuration.xml`:

- When the embedded database is used, the `hibernate.c3p0.timeout` attribute of the component for each server controls the timeout after which an idle connection will be removed from the pool. The default value is 30 minutes.
- When an external Oracle database is used, the `hibernate.connection.oracle.jdbc.ReadTimeout` attribute of the component for each server controls the read timeout, how long to wait for a response from the database before failing a query, for all TCP sockets to the database. The default is five minutes.
- When an external database is used, the `hibernate.c3p0.checkoutTimeout` attribute of the component for each server controls the Database connect timeout, how long to wait for a connection to be established. The default is five minutes.

Server log rotation and monitor scheduling

Note For another way to schedule server log rotation, see *Transfer Log Maintenance application* on page 856 and *Log Entry Maintenance application* on page 839.

Log files are rotated so that you do not lose any information because you have reached a file size limit. The log rotation schedule for log4j files is specified in the log4j configuration files. All other log files are rotated on a regular schedule as directed by a log rotation scheduling tool.

To update schedule of monitor and rotate scripts, the administrator must edit the `<FILEDRIVEHOME>/conf/monitor.schedule.properties` file.

The syntax for schedule is:

Field name	Mandatory	Allowed values	Allowed special characters
Seconds	yes	0-59	, - * /
Minutes	yes	0-59	, - * /
Hours	yes	0-23	, - * /
Day of month	yes	1-31	, - * ? / L W
Month	yes	1-12 or JAN-DEC	, - * /
Day of week	yes	1 to 7 or SAT-SUN	, - * ? / L #
Year	no	Empty, 1970-2099	, - * /

Example usage of special characters:

- * ("all values") – represents all values within a field. For example, * in the minute field means "every minute".
  Example use: * * * * * * * – will fire the job every second, every minute, every hour, every day, every day-of-week, every month, every year.
- ? ("no specific value") – use this operator when you don't want to specify a day of month or day in week; it basically means "any", as in "any day-of-week".
  Example use: * * * * * ? * – will fire that job every second, every minute, every hour, every day, every month, any day-of-week, every year.
- - – use the hyphen to specify ranges. For example, "1-3" in the day field means "first, second and third day in the month".
  Example use: * * * 1-3 * ? * – will fire that job every second, every minute, every hour, days 1, 2 and 3 of the month, every month, any day-of-week, every year.
- , – use the comma as a divider between values. For example, "TUE, THU" in the day-of-week field means "On Tuesday and Thursday".
  Example use: * * * * * TUE, THU * – will fire that job every second, every minute, every hour, any day of the month, every month, Tuesday and Thursday, every year.
- / – use the slash to specify increments. For example, "0/10" in the seconds field means "the seconds 0, 10, 20, 30, 40 and 50". And "5/15" in the seconds field means "the seconds 5, 20, 35, and 50". You can also specify '/' after the '' character – in this case '' is equivalent to having '0' before the '/'. '1/3' in the day-of-month field means "fire every 3 days starting on the first day of the month".
  Example use: * 15/30 * * * * * – will fire the job every second, every 15^{th} and 45^{th} minute in the hour, every hour, every day, every month, every day-of-week, every year.
- L ("last") – is contextual to each of the two fields in which it is allowed. With day-of-month, "L" means "the last day of the month": so its 31^{st} for January, 28^{th} for February (on non-leap years). With the day-of-week field by itself, it simply means the 7^{th} day, which is "SAT".
  Note: If used in the day-of-week field after another value, it means "the last xxx day of the month" – for example "6L" means "the last Friday of the month". You can also specify an offset from the last day of the month, such as "L-3" which would mean the third-to-last day of the calendar month. Do not specify lists, or ranges of values when using the 'L' option.
  Example use: * * * L * * * – will fire the job every second, every minute, every hour, every day, last day of every month, every month, every day-of-week, every year.
- W ("weekday") – specifies the weekday (any day, Monday to Friday) that is nearest to the given day. For example, "8W" for the day-of-month field, means: "the nearest weekday to the 8^{th} of the month". If the 8^{th} is on Saturday, then Friday the 7^{th} is closest. If the 8^{th} happens to be a Sunday, the trigger will fire on Monday the 9^{th} as it is closer to Sunday.
  Note: If you specify "1W" as the value for day-of-month, and the 1^{st} happens to be a Saturday, the trigger will fire on Monday the 3^{rd}, and not Friday, as it is in fact the last day of the previous month. Do not specify lists, or ranges of values when using the 'W' option.
  Example use: * * * 10w * * * – will fire the job every second, every minute, every hour, every 10^{th} day of month (or closest weekday if 10^{th} is a weekend day), every month, any day-of-week (that matches the other criteria), every year.
- # – specifies "the n^{th}" XXX day of the month but is used in context with the respective values. For example, "6#3" in the day-of-week field means "the third Friday of the month" (day 6 = Friday and "#3" = the 3^{rd} one in the month). Other examples: "2#1" = the first Monday of the month

and "4#5" = the fifth Wednesday of the month.
Note: If you specify "#5" and there is not 5 of the given day-of-week in the month, then no firing will occur that month.
Example use: * * * * * 1#3 * – will fire the job every second, every minute, every hour, any day of the month, every month, any day-of-week, every third Sunday of the month, every year.

Appendix F: Firewall settings

The SecureTransport client and server software is frequently used in conjunction with network firewalls, designed to permit access only to authorized protocols and possibly specific source networks, hosts, or users. Firewalls provide essential protection from hackers, corporate raiders, and other types of access that are unauthorized or should be restricted.

A firewall refers to a hardware, software, or combination product that provides stateful inspection or filtering functionality only. There are other types of network devices, such as proxy servers and caching stores that can provide these functions as well, but these devices are not covered in the following topics.

One frequently deployed firewall is the Check Point FireWall-1 product (VPN-1), produced by Check Point Software Technologies Ltd. Other popular products include the Cisco PIX, Symantec Enterprise Firewall (Raptor), and Network Associates Gauntlet firewalls.

The following topics provide how-to instructions for enabling bidirectional connections and configuring ports. They also provide descriptions and lists of the firewall rules.

- *Configure firewall ports* on page 1090 - Provides how-to instructions for configuring firewall ports.
- *Firewall rules* on page 1093 - Provides descriptions and lists of the firewall rules.

Additional notes:

Bidirectional FTP data connections

If you are using FTP for file transfers, you might need to configure your firewall to use bidirectional FTP transfers. Bidirectional FTP data connections might not be enabled in your firewall by default. Bidirectional FTP data connections are not considered as safe since the data connection is interactive and the connection changes the basic way FTP works.

Cisco PIX firewall

Cisco PIX firewalls do not interoperate properly with SecureTransport FTP connections when "stateful inspection" is enabled for the FTP protocol. Disable stateful inspection for the FTP protocol accordingly.

Check Point firewall

To enable Check Point firewalls, in Check Point NG firewalls (AI R55 and higher), set the FTP connection to FTP_BASIC. This allows bidirectional communications and sets the firewall to allow commands not terminated with a newline.

The Check Point firewall must allow bidirectional communication. It must not enforce new line termination.

As the Check Point documentation states, the FTP_BASIC protocol type was introduced in the Check Point R55 NG AI. If you apply it to the FTP object, it enforces a reduced set of the FTP security checks done by the regular FTP protocol type. The following checks are not enforced by FTP_BASIC:

- That every packet is terminated with a newline character, so that the PORT command is not split across packets.
- Bidirectional traffic on the data connection is not allowed, as it can be used improperly.

Note FTP_BASIC also disables FTP BOUNCE protection and so can be viewed as potentially less secure. This, as well as a more secure method to disable the newline check (FTP PACKET check) are described in Check Point Secure Knowledge SK27122.

Configure firewall ports

The exact list of ports to open depends on which SecureTransport functions you use. For example, if you do not enable FTP connections to your server, then you can disregard the ports listed below for FTP.

The following lists show the ports that SecureTransport can use. The values given are the defaults. You can reconfigure SecureTransport after installation to use different values.

The following topics list firewall ports that need to be open:

- *Communication between the outside and SecureTransport Edge* on page 1091 - Lists the firewall ports that need to be open for communication between the outside and SecureTransport Edge.
- *Communication between SecureTransport Server and SecureTransport Edge* on page 1091 - Lists the firewall ports that need to be open for communication between SecureTransport Server and SecureTransport Edge.
- *Communication between SecureTransport Server and an internal network* on page 1092 - Lists the firewall ports that need to be open for communication between SecureTransport Server and an internal network.
- *Internal SecureTransport communication* on page 1092 - Lists the firewall ports that need to be open for internal SecureTransport communication.

Communication between the outside and SecureTransport Edge

These ports must be opened on your external firewall to allow communication between the outside world and your proxy (SecureTransport Edge) server.

- 20 – FTP (secure and non-secure) active-mode data channel
- 21 – FTP (secure and non-secure) control channel (For secure connections, the firewall must allow bidirectional communication.)
- 22 – SSH (SFTP and SCP)
- 80 – HTTP
- 443 – HTTPS
- 444 – Administration Tool (HTTPS)
- 10080 – AS2 (non-SSL)
- 10443 – AS2 (SSL)
- 17617 – PeSIT (non-SSL)
- 17627 – PeSIT over secure socket (non–Transfer CFT compatible)
- 17637 – PeSIT over secure socket (CFT compatible)
- 19617 - PeSIT over pTCP plain socket
- 19627 - PeSIT over pTCP Secured Socket
- User-defined range – FTP (secure and non-secure) passive-mode data channel

Communication between SecureTransport Server and SecureTransport Edge

These port must be open between the private network and the peripheral network (DMZ) for the Transaction Manager Server on SecureTransport Server to connect to the protocol servers on SecureTransport Edge. The protocol is the SecureTransport secure streaming protocol.

- 20021 – FTP Server
- 20022 – SSH Server
- 20080 – HTTP Server
- 20444 – Administration Tool server
- 21080 – AS2 Server
- 27617 – PeSIT Server

Communication between SecureTransport Server and an internal network

All of the ports used for communication between the outside and SecureTransport Edge can be used for user connections originating within your internal network. The following ports might be used for interfacing with infrastructure components:

- 389 or 3268 – LDAP
- 1305 – Axway Sentinel
- 1344 - ICAP
- 1433 - Microsoft SQL Server database (default)
- 1521 – Oracle database (default)
- 44441 – SiteMinder Accounting
- 44442 – SiteMinder Authentication
- 44443 – SiteMinder Authorization

Internal SecureTransport communication

- 7800 thorough 7802 – Hibernate second-level cache
- 8005 – Tomcat shutdown
- 8006 – AS2 shutdown
- 8009 – Tomcat JK connector
- 33060 – MySQL or MariaDB database
- 44431 – Standard Cluster (SC) server synchronization and heartbeat

Ports used for communication outside the firewall (for the servers that you plan to use), might need firewall rules set up so that they are accessible only from certain subnets. For example, allowing internal users to connect using both plain and secure HTTP, while requiring external users to use HTTPS, by opening port 80 only for a certain subnet, while keeping 443 open unconditionally.)

During SecureTransport installation, each daemon is assigned a JMX port from 9994 to 9998. These ports are used for communication between the admin and the protocol daemons; for example, daemon shutdown, configuration change pushes and daemon status checks from the Administration Tool.

- 9994 – AS2 daemon
- 9995 – SSH daemon
- 9996 – FTP daemon
- 9997 – HTTP daemon
- 9998 – PeSIT daemon

The properties used for JMX communication, including the default port numbers, are specified in the `<DaemonsConfigurationCommunication>` element of the `<FILEDRIVEHOME>/conf/configuration.xml` file. You can edit this file to change the default port number. After the daemon configuration has been modified, the daemon must be restarted to activate the changes. It's advisable to check the new port number with netstat.

Firewall rules

You can use the firewall rules in the following tables for active/active (load-balanced) or active/passive (failover) Standard Clusters and Enterprise Clusters.

Note For a non-streaming deployment with active/active or active/passive systems, skip the rules for port 1080. For a non-streaming setup with only a single machine, also skip the Standard Clustering rules. The Destination Port values listed are the default values or ranges used by SecureTransport.

The following topics list the firewall rules:

- *Protocol rules* on page 1093 - Lists the firewall protocol rules.
- *Authentication rules* on page 1096 - Lists the firewall authentication rules.
- *Administration rules* on page 1097 - Lists the firewall administration rules.
- *TM server communication rules* on page 1097 - Lists the firewall Transaction Manager server communication rules.
- *Server transfer rules* on page 1098 - Lists the firewall server transfer rules.
- *Standard Cluster rules* on page 1098 - Lists the firewall Standard cluster rules.
- *Enterprise Cluster rules* on page 1099 - Lists the firewall Enterprise Cluster rules.
- *Protocol rules - outbound from SecureTransport Edge* on page 1099 - Lists the firewall protocol rules for outbound communication from the SecureTransport Edge.

Protocol rules

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
1	Internet	ST Edge Virtual IP (load balancer)	FTP(S)	21	← (inbound)	Client FTP(S) control channel

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
2	Internet	ST Edge Virtual IP (load balancer)	FTP(S)	1024 to 65535 (Actual port range is specified on the Remote Server.)	→ (outbound)	Client FTP(S) data channel, active mode
3	Internet	ST Edge Virtual IP (load balancer)	FTP(S)	1024 to 65535 (Actual port range is specified on the Edge.)	← (inbound)	Client FTP(S) data channel, active mode
4	Internet	ST Edge Virtual IP (load balancer)	HTTP	80	← (inbound)	Client web access (non-secure)
5	Internet	ST Edge Virtual IP (load balancer)	HTTPS	443	← (inbound)	Client web access (secure)
6	Internet	ST Edge Virtual IP (load balancer)	SSH	22	← (inbound)	Client SSH access
7	Internet	ST Edge Virtual IP (load balancer)	AS2 (SSL)	10443	← (inbound)	Partner AS2 access (secure)
8	Internet	ST Edge Virtual IP (load balancer)	AS2 (non-SSL)	10080	← (inbound)	Partner AS2 access (non-secure)

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
9	Internet	ST Edge Virtual IP (load balancer)	PeSIT over plain socket	17617	← (inbound)	Partner PeSIT access (non-secure)
10	Internet	ST Edge Virtual IP (load balancer)	PeSIT over secured socket (non–Transfer CFT Compatible)	17627	← (inbound)	Partner PeSIT access (non-secure)
11	Internet	ST Edge Virtual IP (load balancer)	PeSIT over secured socket (Transfer CFT Compatible)	17637	← (inbound)	Partner PeSIT access (non-secure)
12	Internet	ST Edge Virtual IP (load balancer)	PeSIT over pTCP plain socket	19617	← (inbound)	Partner PeSIT access (non-secure)
13	Internet	ST Edge Virtual IP (load balancer)	PeSIT over pTCP secured socket	19627	← (inbound)	Partner PeSIT access (non-secure)
14	Trusted (secure network)	ST Edge	HTTPS	444	← (inbound)	SecureTransport Administration Tool (if access is required through the firewall)
15	ST Server	Mail server for outgoing mail	SMTP (TCP)	25	→ (outbound)	Ad hoc file transfer email notifications from ST Web Client

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
16	ST Server	SNMP Masters	SNMP (UDP)	162	➔ (outbound)	SNMP monitoring

For a streaming deployment with one SecureTransport Edge and one SecureTransport Server there is no load balancer, so substitute the real IP address of the SecureTransport Edge for the IP address of the load balancer in the Group Destination column for rules 1 through 13.

For outbound AS2 transfers or asynchronous MDN receipts for inbound AS2 transfers, define outbound rules for ports 10080 and 10443. If the AS2 listener is on the SecureTransport Server and an SOCKS5 proxy is on the SecureTransport Edge, define these rules on the firewall between the SecureTransport Edge and the SecureTransport Server. Otherwise, define these rules on the firewall between the SecureTransport Edge and the Internet.

For internal users to upload or download files to SecureTransport, they must log into the SecureTransport Server directly using HTTP(S) or FTP(S). So those ports from the secure network to the SecureTransport Server must be open. In some installations, access is only through a proxy. In this case secure network requires access to the Proxy server.

Both FTP and FTPS use port 20 for the data channel in active mode. If preferred, you can define a passive-port range instead, and set up a similar rule for that range of ports. Both FTP and FTPS use port 21 for the control channel. For more details, see *Passive port range is not defined in the firewall on page 1057*.

HTTP access is optional. To disable HTTP, do not define rule 4.

If the SMTP server that handles ad hoc file transfer email notifications from ST Web Client is in the secure network, do not define rule 15.

Authentication rules

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
17	ST Server	Trusted (secure network)	SiteMinder	44441	➔ (outbound)	SiteMinder Accounting
18	ST Server	Trusted (secure network)	SiteMinder	44442	➔ (outbound)	SiteMinder Authentication
19	ST Server	Trusted (secure network)	SiteMinder	44443	➔ (outbound)	SiteMinder Authorization

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
20	ST Server	Trusted (secure network)	LDAP	389 or 3268	→ (outbound)	LDAP user lookup and authentication

Administration rules

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
21	Trusted (secure network)	ST Server	HTTPS	444	← (inbound)	SecureTransport Administration Tool

TM server communication rules

Network zones define the server ports that the TM Servers running on SecureTransport Servers in the secure network connect to on the SecureTransport Edge servers running in the peripheral network (DMZ). See *Communication across Transaction Manager, protocol, and proxy servers on page 227*.

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
22	ST Servers	ST Edge (FTP Server)	TCP over SSL	20021	→ (outbound)	Transaction Manager streaming protocol
23	ST Servers	ST Edge (HTTP Server)	TCP over SSL	20080	→ (outbound)	Transaction Manager streaming protocol
24	ST Servers	ST Edge (AS2 Server)	TCP over SSL	21080	→ (outbound)	Transaction Manager streaming protocol

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
25	ST Servers	ST Edge (SSH Server)	TCP over SSL	20022	➔ (outbound)	Transaction Manager streaming protocol
26	ST Servers	ST Edge (PeSIT Server)	TCP over SSL	27617	➔ (outbound)	Transaction Manager streaming protocol
27	ST Servers	Administration Tool server	TCP over SSL	20444	➔ (outbound)	Transaction Manager streaming protocol

Do not define these rules in a deployment with no SecureTransport Edge. Define only the rules for the protocols you are using.

Note that in SecureTransport Edge deployment, port 20444 (used by the Administration Tool server) must always be open.

Server transfer rules

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
28	ST Servers	ST Edge	SOCKS	1080	➔ (outbound)	SOCKS5 proxy for server-initiated transfers (out-bound)

Do not define rule 28 for a deployment with no SecureTransport Edge.

Standard Cluster rules

The following rules are required for a Standard Cluster (SC).

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
29	ST Servers	ST Servers	Proprietary	44431	←→ (server-to-server)	Server synchronization and heartbeat
30	ST Servers	ST Servers	HTTPS	444	←→ (server-to-server)	Synchronization

Enterprise Cluster rules

The following rules are required for an Enterprise Cluster (EC). The cluster cache manager uses the first free port starting at 8088.

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
31	ST Servers	ST Servers	TCP and UDP	8088 through 8093	←→ (server-to-server)	Cluster cache management
32	ST Servers	ST Servers	TCP and UDP	7574	←→ (server-to-server)	Coherence cluster multicast port
33	ST Servers	ST Servers	TCP	7	←→ (server-to-server)	Coherence TcpRing/IpMonitor death detection

Protocol rules - outbound from SecureTransport Edge

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
34	ST Edge	Internet	FTP(S)	1024 to 65535 port range specified on the Edge	← (inbound)	Server FTP (S) data channel, active mode

#	Group source	Group destination	Protocol	Destination port	Direction	Purpose
35	ST Edge	Internet	FTP(S)	1024 to 65535 port range specified on the client server	➔ (outbound)	Server FTP (S) data channel, passive mode
36	ST Edge	Internet	FTP(S)	21	➔ (outbound)	Server FTP (S) data channel
37	ST Edge	Internet	HTTP	80	➔ (outbound)	Server HTTP data channel
38	ST Edge	Internet	HTTPS	443	➔ (outbound)	Server HTTPS data channel
39	ST Edge	Internet	SSH	22	➔ (outbound)	Server SSH data channel
40	ST Edge	Internet	AS2	Specified in the transfer site	➔ (outbound)	Server AS2 data channel
41	ST Edge	Internet	PeSIT	Specified in the transfer site	➔ (outbound)	Server PeSIT data channel

If you disable the default HTTP port by not defining rule 4, do not define rule 37.

Define rules 40 and 41 for each port specified for the partnerships in the transfer sites.

Appendix G: IP addresses and host names

This appendix describes valid syntax for IP addresses and host names you can use with Axway SecureTransport. In fields in the Administration Tool, you enter an IP address or host name to represent a host or server, a pattern to represent a range of IP address, or a pattern that SecureTransport matches against input data.

The following topic lists the IP address and host name syntax formats:

- *IP address and host name syntax* on page 1101 - Lists the IP address and host name syntax formats.

IP address and host name syntax

In SecureTransport, you can specify a host address in any of the following formats:

- Exact IPv4 or IPv6 address
- Range of address using Classless Inter-Domain Routing (CIDR) notation
- Range of address using IPv4 address and netmask
- Patten matching an IPv4 address
- Exact host name
- Pattern matching a host name

The following topics IP address and host name syntax usage:

- *Exact IPv4 or IPv6 address* on page 1102 - Describes the use of an exact IPv4 or IPv6 address to specify a single server.
- *Range of address using Classless Inter-Domain Routing notation* on page 1102 - Describes the use of a range of addresses using classless inter-domain routing (CIDR) notation.
- *Range of address using IPv4 address and subnet mask* on page 1102 - Describes the use of an IPv4 address and a subnet mask separated by a colon (`:`) to represent a range of IPv4 addresses.
- *Patten matching an IPv4 address* on page 1102 - Describes the use of an IPv4 address pattern to represent a range of IPv4 addresses.
- *Exact host name* on page 1103 - Describes the use of a literal host name to represent a single host where host names are valid.
- *Pattern matching a host name* on page 1103 - Describes the use of a host name pattern that uses asterisk (`*`) to represent one or more characters and question mark (`?`) to represent one character.

Exact IPv4 or IPv6 address

Use an exact IPv4 or IPv6 address to specify a single server. In IPv6 addresses, two colons (`::`) can represent one sequence of zero bits.

Examples of valid values include the following:

- `172.23.34.45`
- `127.0.0.1`
- `FC00:1234:56:0:0:0:AB:EF`
- `FC00:1234:56::AB:EF`
- `::1`

Range of address using Classless Inter-Domain Routing notation

Classless Inter-Domain Routing (CIDR) notation specifies an IPv4 or IPv6 address and a number of significant bits separated by a slash (`/`). In other contexts, CIDR notation is used to specify an IP address and a subnet. For SecureTransport, use CIDR notation to represent a range of IP addresses.

Examples of valid values include the following:

- `172.23.34.0/24` represents `172.23.34.0` through `172.23.34.255`
- `FC00:1234:56::/120` represents `FC00:1234:56::` through `FC00:1234:56::FF`

Range of address using IPv4 address and subnet mask

Use an IPv4 address and a subnet mask separated by a colon (`:`) to represent a range of IPv4 addresses.

Examples of valid values include the following:

- `172.23.34.0:255.255.255.0` represents `172.23.34.0` through `172.23.34.255`
- `172.56.67.128:255.255.255.224` represents `172.56.67.128` through `172.56.67.159`

Patten matching an IPv4 address

Use an IPv4 address pattern to represent a range of IPv4 addresses. Use an asterisk (`*`) to match any sequence of octal digits.

Examples of valid values include the following:

- `172.23.34.*` represents `172.23.34.0` through `172.23.34.255`
- `172.56.*.*` represents `172.56.0.0` through `172.56.255.255`

Exact host name

Use a literal host name to represent a single host where host names are valid. The host name must resolve to a valid IPv4 or IPv6 address.

Pattern matching a host name

Use a host name pattern that uses asterisk (`*`) to represent one or more characters and question mark (`?`) to represent one character. The pattern specifies any host whose name matches. A host name pattern is valid for values that SecureTransport uses to match a host name, for example, in a user class definition.

Examples of valid values include the following:

- `*.example.com`
- `mail?.example.edu`
- `int?*`

Appendix H: Expression Language

This appendix provides information about the syntax for the expressions used for post-transmission actions, PGP, and account templates supported by Axway SecureTransport.

The following topics list and provide Expression Language variables and functions:

Expression Language overview

Many features of SecureTransport use expressions. The expression language SecureTransport uses is based on the Sun JSP Expression Language. Only the syntax listed in this appendix is supported by SecureTransport, any other syntax is not guaranteed to function properly.

The following SecureTransport features can use the expression language:

- Transfer site post-transmission actions
- Subscription post-transmission actions
- PGP
- Account templates

Note If an account template and its transfer site are defined using expressions, you cannot restart failed transfers for that account template using the **Resubmit** button on the *File Tracking* page.

This appendix provides a list of the supported syntax with examples.

Note When creating expressions that use file paths you must use the forward slash (/) regardless of the platform where the file path is located. For example, instead of writing `c:\tmp\${stenv['loginname']}`, write `c:/tmp/${stenv['loginname']}`.

Expression Language operators

The following operators are supported:

Operator	Description
`[ ]` and `.`	Used to refer to attributes of an object or items in collection.
`+`, `-`, `*`, `/` or `div`, `%` or `mod`	Arithmetic Operators. Both binary and unary operators are supported
`==` or `eq`, `!=` or `ne`, `<` or `lt`, `>` or `gt`, `<=` or `le`, `>=` or `ge`	Relational Operators. provides ability to compare values
`&&` or `and`, `||` or `or`, `!` or `not`	Logical Operators
empty	Empty Operator
`? :`	Conditional Operator

Parentheses can be used in combination with the operators to change precedence.

SecureTransport evaluates operators using the following precedence order, listed from highest to lowest and left to right:

- `[ ] .`
- `( )`
- `-` (unary) `not ! empty`
- `* / div % mod`
- `+ -` (binary)
- `< > <= >= lt gt le ge`
- `== != eq ne`
- `&& and`
- `|| or`
- `? :`

Predefined variables

SecureTransport supports the following predefined variables:

Name	Syntax	Description
Time Stamp	`${timestamp}`	Returns the UNIX epoch time in milliseconds as a decimal integer.
PGP Embedded File Name	`${embedded}`	The original file name embedded in the PGP encoded file. This value is defined after a PGP encoded file is decrypted. If the file was not PGP encoded, returns an empty string.

Predefined variable examples

The following table shows examples of the predefined variables:

Name	Example usage	Example return value
Time Stamp	`${timestamp}`	`1345652729052`
PGP Embedded File Name	`${embedded}`	`original-file-name.txt`

Predefined functions

SecureTransport supports the following predefined functions:

Name Syntax	Description
Date `${date()}` or `${date(`*`date-and-time-pattern`*`)}`	Returns the current date and time. The *date-and-time pattern* is the same format defined in the Java class `java.text.SimpleDateFormat`. If no format is specified then the output is the default date format in the current locale. Be aware that weeks are numbered differently depending on the locale and calendar in use. More details in Troubleshooting section.
dayOffset `${dayOffset ('yymmdd', 'var1')}`	Returns the current date and time with `var1` days added or subtracted as an offset.

Name Syntax	Description
Random ID`${random()}`	Creates a pseudo-random string using both letters and numbers. Format is a 32 byte hex string.
String Concatenation`${concat (var1, var2)}`	Creates a new string concatenating the two variables together.
Substring `${substring (variable, beginIndex,endIndex)}`	Returns a substring for a given string. The substring begins at the specified `beginIndex` and extends to the character at index `endIndex` - 1. Thus, the length of the substring is `endIndex-beginIndex`.
Force Exception`${error()}` or `${error(message)}`	Throws an exception error. If `${error(message)}` is used, a message is returned with the error. This message is logged along with the exception.
File Name`${filename (variable)}`	Returns the target file name with the extension, but without the path to the file.
File Basename`${basename (variable)}`	Returns the target file name without the extension or path to the file.
File Extension`${extension (variable)}`	Returns the target file extension, including the dot. If there is no extension, an empty string is returned.
`${resolve()}`	Removes any occurrence of ., or . in a file path. This function works with POSIX style paths only. If the resolved path is null, empty or returns a `/`, an error is returned.

Predefined function examples

The following table shows examples of the predefined functions:

Name	Example usage	Example return value
Date (default)	`${date()}`	`June 22, 2012 1:42:04 AM`
Date (formatted)	`${date('EEEE, M-d H:m')}`	`Friday, 6-22 1:42`

Name	Example usage	Example return value
DayOffset	`${dayOffset('yymmdd', '-1')}`	`${dayOffset('yyMMdd', '-5')}` - returns 10th if today is 15th of August formatted as per the specified format parameter - `120810`. `${dayOffset('yyMMdd', '+7')}` - returns 22th if today is 15th of August formatted as per the specified format parameter - 120822. `${dayOffset('ddMMyy', '+1') ge '090414'}`
Random ID	`${random()}`	`C7F2119AAECEACCDE16C496C96FEEE39`
String Concatenation	`${concat('str', 'ing')}`	`string`
Substring	`/${substring (stenv.loginname,0,1)}` where `stenv.loginname` is interoperable with `env ['DXAGENT_LOGINNAME']`	`string`
Force Exception	`${error()}`	`com.tumbleweed.util. expressions. InvalidExpressionException Caused by: java.lang.Exception: Unspecified error`
Force Exception (with a specific error message)	`${error(*message*)}`	`com.tumbleweed.util. expressions. InvalidExpressionException Caused by: java.lang.Exception: An error has occurred in this part of the process!`
Full File Name	`${filename($file)}`	`filename.txt`
File Basename	`${basename($file)}`	`filename`

Name	Example usage	Example return value
File Extension	`${extension($file)}`	`.txt`
Path Resolution	`${resolve('.././path')}`	path
Path Resolution with Error	`${resolve('../../..')}`	`com.tumbleweed.util.expressions.InvalidExpansionException: Invalid path resolution: /`
Path Resolution with Error	`${resolve('\\')}`	`com.tumbleweed.util.expressions.InvalidExpansionException: Path must not contain '\' character`

SecureTransport specific named variable sets

Named variable sets separate variables into logical groups. Named variable sets use the following syntax:

`${name['variable']}`

or

`${name.variable}`

SecureTransport uses the following named variable sets:

- `${sess['variable']}` – used with SecureTransport session variables including LDAP
- `${env['variable']}`, `${stenv['variable']}`, or `${stenv.variable}` – used with SecureTransport predefined environment variables
- `${pesit['variable']}` – used with SecureTransport PeSIT variables described in *PeSIT expressions* on page 1110

LDAP session variables can be used with the `sess` named variable set. You can also develop an agent that contains the session variables you want to use. All session variables must be prefixed with `STSESSION_`.

The `env` named variable set contains the entire environment, including any non- SecureTransport-specific variables. Environment variables accessed using `stenv` are preprocessed to remove the `DXAGENT_` prefix, and upper case characters are converted to lower case characters. For example, to use the environment variable `DXAGENT_TARGET`, write the following expression:

`${env['DXAGENT_TARGET']}`

or use the `stenv` named variable set and access the variable as:

```
${stenv['target']} or ${stenv.target}
```

SecureTransport-specific named variable set examples

The following table shows examples of SecureTransport-specific named variables:

Example	Example return value
`${sess['STSESSION_LDAP_DIR_homeDirectory']}`	`/home/user1`
`${sess['STSESSION_LDAP_DN']}`	`cn=john,ou=People,dc=tp,dc=axway,dc=com`
`${sess['STSESSION_LDAP_DIR_uidNumber']}`	`1000`
`${env['DXAGENT_HOMEDIR']}`	`/home/user2`
`${stenv['homedir']}`	`/home/user2`
`${env['DXAGENT_FULLSOURCE']}`	`/st/monitor/download/test.xml`
`${stenv['fullsource']}`	`/st/monitor/download/test.xml`
`${stenv.rawsource}`	`AS2OriginalFile`
`${stenv.site_target}`	`OriginalFile`

PeSIT expressions

The following expressions are valid in transfer profiles:

Expression	PeSIT PI code	Applicability to Push / Pull operations
${pesit.crc}	PI 1	Pull only
${pesit.diagCode}	PI 2	
${pesit.callerID}	PI 3	Pull only
${pesit.senderID}	PI 3	Both Push and Pull
${pesit.serverID}	PI 4	Pull only
${pesit.receiverID}	PI 4	Both Push and Pull
${pesit.version}	PI 6	Pull only
${pesit.checkPointInterval}	PI 7	Both Push and Pull
${pesit.checkPointWindow}	PI 7	Both Push and Pull
${pesit.fileType}	PI 11	Both Push and Pull
${pesit.fileName}	PI 12	Both Push and Pull
${pesit.transferID}	PI 13	Both Push and Pull
${pesit.fileAttributes}	PI 14	
${pesit.restart}	PI 15	Pull only
${pesit.dataEncoding}	PI 16	Both Push and Pull
${pesit.priority}	PI 17	Both Push and Pull
${pesit.restartCheckPoint}	PI 18	Pull only

Expression	PeSIT PI code	Applicability to Push / Pull operations
${pesit.cancelCode}	PI 19	
${pesit.checkPointNumber}	PI 20	
${pesit.compressed}	PI 21	Pull only
${pesit.compressionType}	PI 21	Both Push and Pull
${pesit.accessType}	PI 22	Pull only
${pesit.resyncAllowed}	PI 23	Pull only
${pesit.exchangeBufferSize}	PI 25	Both Push and Pull
${pesit.totalBytes}	PI 27	
${pesit.totalRecords}	PI 28	
${pesit.diagnosticText}	PI 29	
${pesit.recordFormat}	PI 31	Both Push and Pull
${pesit.recordLength}	PI 32	Both Push and Pull
${pesit.fileOrganization}	PI 33	Both Push and Pull
${pesit.fileLabel}	PI 37	Both Push and Pull
${pesit.keyLength}	PI 38	
${pesit.keyOffset}	PI 39	
${pesit.allocationUnit}	PI 41	Push only
${pesit.allocationSize}	PI 42	Push only

Expression	PeSIT PI code	Applicability to Push / Pull operations
`${pesit.creationDateTime}`	PI 51	Push only
`${pesit.extractionDateTime}`	PI 52	
`${extractionDateTimepesit.originalSenderID}`	PI 61	
`${pesit.finalDestinationID}`	PI 62	Push only
`${pesit.msgData}`	PI 91	
`${pesit.serviceParam}`	PI 99	
`${pesit.accountName}`	—	
`${pesit.datetime}`	—	
`${pesit.details}`	—	

Note The expressions `${pesit.originalSenderID}` and are `${pesit.finalDestinationID}` are set only for routed transfers.

Advanced Routing EL functions and variables

For a complete listing of the Advanced Routing Expression Language (EL) functions and variables, refer to *Custom Expression Language functions and variables* on page 1003.

Match and replace functions

The expression language match and replace functions can match a regular expression or replace it.

The syntax for the replace operation is:

```
${variable.replace(<match RE>, <replace RE>)}
```

If the match succeeds, the value is the string with the matched string replaced.

The syntax for a match operation is:

```
${variable.matches(<match RE>)}
```

If the match succeeds, the value is `true`. If it does not, the value is `false`.

Note The match operation returns a logical value that can be used with relational or conditional operators.

For more about regular expressions, see *Regular expressions* on page 1117.

Regular expression examples

The following table shows several examples of using the match and replace operations with regular expressions.

Name	Example	Example return value
Match	`${foo.matches('fo*')}`	`true`
Replace	`${foo.replace('f(.*)', 'm$1')}`	`moo`

Expression examples

This topic provides additional examples on expression usage. In this topic, sample variable values are given to show how various expressions can use the information. The examples provided in this topic apply to:

- Transfer site post-transmission actions
- Subscription post-transmission actions
- PGP
- Account templates

For a complete listing of the Advanced Routing Expression Language (EL) functions and variables, refer to *Custom Expression Language functions and variables* on page 1003.

Expression variables and examples

The following table provides the variable names and values that are used in the subsequent examples:

Variable name	Value
`DXAGENT_LOGINNAME`	`stuser`
`DXAGENT_HOMEDIR`	`/home/users/stuser`
`DXAGENT_TARGET`	`document_12.txt`

Variable name	Value
DXAGENT_TARGETPATH	/home/users/stuser
DXAGENT_TRANSFORMATION_INPUT	/home/users/stuser/PGP/encrypted.pgp

Note The variable DXAGENT_TRANSFORMATION_INPUT is only available from within a PGP transformation agent. Expressions using this variable do not evaluate correctly in other cases. In the example using this variable, assume that the original file name before encryption is original_12.txt.

The following table shows additional examples of the different expressions available for use in SecureTransport:

Example variable name	Example return value
${env['DXAGENT_TARGET']}	document_12.txt
${stenv['target']}	document_12.txt
${stenv["target"]}	document_12.txt
Prefix${stenv ["target"]}.newext	Prefixdocument_12.txt.newext
${stenv.targetpath}	/home/users/stuser
${basename(stenv.target)}	document_12
${extension(stenv.target)}	txt
${filename (stenv.target).replace(' [0-9]','z').replace(' [*&_]','-')}	document-zz.txt
${basename(stenv.target)}-${random}.${extension (stenv.target)}	document_12-C7F2119AAECEACCDE16C496C96FEEE39.txt
${stenv.loginname.replace ('st', 'SecureTransport-')}	SecureTransport-user

Example variable name	Example return value
`${stenv.target.matches ('.*\\.txt') ? 1 : 0}`	`1`
`${stenv.target.matches ('.*\\.pgp') ? 1 : 0}`	`0`
`${stenv.transformation_ input.matches('\.*\\.pgp') ? 1 : 0}`	`1`
`${embedded}`	`original_12.txt`
`${basename(embedded)}- ${timestamp}.${extension (embedded)}`	`original_12-1345652729052.txt`
`${basename(embedded)}- ${date('yyyy.MM.dd_ hhmmss')}.${extension (embedded)}`	`original_12-2012.08.22_162529.txt`
`${resolve(concat (stenv.homedir, '../..'))}`	`/home`
`${empty var ? error ('Missing Variable') : var}`	Either the value of `$var` or an exception

Appendix I: Regular expressions

This appendix provides information about the syntax for regular expressions supported by Axway SecureTransport.

The following topics describe the components of SecureTransport regular expressions:

Regular expression characters

The following table shows the variables and characters allowed within the regular expressions.

Character	Meaning
unicodeChar	Matches any identical Unicode character
\	Used to quote a meta-character (such as *)
\\	Matches a single '\' character
\0*nnn*	Matches a given octal character
\x*hh*	Matches a given 8-bit hexadecimal character
\\u*hhhh*	Matches a given 16-bit hexadecimal character
\t	Matches an ASCII tab character
\n	Matches an ASCII newline character
\r	Matches an ASCII return character
\f	Matches an ASCII form feed character

General character classes

You can specify a character class to match a set of characters.

Character class	Meaning
`[abc]`	Matches any character between the brackets
`[a-zA-Z]`	Use hyphen (-) to specify ranges of characters
`[^abc]`	Matches any character not specified

Predefined character classes

There is a set of predefined character classes. The following table explains these.

Character class	Meaning
`.`	Matches any character other than newline
`\w`	Matches a "word" character, alphanumeric plus underscore (_)
`\W`	Matches a non-word character
`\s`	Matches a whitespace character
`\S`	Matches a non-whitespace character
`\d`	Matches a digit character
`\D`	Matches a non-digit character
`[:alnum:]`	Alphanumeric characters *
`[:alpha:]`	Alphabetic characters *
`[:blank:]`	Space and tab characters *
`[:cntrl:]`	Control characters *
`[:digit:]`	Numeric characters *

Character class	Meaning
`[:graph:]`	Characters that are printable and are also visible. (A space is printable, but not visible, while an "`a`" is both.) *
`[:javastart:]`	Start of a Java identifier *
`[:javapart:]`	Part of a Java identifier *
`[:lower:]`	Lower-case alphabetic characters *
`[:print:]`	Printable characters (characters that are not control characters) *
`[:punct:]`	Punctuation characters (characters that are not letter, digits, control characters, or space characters) *
`[:space:]`	Space characters (for example, space, tab, and form feed) *
`[:upper:]`	Upper-case alphabetic characters *
`[:xdigit:]`	Characters that are hexadecimal digits *
`\p{Alnum}`	Alphanumeric characters †
`\p{Alpha}`	Alphabetic characters †
`\p{Blank}`	Space and tab characters †
`\p{Cntrl}`	Control characters †
`\p{Digit}`	Numeric characters †
`\p{Graph}`	Characters that are printable and are also visible. (A space is printable, but not visible, while an "`a`" is both.) †
`\p{Lower}`	Lower-case alphabetic characters †
`\p{Print}`	Printable characters (characters that are not control characters) †
`\p{Punct}`	Punctuation characters (characters that are not letter, digits, control characters, or space characters) †
`\p{Space}`	Space characters (for example, space, tab, and form feed) †
`\p{Upper}`	Upper-case alphabetic characters †
`\p{Xdigit}`	Characters that are hexadecimal digits †

* For regular expressions used with the Transaction Manager rule ~ `(match)` operator see LDAP domain DN filter (see *Manage DN filters for a domain* on page 490).

† For all other regular expressions.

Boundary matches

The following table defines boundary matches.

Character class	Meaning
`\^`	Matches only at the beginning of a line
`\$`	Matches only at the end of a line
`\b`	Matches only at a word boundary
`\B`	Matches only at a non-word boundary

Note Transaction Manager automatically places a "`\^`" at the beginning of the search string and a "`\$`" at the end of the search string. If you do not want to anchor the regular expression, you must add "`.*`" to the beginning or end:
`.*regex.*`

Regular expression closures

These match a series of characters by matching the pattern they follow zero or more time.

Character class	Meaning
P`*`	Matches the pattern *P* zero or more times
P`+`	Matches the pattern *P* one or more times
P`?`	Matches the pattern *P* zero or one times
P`{`*n*`}`	Matches the pattern *P* exactly *n* times
P`{`*n*`,}`	Matches the pattern *P* at least *n* times
P`{`*n*`,`*m*`}`	Matches the pattern *P* at least *n* but not more than *m* times.

All closure operators (`*`, `+`, `?`, `{`*n*`,`*m*`}`) are *greedy* by default, meaning that they match as many characters of the string as possible without causing the overall match to fail.

A *reluctant* closure matches as few characters of the string as possible without causing the overall match to fail. To specify a reluctant closure, append a `?` to the closure pattern. Valid patterns are *P*`*?`, *P*`+?`, and *P*`??`.

Logical and grouping operators

Use these items to match a sequence of characters and indicate subexpressions.

Operator	Meaning
AB	Matches pattern *A* followed by pattern *B*
A`|`*B*	Matches either pattern *A* or pattern *B*
`(`*A*`)`	Groups pattern *A* as a subexpression for other operations or for back references

Back references

Use the following character classes to refer back to subexpressions in patterns and in replacement strings.

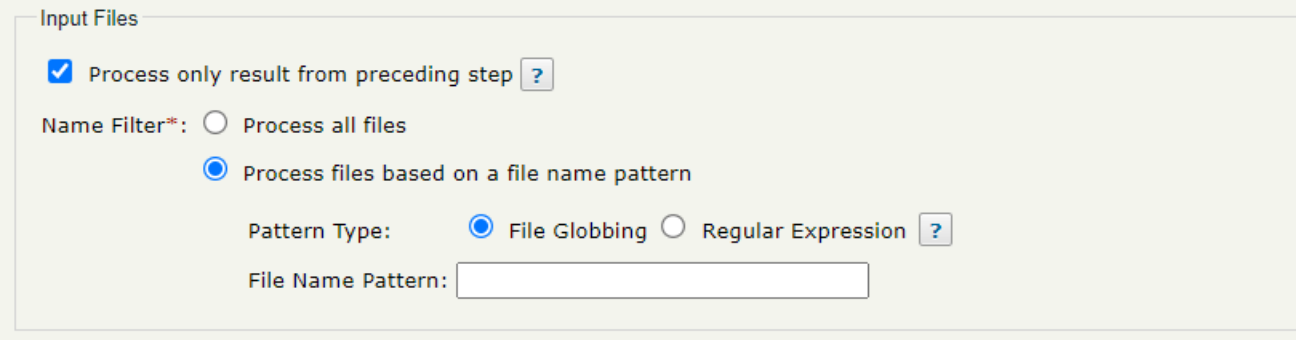
Character class	Meaning
`\1`	Back reference to 1st parenthesized subexpression
`\2`	Back reference to 2nd parenthesized subexpression
`\3`	Back reference to 3rd parenthesized subexpression
`\4`	Back reference to 4th parenthesized subexpression
`\5`	Back reference to 5th parenthesized subexpression
`\6`	Back reference to 6th parenthesized subexpression
`\7`	Back reference to 7th parenthesized subexpression
`\8`	Back reference to 8th parenthesized subexpression
`\9`	Back reference to 9th parenthesized subexpression

Globbing in SecureTransport

A glob is a pattern for matching names based on wildcards. In SecureTransport, glob patterns can be used for matching names of files, directories, transfer sites and even profiles.

Fields that support globs

You can use wildcard characters to specify which files to be processed by an Advanced Routing step; the **File Name Pattern** field is available in all Advanced Routing steps from **Input Files > Process files based on a file name pattern > File Globbing**.



In the AR Send to Partner step, globs are also accepted in the **Account Transfer Sites** and **Transfer Profile** fields.

In the Folder Monitor and SSH transfer sites, you can use wildcard characters in the **Download Pattern** filed to identify the files to be downloaded. Globs are also supported in the **Subfolder Name Pattern** field to determine the subfolders to be monitored.

Note Where SecureTransport accepts glob patterns, it will usually accept also regular expressions.

Glob Syntax

Even though globs resemble regex, they are not regular expressions. Glob patterns follow standard Unix path expansion rules, where you write patterns that represent sets of names using special characters called wildcards.

Here are the most commonly used wildcards:

- An asterisk * matches any number of characters

  Example: `*.xml` matches all files with extension *xml*
- Braces {...} are used to define a list of patterns. The braces are interpreted before any other wildcards.

`*.{zip,txt}` matches any file with extension *xml* or *txt*.

- A question mark ? matches exactly one character.

  Example: `foo.??` matches file names starting with `foo.` that have a double character extension.

- Square brackets [...] match any character within the group, i.e., [abc] means "any character from a, b, or c"

  Example: `*.[0-9]` matches file names having a single character extension that is a numeric value.

- An exclamation mark (!) is used for exclusion.

  Example: `*.[!0-9]` matches file names having a single character extension different from a numeric value.

Examples of complex globs:

- `[a-c]test*` – matches a file name which include any of the letters "a", "b" or "c" , followed by the string "test". The asterisk * character means that any character(s) can follow the word "test".
- `*a*` – matches file names which include any sequence of characters, which include the letter "a".
- `[0-9]*${date("YYYY")}.txt` – matches file names that start with a numeric value, followed by any character, which is followed by the current year and have extension *txt*.

Appendix J: Velocity email notification package

Email notifications triggered by the `ServerTransferNotify` rule package, as well as some others, such as password reset and ICAP scan related notifications, are commonly referred to as *Velocity email notifications*.

The following topics describe how to set up and configure the Velocity email notification package:

- *Velocity overview on page 1124* - Provides a Velocity email notification package overview.
- *Customize the email notification templates on page 1125* - Provides how-to instructions for customizing the email notification templates.
- *Velocity troubleshooting on page 1127* - Provides how-to instructions for troubleshooting the Velocity email notification package.

Velocity overview

Velocity is an open source template engine provided by the Jakarta Apache Project. It uses the Velocity Template Language (VTL) which provides an easy and simple way to incorporate dynamic content in an HTML email.

VTL uses *references* to embed dynamic content into an HTML email and a variable is one type of reference.

The following HTML code snippet shows a VTL statement that can be embedded in an HTML email notification generated by SecureTransport:

```
<tr id="areabody" bgcolor="white">
   <td colspan="2">
      You have received a new File <b>$DXAGENT_TARGET</b> from the
      Remote Host $DXAGENT_REMOTEHOST
   </td>
</tr>
```

For more information, refer to the Apache Velocity Project website.

Customize the email notification templates

SMTP settings can be configured through the Administration Tool as well as through email notification templates. It is recommended that you configure the SMTP transport security settings globally by following the *Set up email notifications via SMTP* on page 202 procedure. If needed, these settings can then be overridden on a per-template basis by using the custom fields described in this topic.

The templates are located in `<FILEDRIVEHOME>/conf/mailer-templates`. You can customize the email notification template with any text or HTML editor. An HTML editor is recommended because you can use it to preview the notification before deploying it in SecureTransport.

Any of the following fields can be set in the XHTML email notification template:

- `$subject` — The subject line for the email.
- `$mailfrom` — Who the email is coming from.
- `$mailto` — Who the email is going to.
- `$mailserver` — The name of the local SMTP mail server that will be used to deliver the email.
- `$mailserverport` — The TCP port on the SMTP server to connect to.
- `$smtpUser` — The user that is used when authenticating to an SASL-enabled SMTP server.
- `$smtpPassword` — The authentication password for the user specified in the `smtpUser` field.
- `$smtpTLS` — Specifies the connection security type required by the SMTP server.
- `$smtpVerifyServerCertificate` — Specifies whether the server certificate should be verified.
- `$smtpVerifyServerIdentity` — Specifies whether the server identity should be verified.
- `$smtpProtocols` — Specifies the enabled SMTP protocols.
- `$smtpCipherSuites` — Specifies the enabled cipher suites for SSL connections.

The following code example shows an XHTML file with some customized fields:

```
<?xml version="1.0" encoding="windows-1252"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
   "http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<!-- #set( $subject = "ST was unable to deliver a file to
   $DXAGENT_SITE_ATTR_HOST") -->
<!-- #set( $mailfrom = "admin@example.com") -->
<!-- #set( $mailto = $DXAGENT_ACCOUNT_EMAIL) -->
<!-- #set( $mailserver = "mail.example.com") -->
<!-- #set( $mailserverport = "25") -->
<!-- #set( $smtpUser = "user") -->
<!-- #set( $smtpPassword = "password") -->
```

```
<!-- #set( $smtpTLS = "StartTLS") -->
<!-- #set( $smtpVerifyServerCertificate = "false") -->
<!-- #set( $smtpVerifyServerIdentity = "false") -->
<!-- #set( $smtpProtocols = "TLSv1.2, TLSv1.3") -->
<!-- #set( $smtpCipherSuites = "TLS_AES_256_GCM_SHA384,TLS_AES_
128_GCM_SHA256") -->
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
   <meta http-equiv="Content-Type" content="text/html;
      charset=windows-1252" />
</head>
<body>
   <table border="0" bgcolor="#C0C0C0" width="100%">
      <tr id="areatop">
         <td style="text-align: left">Axway</td>
         <td style="text-align: right">SecureTransport</td>
      </tr>
      <tr id="areatopmain">
         <td colspan="2" align="center">
            File Delivery Failure
         </td>
      </tr>
      <tr id="areabody" bgcolor="white">
         <td colspan="2">
            Delivery of the file <b>$DXAGENT_TARGET</b> to the
host
            <b>$DXAGENT_SITE_ATTR_HOST</b> failed after
            <b>$DXAGENT_PERSISTED_EVENT_RETRY_COUNT</b>
attempts.
            <p>Inspect the File Tracking Page in the ST
Administration Tool
            for more information on the reason for the
failure.</p>
         </td>
      </tr>
   </table>
</body>
</html>
```

Note You can hard code the values directly into the email notification template, or you can use any SecureTransport environment variable. In the previous example, the `mailfrom` address is hard coded into the template, but the `mailto` address is calculated at runtime from the `$DXAGENT_ACCOUNT_EMAIL` variable.
Any value set in the Invocation Parameter for the TM rule overrides the value set in the notification template.

Velocity troubleshooting

To add additional logging of email notifications events, edit the `tm-log4j.xml` file in the `<FILEDRIVEHOME>/conf/` directory and add the following code in the `loggers` section of the file:

```
<Logger name="com.tumbleweed.st.server.util.mailer"
level="DEBUG" additivity="false">
   <AppenderRef ref="ServerLog"/>
   <!--<AppenderRef ref="Stdout" />-->
   <!--<AppenderRef ref="Stderr" />-->
</Logger>
<Logger name="javax.mail" level="DEBUG" additivity="false">
   <AppenderRef ref="ServerLog"/>
   <!--<AppenderRef ref="Stdout" />-->
   <!--<AppenderRef ref="Stderr" />-->
</Logger>
       <Logger name="org.apache.commons.mail" level="DEBUG"
additivity="false">
   <AppenderRef ref="ServerLog"/>
   <!--<AppenderRef ref="Stdout" />-->
   <!--<AppenderRef ref="Stderr" />-->
</Logger>
```

SecureTransport Glossary J

This glossary highlights some of the key terms that will help you get familiar with and understand SecureTransport better.

The terms are separated into the following sections:

Server management

Term	Definition
Administration Tool	The Administration Tool is a web service that hosts the SecureTransport Administration UI and serves as an endpoint for the API service.
Folder Monitor	The Folder Monitor is a service available only on SecureTransport Servers that allows scanning designated folders for specific files. See *Folder Monitor transfer sites* on page 557.
Protocol daemon	A protocol daemon is a service that includes one or more servers of the same protocol. Supported protocols in SecureTransport (Server and Edge) for client-initiated transfers are HTTP, FTP, SSH, PeSIT, and AS2.
Protocol server	A protocol server (also called listener) is a combination of a protocol, port(s), and various connection and protocol settings. There is one default server per daemon. See Protocol servers.
Scheduler	The Scheduler is a service available only on SecureTransport Servers that allows executing server-initiated actions based on a schedule. See *Server Control: Scheduler* on page 261.

Term	Definition
Transaction Manager (TM)	The Transaction Manager is an event-based rule engine processor that is available only on SecureTransport Servers and handles all business operations - transfers, authentication, notifications, etc. See *Transaction Manager Settings* on page 217.

Account management

Term	Definition
Account	A user account represents a physical user or another application that must be able to log in to SecureTransport. It is a key component for file flows through transfer sites, transfer profiles, routes, and subscriptions. See *User accounts* on page 501.
Account Template	An account template allows SecureTransport to use credentials from external user repositories such as LDAP, SSO, and Active Directory without creating local accounts. See *Account templates* on page 717.
Account Type	An account type is used to group SecureTransport accounts into two categories: internal users and partner accounts. Internal accounts are used for transfers within a single organization. Partner accounts are used for transfers between organizations and partners.
GID	A GID is a numeric group ID used within each account to determine access rights and privileges valid for this user group on the SecureTransport system. It is not required for the GID to exist on the platform.
Login Name	A login name is the unique name SecureTransport uses to identify the account upon login.
UID	A UID is a numeric user ID used within each account on UNIX and Linux platforms. It is not required for the UID to exist on the platform.
Unlicensed User	An unlicensed user does not require a full-scale account license to connect to ST Web Client to retrieve files. Unlicensed users can reply only once to the sender of a file transfer email. See *Unlicensed users* on page 520.

Organization management

Term	Definition
Administrator	An administrator can access the SecureTransport Server, Edge, and API to perform administrative tasks based on their role and permissions. See *Manage administrator accounts* on page 700.
Administrator Role	Administrator accounts are assigned an administrative role that defines account privileges and permissions. See *Administrative roles* on page 711.
Business Unit	A business unit allows grouping multiple user accounts under a common set of rules and configurations. See *Business units* on page 746.
Delegated Administrator	A delegated administrator manages specific user groups referred to as business units. Tracking information is also displayed based on the assigned business unit. See *Delegated administration* on page 707.

Authentication

Term	Definition
Download Restriction	A download restriction is used to grant or deny users the permission to download files based on their home folder path.
Filesystem Restriction	A filesystem restriction is used to control permissions for specific user classes to modify files and directories within a user's home folder path.
Internal CA	During the installation of SecureTransport, an internal Certificate Authority is created automatically and can be used to generate self-signed server or client certificates. In a clustered environment, all SecureTransport Servers must share the same internal CA. See *Manage the internal CA* on page 57.
Keystore Password	SecureTransport stores all available certificates for the system in the database and in its own certificate keystore, protected by the keystore password. See *Change the certificate keystore password* on page 62.
LDAP Domain	An LDAP domain defines the connectivity to one or more LDAP servers. The default LDAP domain is queried for each account login attempt. See *LDAP domains* on page 478.
Local Certificate	A local certificate is used for server authentication, to decrypt incoming data, and to sign data or receipts. See *Manage local certificates and certificate signing requests* on page 48.

Term	Definition
Login Certificate	A login certificate is an account-level certificate used for logging in to SecureTransport. SecureTransport can be used to generate X509 login certificates, and administrators can import their own X509 login certificates and SSH keys. See *Manage login certificates* on page 527.
Login Restriction Policy	A login restriction policy limits end user access to SecureTransport Server or Edge. Policies can be assigned to an account, business unit or account template. See *Manage Login Restriction Policies* on page 811.
Login Restriction Rule	As part of a login restriction policy, a login restriction rule defines the login conditions for end users trying to access SecureTransport. See *Manage Login Restriction Policy rules* on page 813.
Login Settings	Login settings are global rules for enforcing authentication methods for administrators or end-users, such as client certificate authentication, LDAP, SSO, etc. See *Login settings* on page 465.
Network Zone	A network zone defines the communication between a SecureTransport Server and a SecureTransport Edge. Network zones can be used in a transfer site setup to determine if outgoing connections should be routed via proxy (Edge or HTTP proxy). See Create a network zone.
Partner Certificate	A partner certificate is an account-level public certificate used for encrypting PGP and AS2 data before sending it to the respective account, and for verification of the signature of data from this account. See *Manage partner certificates* on page 530
Private Certificate	A private certificate (including private keys) is an account-level certificate used to log in to a remote transfer site, and also for decrypting and signing PGP and AS2 data. See *Manage private certificates* on page 534.
Trusted CA	Trusted Certificate Authorities represent a list of root and intermediate CAs that are used to build the certificate chain for client and server certificates (for example, for protocol daemons and authentication to remote sites). See *Manage trusted CAs* on page 55.
Upload Restriction	An upload restriction is used to grant or deny users the permission to upload files within their home folder path.
User Class	A user class defines a set of SecureTransport users with common characteristics and privileges. User classes are used to set up login restrictions, map LDAP attributes, define SSL rules, or access resources. See *User classes* on page 771.

Processing

Term	Definition
Advanced Routing Application	When linked to a subscription, an Advanced Routing Application provides options to create multiple automated flows for file transformations, routing, and transfers between different participants, partner systems, and applications. See *Advanced Routing* on page 864.
Application	An application is a set of workflows you create to perform file processing or maintenance tasks. See *Manage applications* on page 819.
Basic Application	When linked to a subscription, a Basic Application can perform server-initiated transfers and limited data transformations. See *Basic Application* on page 832.
Home Folder	A home folder is the absolute path used by SecureTransport to create subscription folders, and to retrieve and process files from the account. In a clustered environment, the home folder should be in shared storage.
Publish To Account	A Publish To Account step enables SecureTransport to move files to a specified account (on the same SecureTransport Server) as part of a route. See *Publish To Account* on page 943.
Pull From Partner	A Pull From Partner step enables SecureTransport to pull files from an FTP, HTTP, Generic-HTTP(S), or SSH transfer site as part of a route. See *Pull From Partner* on page 956.
Route	A route consists of transformation and/or routing steps that are executed in a predefined order. Routes are part of route packages and route package templates. See *Manage Routes* on page 881.
Route Package	A route package is a collection of routes.
Route Package Template	A route package template is used for adding route packages per account. At least one route package template has to be created to set up an Advanced Routing flow. See *Manage Route Package Templates* on page 875.
Send To Partner	A Send To Partner step enables SecureTransport to send a file to a specified partner account as part of a route. See *Send To Partner* on page 948.
Shared Folder Application	A Shared Folder Application allows sharing data storage and files between accounts that are subscribed to the same application. See *Shared Folder application* on page 847.
Step	A step in a route is a file transformation or file routing action that can be based on a triggering condition.

Term	Definition
Subscription	A subscription is the connection between a user account and an application. For each subscription, SecureTransport creates a subscription folder in a user's home folder, where it stores and manages all files that are transferred or transformed as a result of application activity. See *Manage subscriptions* on page 664.
Transfer Profile	A transfer profile is a set of rules that a PeSIT transfer site uses when exchanging files. See *Transfer profiles* on page 640.
Transfer Site	A transfer site is a location such as a local folder or protocol server that is used when sending or receiving files during a server-initiated transfer. See *Transfer sites* on page 540.

Logging, reporting and notifications

Term	Definition
Action By	The Action By field on the *File Tracking* page indicates the initiator of a file transfer - either SecureTransport (Server) or a user account (User). Transfers that are initiated by SecureTransport are called server-initiated transfers (SITs). Transfers that are initiated by a user are called client-initiated transfers (CITs).
Active User	An active user is a user who has logged in to SecureTransport at least once in the last 60 days. See *Display active users* on page 758.
CoreID	A CoreID is a unique flow identifier associated with a specific file and its transformation and routing. In case of retrying or resubmitting a transfer, the CoreID is reused.
File Tracking	SecureTransport records information about the file transfers that it processes in a transfer log. For administrators, that information is accessible on the *File Tracking* page in the Administration Tool, and through the `/logs/transfers` REST API resource. For end users, transfer log data can be monitored through the `/transfers` REST API resource, and the Upload Monitor in ST Web Client. See *File Tracking* on page 302.
Monitor Server	The Monitor Server performs periodical checks to identify whether all SecureTransport Servers and Edge servers are functional and automatically restarts them if needed. See *Monitor Server* on page 293.
Session ID	A session ID is a unique identifier that is used to track the file(s) during one or multiple transfers that happened in the same session.
Transfer ID	A transfer ID is a unique identifier of a transfer in SecureTransport that is used to track what happened to a file during a single transfer.

Term	Definition
Transfer Log	The transfer log tracks all file exchanges on the system. Additional information is also recorded, such as the transfer protocol, the status of the transfer, and whether the transfer was initiated by the server or by a user. Tracking information is kept in the database and in a log file called `xferlog`. See *Configure transfer log* on page 192.

Maintenance

Term	Definition
Account Maintenance	A maintenance schedule that deletes, disables, or purges accounts based on their inactivity or age. See *Account Maintenance application* on page 823.
Audit Log Maintenance	A maintenance schedule that deletes audit log records from the database if they are older than the specified number of days/months. See *Audit Log Maintenance application* on page 828.
Axway Sentinel Link Maintenance	A maintenance schedule that removes all SentinelLinkData entries for files that no longer exist. See *Axway Sentinel Link Data Maintenance application* on page 830.
File Maintenance	A maintenance schedule that deletes files from account home folders based on a specified retention or expiration period. See *File Maintenance application* on page 834.
Log Entry Maintenance	A maintenance schedule that deletes server log records from the database if they are older than the specified number of days. See *Log Entry Maintenance application* on page 839.
Login Threshold Maintenance	A maintenance schedule that unlocks accounts which were locked as per the selected "Lock account after N successful logins" account settings option. See *Login Threshold Maintenance application* on page 845
Package Retention Maintenance	A maintenance schedule that deletes expired file packages from AdHoc file transfers. See *Package Retention Maintenance application* on page 846
Transfer Log Maintenance	A maintenance schedule that deletes transfer log records from the database if they are older than the specified number of days. See *Transfer Log Maintenance application* on page 856.
Unlicensed Account Maintenance	A maintenance schedule that deletes inactive unlicensed user accounts older than the specified number of days. See *Unlicensed Accounts Maintenance application* on page 861.