SecureTransport

Version 5.5

29 August 2024

Installation Guide

Copyright © 2024 Axway

All rights reserved.

This documentation describes the following Axway software:

Axway SecureTransport 5.5 Modernized Standard Cluster (Beta)

No part of this publication may be reproduced, transmitted, stored in a retrieval system, or translated into any human or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual, or otherwise, without the prior written permission of the copyright owner, Axway.

This document, provided for informational purposes only, may be subject to significant modification. The descriptions and information in this document may not necessarily accurately represent or reflect the current or planned functions of this product. Axway may change this publication, the product described herein, or both. These changes will be incorporated in new versions of this document. Axway does not warrant that this document is error free.

Axway recognizes the rights of the holders of all trademarks used in its publications.

The documentation may provide hyperlinks to third-party web sites or access to third-party content. Links and access to these sites are provided for your convenience only. Axway does not control, endorse or guarantee content found in such sites. Axway is not responsible for any content, associated links, resources or services associated with a third-party site.

Axway shall not be liable for any loss or damage of any sort associated with your use of third-party content.

Revision history

The following changes are added to the SecureTransport 5.5 Installation Guide:

SecureTransport 5.5 Updates	Content updates
August 2024	Multiple changes across the whole guide to reflect the newly added support for Secret Vault integration.
July 2024	• *Requirements for clustered deployments* on page 19 updated • *Requirements for installing on Linux* on page 15 updated
June 2024	• *System requirements* on page 13 updated • *Requirements for installing on Linux* on page 15 updated
May 2024	• *Requirements for installing on Linux* on page 15 updated
April 2024	• *Run SecureTransport as a service after non-root installation* on page 41 updated • *Requirements for installing on Windows* on page 18 updated
March 2024	• *Requirements for installing on Linux* on page 15 updated
December 2023	• Clarified that customers must have an Enterprise Clustering license to use an external database.
November 2023	Added more comprehensive documentation on how to check and set the resource limits required by SecureTransport. The following topics are updated: • *FAQ and Troubleshooting* on page 72 • *Requirements for installing on Linux* on page 15
October 2023	• *System requirements* on page 13 updated • *Requirements for installing on Linux* on page 15 updated • *FAQ and Troubleshooting* on page 72 updated with instructions on how hot to check and set resource limits
September 2023	• Complete rewrite of the *Plan Your Installation* on page 11 and *System requirements* on page 13 topics
May 2023	• *Silent installation* on page 52updated • *Requirements for Microsoft SQL Server databases* on page 27 updated

SecureTransport 5.5 Updates	Content updates
April 2023	• *Run SecureTransport as a service after non-root installation* on page 41 updated
March 2023	• OS-specific requirements updated
January 2023	• *Run SecureTransport as a service after non-root installation* on page 41 updated
September 2022	• Updates for clarity; installation instructions are separated into two sections based on the installation method - interactive or silent.
August 2022	• AIX requirements updated
May 2022	• *Requirements for external PostgreSQL databases* on page 26 updated
March 2022	• *Requirements for Microsoft SQL Server databases* on page 27 updated for clarity

Contents

Preface

This guide provides instructions for installing the SecureTransport software and provides information on the following topics:

- Installation prerequisites
- Installing SecureTransport or updating from previous SecureTransport 5.5 releases
- Required post-installation tasks
- Uninstalling SecureTransport

Who should read this guide

This guide is intended for system administrators who install SecureTransport on Windows and Linux and perform its initial configuration. As the SecureTransport installer, you must have a working knowledge of system platforms and networks used by your SecureTransport instances. You must have administrative privileges on the computers where you will install SecureTransport and appropriate access to systems that SecureTransport depends on, such as an external database and file system. This guide is also intended for enterprise personnel involved in installing software and Axway Professional Services personnel. Familiarity with Axway products is recommended.

This guide presumes you have knowledge of:

- Your company's business processes and practices
- Your company's hardware, software, and IT policies
- Your OS platform and network

Others who may find parts of this guide useful include network or systems administrators and other technical or business users.

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

Plan Your Installation 1

This topic contains information about the SecureTransport on-premises deployment options and considerations. For other deployment options (Azure, AWS, Virtual Appliance), refer to SecureTransport how-to guides.

Deployment options

During the installation you will be prompted to choose the deployment option. There are three options for on-premise deployment:

- **Standard Cluster** - Allows active/active or active/passive cluster operations. It uses an embedded MariaDB database in each node and has no dependency on an external database. You can have up to three servers (nodes) in an active/active Standard Cluster, or one active server and one standby server in an active/passive deployment.

  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.

- **Enterprise Cluster** - Allows high-performance cache-management layer significantly improves efficiency, provides near-linear scaling of up to 20 nodes, and enables very large scale configurations. An Enterprise Cluster requires your organization to provide and maintain an external database and a high-performance shared file system for the user files. Supported external databases: PostgreSQL, Microsoft SQL Server, Oracle Database. For details on supported versions, see Axway and third-party software support. The database requirements must be in place before installing SecureTransport. See *External database requirements* on page 14.

- **Standalone** - Suitable for small-scale operations. It can use either the embedded database or an external one, and a local file system for user files.

Note To use an external database in either a single-node or multi-node environment, you need a license for the Enterprise Clustering option.

Streaming deployment

SecureTransport Edge is the gateway used in the perimeter network (also called demilitarized zone or DMZ) in a typical multilayer security architecture deployment. It proxies the connections to and from external partner systems and the SecureTransport Servers in your internal secure network. For details, see Streaming deployment.

During the installation you will be prompted to choose between Server or Edge node.

System requirements

Refer to *System requirements* on page 13 for the hardware and operating system requirements for installing SecureTransport and consider your options.

Server certificates

During installation, SecureTransport generates a temporary self-signed CA, as well as a local temporary server certificate, named *admind*. Certificates, issued by a trusted CA, can be imported later and assigned to SecureTransport services.

Secure connection between SecureTransport and the database can be set up after the installation. Consider that you will need an X509 certificate, a private key, and the Certificate Authority (CA) chain. SecureTransport can generate and validate the required certificates as self-signed. You can also use certificates in PEM format generated by a trusted authority.

Default ports and firewall requirements

Consider a port to be used for the Administration Tool. The default is 444 for root installation and 8444 for non-root installation. Local port 8004 is used by the Administration Tool and should not be used by another process during installation. Consider the firewall rules that will be needed for protocol services, as well as clustered or streaming deployments. See Firewall settings.

System requirements 2

This section describes the system requirements for installing SecureTransport.

Hardware requirements

The following is the required minimum hardware for SecureTransport Servers and Edges (including cloud VMs):

	SecureTransport Server	SecureTransport Edge
CPUs	4	2
RAM	16 GB	8 GB
Free disk space	200 GB	100 GB

A functional network connection and DNS resolution is required for both Servers and Edges. Check Firewall settings.

In a virtualized or hyper-converged infrastructure, the listed CPU and RAM capacity must be reserved.

Additional resoures may be needed depending on the enabled services and load. For details, consult Performance tuning for increased transfer load.
For highly loaded servers, see the Capacity Planning Guide (login required).

Operating system requirements

SecureTransport can be deployed on a 64-bit (x86_64) version of the following operating systems:

- Linux (recommended)
- Windows (using Cygwin)

For more information on supported distributions and versions, see Axway and third-party software support.

OS user

Prepare a user account for the SecureTransport application. All SecureTransport processes will run as this service user account.
User privileges requirements:

- Linux: Local non-root or root user (see requirements)
- Windows: Local Administrator user (Installation by a domain administrator is not supported.)

Installation location

The SecureTransport installation directory must meet the requirements:

- Located on local disk storage. Installation on shared storage is not supported, except for private SAN logical unit LUN.
- Must not contain special characters, including tilde or white space.
- Have sufficient disk space (see requirements)
- All SecureTransport nodes in a clustered deployment must use the same local installation directory path name and have access to the shared storage for user data.

OS-specific requirements

- *Requirements for installing on Linux* on page 15
- *Requirements for installing on Windows* on page 18

Cluster requirements

- *Requirements for clustered deployments* on page 19
- *Secret file and vault integration* on page 19

External database requirements

- *Requirements for external PostgreSQL databases* on page 26
- *Requirements for Oracle databases* on page 28
- *Requirements for Microsoft SQL Server databases* on page 27

Support

SecureTransport relies on a stable and well-maintained network, storage, virtualization, and database infrastructure. If an issue related to these components is suspected, Axway Support may require your organization's involvement in troubleshooting and identifying a resolution, including engaging the respective vendor support teams.

Requirements for installing on Linux

This page lists the requirements for installing SecureTransport on Linux.

Required OS package dependencies

SecureTransport runtime requirements:

- `glibc`
- `glibc-langpack-en`
- `libaio`
- `zlib`

Axway Installer and operational scripts requirements:

- `bash`
- `awk`
- `grep`
- `ps`
- `sed`
- `which`
- `tar`
- `gzip`
- `unzip`

Additional requirements:

- For SecureTransport Update releases up to 5.5-20240328:
  - 32-bit `glibc.i686` (provides *ld-linux.so.2*)
  - 32-bit `zlib.i686`
- For deployments with MariaDB embedded database:
  - *perl.x86_64* with these Perl packages:
    - *perl-Data-Dumper.x86_64*
    - *perl-Getopt-Long.noarch*
  - *ncurses-compat-libs.x86_64*
  - *libxcrypt-compat.x86_64*
    Alternatively, on RHEL 9, you can create the following symbolic links instead:

```
ln -s /usr/lib64/libncurses.so.6 /usr/lib64/libncurses.so.5
ln -s /usr/lib64/libtinfo.so.6 /usr/lib64/libtinfo.so.5
ln -s /usr/lib64/libcrypt.so.2 /usr/lib64/libcrypt.so.1
```

OS user (non-root)

The OS user that will run SecureTransport must meet the following requirements:

- Be defined in the `/etc/passwd` file.
- Have a valid shell that allows login and command execution (`/bin/bash` or `/bin/sh`).
- Be the owner of an existing and accessible home folder with Read, Write and Execute permissions.
- Have access to a temporary directory with sufficient free space (20 GB recommended) and Read, Write, and Execute permissions. (It must not reside on a `noexec` mounted filesystem).

Note The SecureTransport installation routine calls crontab to set up some entries for log rotation and process monitoring. To instruct the installer to skip the calls to crontab:
`export INSTALL_CRON=false`

OS limits

- Sufficient resource limits set at the following levels: system, user, and service. The table below shows where and which parameters need to be checked and tuned if their values do not meet the SecureTransport requirements.

Resource	Parameter in the systemd unit files for the SecureTransport services (service level)	Parameter in system.conf (system-wide)	Parameter in limits.conf (user Level)	Minimum Value
Max open files	LimitNOFILE	DefaultLimitNOFILE	nofile	65536
Max processes	LimitNPROC	DefaultLimitNPROC	nproc	65536
Max locked memory	LimitMEMLOCK	DefaultLimitMEMLOCK	memlock	4194304
Max threads per process	TasksMax	DefaultTasksMax	-	65536

For instructions on how to check and set resource limits, see *FAQ and Troubleshooting* on page 72.
For details on how and where to create service files, see *Run SecureTransport as a service after non-root installation* on page 41.

- Check if the OS kernel parameters meet the requirements of SecureTransport installation. The following table lists the required parameter values and the commands you can use to check the current limits and set new values without reboot.

Resource	Parameter in sysctl.conf (system-wide)	Value	Check system-wide limits	Set sysctl.conf and apply dynamically

Max receive window*	net.core.rmem_max	12852000 or higher	cat /proc/sys/net/core/rmem_max
			or
			sysctl net.core.rmem_max
				Edit the file /etc/sysctl.conf to add
				`net.core.rmem_max=12852000`
				Apply with the command
				`sysctl -p`

Max send window*	net.core.wmem_max	12852000 or higher	cat /proc/sys/net/core/wmem_max
			or
			sysctl net.core.wmem_max
				Edit the file /etc/sysctl.conf to add
				`net.core.wmem_max=12852000`
				Apply with the command
				`sysctl -p`

TCP autotuning*	net.ipv4.tcp_moderate_rcvbuf	1	cat /proc/sys/net/ipv4/tcp_moderate_rcvbuf
			or
			sysctl net.ipv4.tcp_moderate_rcvbuf
				Edit the file /etc/sysctl.conf to add
				`net.ipv4.tcp_moderate_rcvbuf=1`
				Apply with the command
				`sysctl -p`

Resource	Parameter in sysctl.conf (system-wide)	Value	Check system-wide limits	Set sysctl.conf and apply dynamically
Max number of file descriptors	fs.file-max	65536 or higher	cat /proc/sys/fs/file-max or sysctl fs.file-max	Edit the file /etc/sysctl.conf to add `fs.file-max=65536` Apply with the command `sysctl -p`

* required for Enterprise Cluster

- Security hardening modules like SELinux or SUSE AppArmor must not be in Enforcing mode during installation.
  - SELinux can be disabled or set to permissive mode in */etc/sysconfig/selinux*.
  - AppArmor can be disabled using Yast from *System Services(Runlevel)>Expert Mode*.
- System locale must be set to `LC_CTYPE=en_US.UTF-8`.
- System `umask` must be set to one of the following:
  - on non-root installations: 077, 027, 022
  - on root installations: 022
- The host name of the server must resolve to its actual IP address (not the loopback address 127.0.0.1).
- Any running antivirus software must be disabled during the installation.

Requirements for installing on Windows

This page lists the requirements for installing SecureTransport on Windows.

- *Microsoft Visual C++ Redistributable Package (x64)* latest version installed on the system.

  Additional requirements for Update releases up to 5.5-20240328: *Microsoft Visual C++ 2010 SP1 Redistributable Package (x64)*.
- Local Administrator user (Installation by a domain administrator is not supported.)
- A temporary directory with sufficient free space (20 GB recommended), write and execute permissions.
- Any running antivirus software must be disabled during the installation.
- Any application firewalls, which could block SecureTransport execution or network access, must be disabled.

Requirements for clustered deployments

Make sure the following requirements are met before installing SecureTransport in Legacy Standard Cluster (SC) or Enterprise Cluster (EC):

- All SecureTransport Servers in a cluster must run on the same operating system.
- All SecureTransport Edge servers that synchronize configuration must run on the same operating system.
  However, in a multi-tier security deployment, the operating system for SecureTransport Server systems can be different from the operating system for the SecureTransport Edge systems.
- All your Server nodes must have the same SecureTransport version. Do not attempt to add Server nodes with a different SecureTransport version to the cluster.
- All nodes must use the same SecureTransport installation path.
- All nodes use share the same database schema. The schema is created when you install the first server in the cluster.
- All nodes share the same secret file (named `taeh` file.
  You can have the installer create the `taeh` file or import an existing file for the first server in the cluster. You must copy and import the `taeh` file to the second and subsequent servers in the cluster before you install. For more information, see *Secret file and vault integration* on page 19.
- To use an external database in either a single-node or multi-node environment, you need a license for the Enterprise Clustering option.
- The database password must not contain the dollar sign (`$`) and it is recommended not to contain any special characters due to specific database restrictions, e.g., PostgreSQL does not allow `%`, `&` or `+`, Microsoft SQL Server does not allow curly brackets (`{,}`), and MariaDB does not allow `%`.
- For Enterprise Cluster, check the prerequisites for your database and make sure it is properly configured. See *External database requirements* on page 14. Also, check if the database connection pool is properly sized: it depends on the number of nodes, allowed number of max connections per node and running services. For more information on connections pool settings, refer to the "Connection Pool Configuration" section in the SecureTransport Administrator's Guide.

Secret file and vault integration

The SecureTransport secret can be stored internally, inside a file, or externally, inside a secret vault.

Secret file

The secret file (also called `taeh` file) contains randomly-generated data used by SecureTransport for encryption. It must be the same for all SecureTransport Servers or SecureTransport Edges in a cluster.

The installer creates the secret file (named `taeh`) when you install a standalone server or the first SecureTransport Server in a cluster. It is stored in the `<FILEDRIVEHOME>/bin/taeh` directory, where `<FILEDRIVEHOME>` is the directory where SecureTransport is installed.

For the second and subsequent servers of the cluster, you must copy the `taeh` file from the first cluster node before you install SecureTransport. You import the secret file during server installation. After you configure the server, you cannot change the encryption secret as this will prevent the decryption of operation-critical configuration elements, like certificates and passwords.

Store the secret in an external vault

As of Update 5.5-20240829, the SecureTransport secret can be stored in a *HashiCorp Vault*.

Vault integration is enabled at runtime if a `taeh.properties` file exists in the `<FILEDRIVEHOME>/conf` directory. This file is created by SecureTransport based on a specific `.properties` file, and contains the details for connecting to the vault and accessing the secret. The connection is established via REST using AppRole authentication.

This feature is supported on:

- existing deployments - see the *Secret vault integration* section in the *SecureTransport 5.5 Administrator Guide*.
- fresh installations - see the procedure below.

Prerequisites

- A configured and ready to use HashiCorp Vault, with established connectivity.

Generate your secret

If you have an existing `taeh` file from a previous SecureTransport deployment, you can reuse the secret. Otherwise, you can manually create a base64-encoded vault secret containing randomly-generated 1024 bytes of binary data in two ways:

- with OpenSSL:

```
echo "$(openssl rand -hex 1024)" | xxd -r -p |  base64 -w 0
```

- with the HashiCorp Random service:

```
curl -X 'POST' \
'https://<vaulthost>:<port>/v1/sys/tools/random' \
-H 'accept: application/json' \
-H 'Content-Type: application/json' \
-H 'X-Vault-Token: <token>' \
-d '{
"bytes": 1024,
"format": "base64"
}'
```

```
```

Import the secret in the vault

Note The secret must be base64-encoded.

1. Add the secret root path, for example *st*, and enable the *v2* secret engine:

```
curl --header 'X-Vault-Token: <token>' \
--request POST \
--data '\{ "type":"kv-v2" }' \
https://<vaulthost>:<port>/v1/sys/mounts/st
```

2. Add the secret in the vault, e.g., under the *st/data/taeh* path.
   In this example, we import the *base64secret* as the value of the *secret* key within the *data* JSON object.

```
curl --header 'X-Vault-Token: <token>' \
--request POST \
--data '\{ "data" : {"secret":"<base64secret>"} }' \
https://<vaulthost>:<port>/v1/st/data/taeh
```

Set up application authentication

1. Enable *approle* authentication:

```
curl \
--header "X-Vault-Token: <token>" \
--request POST \
--data '{"type": "approle"}' \
http://<vaulthost>:<port>/v1/sys/auth/approle
```

2. Create a vault policy allowing SecureTransport to read the secret:

```
curl \
--header "X-Vault-Token: <token>" \
--request POST \
--data '{ "policy" : "path \"st/data/taeh\" \n {\n capabilities=[\"read\"]\n}\n"
}' \
http://<vaulthost>:<port>/v1/sys/policies/acl/st-secret-readonly
```

3. Create the application role and save the *role_id*:

```
curl \
--header "X-Vault-Token: <token>" \
```

```
--request POST \
--data '{"bind_secret_id": true,   "local_secret_ids": false,  "token_
type":"batch"}' \
http://<vaulthost>:<port>/v1/auth/approle/role/st_app

curl \
--header "X-Vault-Token: <token>" \
--request GET\
https://<vaulthost>:<port>/v1/auth/approle/role/st_app/role-id
```

4. Create the *secret_id* and save the response. The *secret_id* is required for authentication and cannot be retrieved later.

```
curl \
--header "X-Vault-Token: <token>" \
--request POST \
--data '{"ttl": "0"}' \
http://<vaulthost>:<port>/v1/auth/approle/role/st_app/secret-id
```

5. Assign the policy to the application role:

```
curl \
--header "X-Vault-Token: <token>" \
--request POST \
--data '{"token_policies": [
    "st-secret-readonly"
  ] }' \
https://<vaulthost>:<port>/v1/auth/approle/role/st_app/policies
```

Configure the vault client connection in SecureTransport

To set up the vault client connection:

1. Navigate to `<AxwayHome>/Components/Configuration/vault` and locate the two template files:
   - `hashicorp-vault-conf-template.properties` - for the default HashiCorp Vault setup.
   - `vault-conf-template.properties` - for a more advanced or non-standard vault setup.
2. Copy one of the templates in a temporary location. Keep the .properties extension.
3. Edit the file to configure the vault connectivity, authentication and full secret path.

   The table below provides a description and example values for the vault properties. As a minimum, specify the following: *vault.api.base-url*, *vault.api.ca*, *vault.api.auth.body*, *vault.api.fetch.json-path-prefix*, *vault.secrets.taeh.uri-suffix*, *vault.secrets.taeh.json-path*. You can leave the rest of the properties unchanged.

Property	Description	Example value
`vault.api.base-url`	Vault server address. Communication is supported only over HTTPS.	`https://vaulthost.example.com:8200`
`vault.api.ca`	The path to the vault server CA(s) file in PEM format.	`/home/TMWD/example-ca.pem`
`vault.api.insecure`	If set to true, skips server certificate validation. For testing purposes only.	`false`
`vault.api.skip.hostname.verification`	If set to true, skips host name verification. For testing purposes only.	`false`
`vault.api.tls.protocols`	A comma-separated list of TLS protocols for vault connection. If left empty, the default SecureTransport TLS protocols are used.	`TLSv1.3, TLSv1.2`
`vault.api.tls.ciphers`	A comma-separated list of TLS cipher suites for vault connection. If left empty, the default cipher suites are used. If set to `FIPS`, the SecureTransport list of FIPS cipher suites is used.	`TLS_AES_256_GCM_SHA384, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384` `FIPS`
`vault.api.open-timeout`	Open connection timeout, in seconds	`5`
`vault.api.read-timeout`	Read response timeout, in seconds	`30`

Property	Description	Example value
`vault.api.max-retries`	Maximum number of retries for vault client authentication and fetch secret operation	`3`
`vault.api.retry-interval-ms`	Waiting interval between failures, in milliseconds	`100`
`vault.api.auth.path`	Authentication URI. Leave empty if authentication API is not used.	`/v1/auth/approle/login`
`vault.api.auth.body`	Authentication request JSON body. Only HTTP POST is supported. Specify the *role_id* and *secret_id* that you saved during the *Create application authentication* procedure.	`{"secret_id":"<value>", "role_id":"<value>"}`
`vault.api.auth.token`	Authentication token JSON path in the authentication response	`$.auth.token`
`vault.api.auth.header.<header name>=<header value>`	Authentication API request headers	`vault.api.auth.header.X-Vault-Request=true` `vault.api.auth.header.X-Vault-Namespace=<Namespace>`

`vault.api.auth.url`
`vault.api.auth.ca`
`vault.api.auth.insecure`
`vault.api.auth.tls.ciphers`
`vault.api.auth.tls.protocols`
	These properties should be used only if the authentication token is generated by another application (Identity Provider).	

Property	Description	Example value
`vault.api.fetch.uri`	Secret API URI. Only HTTP GET is supported.	`/v1`
`vault.api.fetch.json-path-prefix`	JSON path prefix for the secret in the response body, as defined in the *Import the secret in the vault* procedure.	`$.data.data`
`vault.api.fetch.auth-header.name`	Authentication header name	`X-Vault-Token`
`vault.api.fetch.auth-header.value`	Authentication header value. It can be set to an expression using `${vault.api.auth.token}`, or it can be a static value. If left empty and the `vault.api.auth.path` and `vault.api.auth.token` are set, the `token` will be used.	`${vault.api.auth.token}`
`vault.api.fetch.header.<header name>=<header value>`	Fetch API request headers	`vault.api.fetch.header.X-Vault-Request=true` `vault.api.fetch.header.X-Vault-Namespace=<namespace>`
`vault.secrets.taeh.uri-suffix`	Specific path parameter for the `taeh` file that will be appended to the `vault.api.fetch.uri`, as defined in the *Import the secret in the vault* procedure.	`/st/data/taeh`

Property	Description	Example value
`vault.secrets.taeh.json-path`	Secret relative JSON path to the secret field that will be appended to `vault.api.fetch.json-path-prefix`, as defined in the *Import the secret in the vault* procedure.	`.secret`

Requirements for external PostgreSQL databases

- For a complete list of supported PostgreSQL versions, refer to *Axway and third-party software support*.
- SecureTransport can connect to an external PostgreSQL database over a plain or secure connection. Starting with SecureTransport 5.5-20201029, a certificate must be present for both secure and non-secure connections, and the external database server's certificates that contain `keyUsage` extensions MUST also have the `digitalSignature` indicator enabled.
- Server requirements:
  - physical or virtual server with 8 vCPUs or more
  - 64 GB RAM on the server (we assume the server will be used only for PostgreSQL) or more
- Parameters as follows:
  - `max_connections`: 1000
  - `shared_buffers`– 1/4 of the available RAM
  - `effective_cache_size`– 3/4 of the available RAM
  - `maintenance_work_mem`– at least 512MB
  - `work_mem`– half of the available RAM
  - `autovacuum` should be turned on
- Tablespace requirements:
  The following tablespaces have to be created:
  - `st_data` – configuration, such as account, sites, and certificates
  - `st_filetracking` – file tracking tables
  - `st_serverlog` – server log tables

All three tablespaces should be created on fast disks (preferably SSDs) on location with enough free space to handle at least the predicted SecureTransport workload multiplied by two.

- User/login/role requirements:

  The role, which will be used for SecureTransport, should be created with the `LOGIN` privilege. The user/login/role must be granted `CREATE` privileges on the above three tablespaces.

- Database requirements:
  - Database should be created with (default) tablespace `st_data`.
  - The owner of the database should be the above mentioned user/login/role.
- Password:

  For SecureTransport Server Enterprise Cluster installations, the database password must not contain any of the following symbols: `%`, `&`, `+`, `$` or other special characters.

During the installation, you need the following information:

- Host name or IP address
- Listener port number
- Database user name and password
- If the database is used over secure connection, you will also need the public key of the database server certificate.

Requirements for Microsoft SQL Server databases

- For a complete list of supported Microsoft SQL Server versions, refer to *Axway and third-party software support*.
- The Microsoft SQL Server collation must be defined as case insensitive (SQL_Latin1_General_CP1_CI_AS).
- SecureTransport can connect to an external Microsoft SQL Server database over a plain or secure connection. Starting with SecureTransport 5.5-20201029, a certificate must be present for both secure and non-secure connections, and the external database server's certificates that contain `keyUsage` extensions MUST also have the `digitalSignature` indicator enabled.
- The database must have the `READ_COMMITED_SNAPSHOT` option set to ON.

  To check if the option is enabled, execute the following query:

  SELECT `is_read_committed_snapshot_on` FROM `sys.databases` WHERE name = *yourdatabase*

  If it is not set, you can set it by executing the following:

  ALTER DATABASE *yourdatabase* SET `READ_COMMITTED_SNAPSHOT` ON

- The database must have the following filegroups defined with at least one file in each filegroup:
  - `ST_DATA` – configuration, account, sites, and certificates (the default filegroup)
  - `ST_FILETRACKING` – file tracking tables
  - `ST_SERVERLOG` – server log tables

    Note You cannot direct log data to separate Microsoft SQL Server databases.
- If you would like to use the export functionality of the Transfer Log or the Log Entry maintenance application (not recommended if another database backup solution is in place), two additional filegroups need to be created: `ST_FILETRACKING_ARCHIVE` and `ST_SERVERLOG_ARCHIVE`. In this case, the database user needs write permissions on the export directory, as well as Backup database and Backup log permissions.
- The default filegroup for the database user must be `ST_DATA`.
- The database must have a user defined that is mapped to the login that SecureTransport will use to access the database.
- The authentication mode for the user must be Microsoft SQL Server Authentication.
- The user must either have the `db_owner` role or be a member of these three fixed roles `db_datareader`, `db_datawriter`, and `db_ddladmin`. If you have a custom setup, make sure the user has permission to execute stored procedures.
  Users with the `ddladmin` role can apply changes to the database schema. This role is required during upgrade, service pack and fresh installations.
- The database password cannot contain curly brackets (`{,}`). In case of EC, it cannot contain the dollar sign (`$`) or other special characters.

During the installation, you need the following information:

- Database access port number
- Database user name and password
- Database name
- If the database is used over secure connection, you will also need the database server certificate signer or JKS keystore which contains it in order to trust the connection. In addition, the optional parameter **Common Name (CN)** can be provided and installer will verify the provided information against the database certificate's CN. If it is not provided, the installer will not check the certificate CN and will trust any.

Requirements for Oracle databases

- For a complete list of supported Oracle Database releases, refer to *Axway and third-party software support*.
- SecureTransport can connect to an external Oracle database over a plain or secure connection. Starting with SecureTransport 5.5-20201029, a certificate must be present for both secure and non-secure connections, and the external database server's certificates that contain `keyUsage` extensions MUST also have the `digitalSignature` indicator enabled.

- To support unicode characters in filenames and directories, the database should use AL32UTF8 encoding.
- The Oracle database can use, but does not require, the Real Application Clusters (RAC) option.
- SecureTransport can use, but does not require, more than one Oracle database to store its data.
- Settings and parameters as follows:
  - Redo log groups: 3
  - Redo log file size: 500 MB (For more about redo log file use, see *SecureTransport 5.5 Capacity Planning Guide*)
  - Gather optimizer statistics: Weekly or with any 10 percent change in the record count.
  - `DB_CACHE_SIZE`: 1 GB or larger. You should set this as high as possible to improve performance.
  - `OPEN_CURSORS`: at least 1000
  - `SHARED_POOL_SIZE`: 150 MB per node in the cluster
  - `PROCESSES`: 1000 or more
- The database must have the following tablespaces defined:
  - `ST_DATA` – configuration, such as account, sites, and certificates. This must be the default tablespace for the database user.
  - `ST_FILETRACKING` – file tracking tables
  - `ST_SERVERLOG` – server log tables

    Note When you direct log data to separate databases, you must define the `ST_FILETRACKING` tablespace in the database for the transfer log and the `ST_SERVERLOG` tablespace in the database for the server log.

  Set `AUTOEXTEND ON` for all tablespaces and datafiles.

  See *SecureTransport 5.5 Capacity Planning Guide* for information you can use to set initial sizes for the tablespaces. Give the user unlimited quota on all tablespaces.
- The database must have a user defined who have been granted the following system privileges:
  - `CREATE OPERATOR`
  - `CREATE PROCEDURE`
  - `CREATE SEQUENCE`
  - `CREATE SESSION`
  - `CREATE TABLE`

  These privileges can be granted through a role or, preferably, directly. There are a few restrictions in the use of roles:
  - UPDATES_DB_LOG table is not populated, therefore logging for DB errors during installation or upgrade is limited
  - To install updates, the `update.sh` script should be run with an additional argument.

    - Transfer Log Maintenance and Log Entry Maintenance applications fail to export old partitions with "ORA-31623: a job is not attached to this session via the specified handle" error.
- For SecureTransport Server Enterprise Cluster installations, the database password cannot contain the dollar sign (`$`) or other special characters.

During the installation, you need the following information:

- Host name, IP address, or SCAN name of the database server in case Oracle RAC is utilized
- Listener port number
- Database user name and password
- Service name
- If the database is used over secure connection, you will also need the database server certificate signer or JKS keystore which contains it in order to trust the connection. In addition, the optional parameter **Distinguished Name (DN)** can be provided and installer will verify the provided information against the database certificate's DN. If it is not provided, the installer will not check the certificate DN and will trust any.

Data pump database management system

Before installation, you can set the `DATA_PUMP` system environment variable to either **false** or **true** depending upon if you want to *disable* or *enable* the data pump. By default, if data pump system environment variable is not set, the data pump will be enabled and export database functionality procedures will be deployed.

Execute the following command to set the `DATA_PUMP` system environment variable to `false`:

```
export DATA_PUMP=false
```

When the `DATA_PUMP` system environment variable is set to `false`, the installer will detect that the data pump is disabled and a warning message will be displayed in the `install.log` file with the following content:

```
DEBUG [external_db_configuration] 2016-07-05 15:47:25,761 EEST WARN [main]
com.tumbleweed.st.server.appframework.util.OracleDatabaseConfigurator -
DataPump capabilities are disabled. The configuration will proceed without
deploying the data pump procedures.
```

When the data pump disabled, the installer will not deploy the database procedures to export partitions. Microsoft SQL Server to Oracle database migrations will assume that the target database and user have data pump available. When deploying on multiple databases, all databases inherit the same data pump behavior as the first one. On upgrade, also valid for all patches and service packs, data pump will be enabled by default and you cannot change it.

Interactive installation 3

In interactive mode, you provide configuration information on the command line. The following topics contain the procedures for installing SecureTransport by using the interactive console method on Windows and UNIX-based platforms:

- *Install SecureTransport on Linux on page 31*
- *Install SecureTransport on Windows on page 44*

Install SecureTransport on Linux

SecureTransport 5.5 uses the Axway Installer in console mode for new installations. To navigate through the wizard, use the following commands:

- To go to the next dialog, type `Next`, `N`, or `n` or press Enter if the installer prompts you with `>Next`.
- To go to the previous dialog, type `Previous`, `P`, or `p`.
- To exit the installation type `Quit`, `Q`, or `q`.

For field values, use the following:

- To edit an option, select it by typing its number from a list and enter the new value.
- To accept a value, press Enter.

The following topics provide the how to instructions for installing SecureTransport on Linux:

- *Install SecureTransport with embedded database on Linux on page 31*
- *Install SecureTransport with external database on Linux on page 37*

Install SecureTransport with embedded database on Linux

Before you install SecureTransport, check the prerequisites for Linux deployments here.

This section provides instructions for installing SecureTransport on Linux where the installation will use an embedded MySQL or MariaDB database:

- Stand-alone SecureTransport Server
- SecureTransport Server in a Standard Cluster
- SecureTransport Edge

Installation procedure

This procedure assumes that SecureTransport is not installed on the system. Only one instance of SecureTransport Server or SecureTransport Edge is supported in a production environment.

1. Log in to the system as the user who will run SecureTransport.
2. Download the SecureTransport install package for your operating system:

   `SecureTransport_5.5_Install_<OS>-<processor>_<BuildNumber>.zip`

   where the placeholders represent the following:

   - `<OS>` is the operating system: `linux`
   - `<processor>` is the type of processor running the operating system: `power-64` or `x86-64`.
   - `<BuildNumber>` is the actual build number listed in the installer executable file.
3. Copy the install package into a temporary directory and navigate to that temporary directory.
4. Extract the installation files using the following commands:

   ```
   unzip SecureTransport_5.5_Install_<OS>-<processor>_<BuildNumber>.zip
   ```
5. Enter the following commands to run the Axway Installer:

   ```
   ./setup.sh -m console
   ```

   The installer initializes and displays a welcome message and a prompt.
6. Press Enter.

   The installer will display the license agreement page by page. After each one there is a **Press ENTER to continue** prompt. After all license agreement pages are displayed, the installer displays the license agreement and the following prompt (for accepting or rejecting the license agreement):
7. Enter 1 to accept the license agreement. Enter 2 to reject the license agreement and cancel the installation.
8. On the prompt for Installation directory, specify an installation location for Axway installer:

   ```
   ---------------------------------------
   Installation directory
   ---------------------------------------
   Specify the directory where you want to install the products and
   documentation.
   1: Installation Directory:           /<userHome>/Axway


   Enter 1 to select an option or (Next, Previous, Quit).
   :>Next
   ```

- To accept the default installation directory, press Enter. The default installation directory for Axway Installer is `/<userHome>/Axway`, where `<userHome>` represents the home directory of the user running the installation.
- To change the installation location, enter 1 and then enter the absolute path to your desired installation directory.

In this directory, the Axway installer files are deployed. It is used as the parent directory in the default SecureTransport installation location which you specify at a later step (10).

Note SecureTransport should not share the same installer directory with another Axway products.

9. Choose the installation type. Press Enter to accept the default, SecureTransport Server.

```
----------------------------------------
Modules
----------------------------------------
Select the modules you want to install, then type Next to continue the
configuration.

Axway SecureTransport V5.5:
1 :[x] Server
2 :[ ] Edge
3 :[ ] ServerDocker
4 :[ ] EdgeDocker


Enter a number[1-4] (Next, Previous, Quit)
:>Next
```

Note The `EdgeDocker` and `ServerDocker` modules are used **ONLY** in the docker image build process and are not supported for other purposes.

10. Specify an installation directory for SecureTransport:

```
----------------------------------------
Installation directory
----------------------------------------
To specify the directory where the product is installed, type the path
directly, or press Enter to select the displayed default.
1: Select the installation directory for SecureTransport:
<AxwayHome>/SecureTransport


Press 1 to change the selected option or (Next, Previous, Quit).
:>Next
```

- To accept the default location, press Enter. The default SecureTransport installation folder is `<AxwayHome>/SecureTransport`, where `<AxwayHome>` is the directory you specified in step 8.
- To change the SecureTransport installation location, enter 1 and enter the absolute path to your desired installation directory.

Consider the following:

- Do not use the directory where you copied the installer files.
- All SecureTransport Servers in a cluster must use the same installation directory.
- The name of the SecureTransport installation directory cannot contain space characters, the tab character, or the ~ character. For example, `/root/Axway/STServer` is valid, but `/root/Axway/ST Server` is not.

This installation directory is referred to as `<FILEDRIVEHOME>` throughout this document and other SecureTransport documents.

11. On the database selection prompt, press Enter to accept the default, the embedded database.

```
----------------------------------------
Database settings
----------------------------------------
[1] Embedded Database (MariaDB)
 or [1] Embedded Database (MySQL)
[2] External Oracle Database
[3] External Microsoft SQL Server Database
[4] External PostgreSQL Database


Enter a number [1-4] to select an option or (Previous, Quit).
:>1
```

12. The following step depends on whether you are using MySQL or MariaDB.

On MySQL, the installer displays the following dialog:

```
----------------------------------------
Database settings
----------------------------------------
Provide the settings for the MySQL database:
1: Port: 33060


Press 1 to change the selected option or (Next, Previous, Quit).
:>Next
```

The required information is:

- **Port:** select a new port number for the embedded database installed by SecureTransport or accept the default setting.

On MariaDB, provide answers to the following dialog:

```
----------------------------------------
Provide the settings for the MariaDB database:
1: Port: 33060
2: Use secure connection: true
3: Auto generate certificates: true
4: Certificate Authority File Path:
```

```
5: Server Private key File Path:
6: Server Certificate File Path:

Enter a number [1-6] to select an option or (Next, Previous, Quit).
:>Next
```

The required information is:

- **Port**: select a new port number for the embedded database installed by SecureTransport, or accept the default setting.
- **Use secure connection**: When selected, the database connection will be established over a secure connection. The default value is `true`.
    - **Auto generate certificates**: When set to true, the certificates required for a secure connection to the database will be automatically generated by SecureTransport with a validity period of 365 days. In this case, any value in **Certificate Authority File Path**, **Server Private key File Path** or **Server Certificate File Path** is ignored.
- **Certificate Authority File Path**: The absolute file path to the CA certificate in PEM format, used to create and sign the Server Private key and Server Certificate files.
  Only applicable when **Use secure connection** is set to `true` and **Auto generate certificates** is set to `false`.
- **Server Private key File Path**: A private key in PEM format, used by the MariaDB Server.
  Only applicable when **Use secure connection** is set to `true` and **Auto generate certificates** is set to `false`.
- **Server Certificate File Path**: an X.509 certificate in PEM format used by the MariaDB Server.
  Only applicable when **Use secure connection** is set to `true` and **Auto generate certificates** is set to `false`.

13. The installer displays the default SecureTransport ports, nightly log rotation, and import secret file configuration:

```
----------------------------------------
Configuration
----------------------------------------
  Select the options that you want to enable:
  SecureTransport Ports
1: SSL Admin UI Port:           444
2: Tomcat Shutdown Port:        8005
3: Enable Nightly Log Rotation: true
  Import Secret File
To synchronize the configuration of this ST Server with another ST
Server you must import the same secret file (located on the remote ST
```

```
Server at: <installation path>/bin/taeh). Otherwise, leave the field
empty to generate a new secret file.
4: Secret File Path:

 Enter a number [1-4] to select an option or (Next, Previous, Quit).
:>Next
```

14. Accept or modify the default configuration.

    The information required is:

    - **SSL Admin UI Port** – The port used to connect to the Administration Tool. When you install SecureTransport as a non-root user, the default value for Admin port number is 8444.
    - **Tomcat Shutdown Port** – The port used to shut down the Tomcat server
    - **Enable Nightly Log Rotation** – Select if you want the system to perform automatic backup and purging of log files on a nightly basis. When this feature is enabled, SecureTransport Server backups log files, generated on the respective day, and creates a new one for the subsequent day. The server takes a back up and creates a new log file at 23:59 or 00:00 hours, depending on the log file type. This option is enabled by default. You can enable or disable the nightly log rotation after installation. For more information, see the *SecureTransport 5.5 Administrator Guide*, Server Log Rotation Scheduling.
    - **Secret File Path** – The path to either the secret (`taeh`) file or the vault `.properties` file, depending on where the secret is stored - internally in SecureTransport or externally in a vault. If you leave it blank, the installer creates a new secret `taeh` file. See *Secret file and vault integration on page 19*.
      For the second and subsequent servers of the cluster, you must copy the respective file from the first cluster node before you install SecureTransport.

    When you have modified these values as required for your installation, press Enter.

    The installer prepares the installation execution and displays its last prompt:

```
----------------------------------------
Installation execution
----------------------------------------
All selected products are ready to install. Type Next to start installing.
If not, type Previous to make changes.


Enter (Next, Previous, Quit).
>Next
```

15. Press Enter to start the installation.

    The installer displays progress messages as it completes the installation tasks. When the installation is complete, a success message is displayed.

16. Press Enter to exit the installer or select update mode to apply patches or service packs without leaving the installer.

The installer also creates a log file, `<AxwayHome>/install.log`.

After successfully installing SecureTransport, you must perform a number of post-installation steps, such as applying your SecureTransport licenses, and enabling, configuring, and starting the SecureTransport services. For more information, see *SecureTransport Getting Started Guide*.

Install SecureTransport with external database on Linux

Before you install SecureTransport, check the prerequisites for Linux deployments here.

This section provides instructions for installing SecureTransport on Linux where the installations will use an external database server:

- SecureTransport Server in an Enterprise Cluster. See *Requirements for clustered deployments* on page 19
- Stand-alone SecureTransport Server when an external database server (PostgreSQL, Oracle, or Microsoft SQL Server) is required

To use an external database, either in a single or multi-node environment, you need a license for the Enterprise Clustering (EC) option.

Note In Enterprise Cluster deployments, all your SecureTransport Server nodes must have the same version as your first installed Server node. Do not attempt to add Server nodes to your cluster if the SecureTransport version does not match the one on already installed nodes.

Installation procedure

This procedure assumes that SecureTransport is not installed on the system. Only one instance of SecureTransport Server or SecureTransport Edge is supported in a production environment.

1. Log in to the system as the user that will run SecureTransport.
2. Download the SecureTransport install package for your operating system:

   `SecureTransport_5.5_Install_<OS>-<processor>_<BuildNumber>.zip`

   where the placeholders represent the following:

   - `<OS>` is the operating system: `linux`
   - `<processor>` is the type of processor running the operating system: `power-64` or `x86-64`.
   - `<BuildNumber>` is the actual build number listed in the installer executable file.
3. Copy the install package into a temporary directory and navigate to that temporary directory.
4. Extract the installation files using the following command:

```
unzip SecureTransport_5.5_Install_<OS>-<processor>_<BuildNumber>.zip
```

5. Enter the following command to run the Axway Installer:

```
./setup.sh -m console
```

   The installer initializes and displays a welcome message and a prompt.

6. Press Enter.

   The installer will display the license agreement page by page. After each one there is a **Press ENTER to continue** prompt. After all license agreement pages are displayed, the installer displays the following prompt (for accepting or rejecting the license agreement):

```
[1] I accept the terms of the license agreement.
[2] I do not accept the terms of the license agreement.
Enter a number [1-2] to select an option or (Previous, Quit).
:>2
```

7. Enter 1 to accept the license agreement. Enter 2 to reject it and cancel the installation.
8. On the prompt for the Axway installer directory:
   - To accept the default installation directory, press Enter. The default installation directory for Axway Installer is `/<userHome>/Axway`, where `<userHome>` represents the home directory of the user running the installation.
   - To change the installation location, enter 1 and then enter the absolute path to your desired installation directory.

   In this directory, the Axway installer files are deployed. It is used as the parent directory for the default SecureTransport installation location which you specify at a later step (10).

   Note SecureTransport should not share the same installer directory with another Axway product.

9. Choose the installation type. Press Enter to accept the default, SecureTransport Server.

```
----------------------------------------
Modules
----------------------------------------
Select the modules you want to install, then type Next to continue the
configuration.

Axway SecureTransport V5.5:
1 :[x] Server
2 :[ ] Edge
3 :[ ] ServerDocker
4 :[ ] EdgeDocker


Enter a number[1-4] (Next, Previous, Quit)
:>Next
```

   Note The `EdgeDocker` and `ServerDocker` modules are used **ONLY** in the docker image build process and are not supported for other purposes.

10. Specify the SecureTransport installation directory:

- To accept the default location, press Enter. The default SecureTransport installation folder is `<AxwayHome>/SecureTransport`, where `<AxwayHome>` is the directory you specified in the previous installation directory step.
- To change the SecureTransport installation location, enter 1 and enter the absolute path to your desired installation directory.

  Consider the following:

  - Do not use the directory where you copied the installer files.
  - All SecureTransport Servers in a cluster must use the same installation directory.
  - The name of the SecureTransport installation directory cannot contain space characters, the tab character, or the ~ character. For example, `/root/Axway/STServer` is valid, but `/root/Axway/ST Server` is not.

This installation directory is referred to as `<FILEDRIVEHOME>` throughout this document and other SecureTransport documents.

11. On the database selection prompt, select the external database type for SecureTransport to use.
12. Supply the database settings. Depending on the type of database you are connecting to, the required settings vary.
    - **Host** – The FQDN or IP address for your on-premises database server; the server name of your Azure Database for PostgreSQL server.
    - **Port** – The number of the port used to access the server or cluster. Defaults: 1521 for Oracle, 1433 for Microsoft SQL Server, 5432 for PostgreSQL.
    - **Login Name** – The name of the user authorized to create and populate the SecureTransport schema; in the case of Azure Database for PostgreSQL server, this is the server admin login name.
    - **Password** – The password for the user, not displayed.
    - **Service Name** – Used to connect to the Oracle server or cluster
    - **Database Name** – Used to connect to the Microsoft SQL Server or the PostgreSQL Server
    - **Use secure connection** - When selected, the database connection will be established using SSL. The default value is: **true**.
    - **Server Certificate DN** (Optional) - This is the Server certificate DN value. If provided, the installer will explicitly match the provided value against the certificate provided by the database server.
    - **TrustStore File Path** - PEM or DER file, or JKS (Java Key Store) keystore containing the trusted certificates needed by the installer to establish a chain of trust.

      Note Make sure that all potential certificates for connections between SecureTransport and your database either have no key usage specified or include a Digital Signature key usage extension.

    - **Certificate File** - The public key certificate file. TrustStore files are not supported for PostgreSQL.

- **Use Proxy Authentication** - Set this option to `true` to use the native Oracle proxy authentication feature. You also need to provide the user name of the proxied account.
- **Use Kerberos mode** - This option is supported only for Oracle databases. It enables Kerberos authentication for secure communication between SecureTransport and the Oracle database server. Check the prerequisites.

  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.

  When configuring Kerberos mode:

  - Do not enter **Password** and **Login name**.
  - Specify the location of the **Kerberos credentials cache file**.
  - **Use Kerberos configuration file** - When set to `false`, SecureTransport copies the file to the `<FDH>/conf` directory and synchronizes it between the nodes. The file can be updated via the *Use the Server Configuration Files* page. When the option is set to `true`, SecureTransport references the file directly by specified path. In this case, the file is not synchronized between nodes. You specify it per node. SecureTransport automatically reloads the Kerberos configuration when there's a change in the file.

Note The database connection information must be the same for all SecureTransport Servers in a cluster.

13. The value for **Use existing database schema** depends on whether you are installing the SecureTransport on a stand-alone server or the first server of an Enterprise Cluster, or on another clustered server:
    - If you are installing the first server in a cluster or a stand-alone server, accept the default so that the installer creates the database schema.
    - If you are installing the second or a subsequent server in the cluster, set **Use existing database schema** to `true` to use the database schema created when you installed the first server.

    When you have entered all the settings, press **Enter** to accept the values.

    The installer tests the connection to the database. If the installer cannot connect, it displays an error message and you must correct the database settings.
14. When the installer verifies the database connection, it displays the default SecureTransport ports, nightly log rotation, and import secret file configuration:
15. Accept or modify the default settings.

    If you selected **Use existing database schema** for the second or subsequent server in an Enterprise Cluster, the following three fields are not available.

    - **SSL Admin UI Port** – The port used to connect to the Administration Tool. When you install SecureTransport as a non-root user, the default value for Admin port number is 8444.

- **Tomcat Shutdown Port** – The port used to shut down the Tomcat server

The following field is always available:

- **Enable Nightly Log Rotation** – Select if you want the system to perform automatic backup and purging of log files on a nightly basis. When this feature is enabled, SecureTransport Server backups log files, generated on the respective day, and creates a new one for the subsequent day. The server takes a back up and creates a new log file at 23:59 or 00:00 hours, depending on the log file type. This option is enabled by default. You can enable or disable the nightly log rotation after installation. For more information, see the *SecureTransport Administrator Guide*.
- **Secret File Path** – The path to either the secret (`taeh`) file or the vault `.properties` file, depending on where the secret is stored - internally in SecureTransport or externally in a vault. If you leave it blank, the installer creates a new secret `taeh` file. See *Secret file and vault integration on page 19*.
  For the second and subsequent servers of the cluster, you must copy the respective file from the first cluster node before you install SecureTransport.

When you have modified these values as required for your installation, press **Enter**.

The installer prepares the installation execution and displays its last prompt:

16. Press **Enter** to start the installation.

    The installer displays progress messages as it completes the installation tasks. When the installation is complete, a success message is displayed.

17. Press **Enter** to exit the installer or select update mode to apply patches or service packs without leaving the installer.

The installer also creates a log file, `<AxwayHome>/install.log`.

After successfully installing SecureTransport, you must perform a number of post-installation steps, such as applying your SecureTransport licenses, and enabling, configuring, and starting the SecureTransport services. For more information, see *SecureTransport Getting Started Guide*.

- *Silent installation on page 52*

Run SecureTransport as a service after non-root installation

During a non-root SecureTransport installation, the installer cannot set up and configure SecureTransport to run out of `systemctl`. To allow the services to start during system boot, service files and a target file must be created in the appropriate location, and all files must be linked appropriately. This topic describes the process to configure SecureTransport services to start automatically during system boot.

1. Log on as root.
2. Create a service unit file for each SecureTransport service following the *Service Unit File Example on page 43*.

a. Create `securetransport_db.service` only if using an embedded database.

b. Create `securetransport_admin.service`.

c. Create `securetransport_as2d.service`.

d. Create `securetransport_ftpd.service`.

e. Create `securetransport_httpd.service`.

f. Create `securetransport_pesitd.service`.

g. Create `securetransport_sshd.service`.

h. Create `securetransport_tm.service` only on SecureTransport Servers.

i. Create `securetransport_stop_socks.service` only on SecureTransport Edge Servers.

3. In the service files, add the following lines under the Service section:

```
LimitNOFILE=65536
LimitNPROC=65536
LimitMEMLOCK=1048576
TasksMax=65536
```

When SecureTransport starts as a service at boot time, the limits set in the */etc/security/limits.conf* file are ignored. In this case, the limits specified in the service unit files have the highest priority and override the system defaults in the *system.conf* file. If the latter does not set any limits, then the Linux kernel defaults will be taken.

For details on unit files, see *How to set resource limits over systemd* on page 73.

4. Create a target unit file (`securetransport.target`) for all SecureTransport services following the *Target Unit File Example* on page 44.

5. Move or copy the service files and the target file to the `/etc/systemd/system` directory.

6. In `/etc/systemd/system`, change the permissions of the `securetransport_*.service` files and the `securetransport.target` file to "`644`".

7. Enable the service units (`securetransport_*.service`) by running the following command:

```
# systemctl enable <service_name>
```

8. Enable `securetransport.target` to create the required links and set the SecureTransport services to start on boot.

```
# systemctl enable securetransport.target
```

9. Reload the `systemctl` daemon by running the following command:

```
# systemctl daemon-reload
```

10. Reboot the system and verify that the SecureTransport services start as expected by running the following command:

```
# systemctl --type=service | grep securetransport
```

Tip Check the *FAQ and Troubleshooting* on page 72 section if you're running SecureTransport with SELinux in Enforcing mode and some services fail to start automatically.

Service Unit File Example

```
[Unit]
Description=SecureTransport service unit
# If this is an Enterprise Cluster core server, or the
# securetransport_db service file, remove the lines Requires= and After=
Requires=securetransport_db.service
After=securetransport_db.service
[Service]
Type=forking
# Replace the non-root user with the user specified in the non-root ST
installation
# Replace <FILEDRIVEHOME> with the absolute path to the ST directory
# Replace PIDFile with the corresponding PID file of the service,
# e.g., tomcat.pid
# Replace <service_start_script> with the ST start script for the
# service being defined, e.g., start_admin
# Replace <service_stop_script> with the ST stop script for the
# service being defined, e.g., stop_admin
# Consider adding SuccessExitStatus=SIGKILL to the securetransport_tm.service
file to signify successful termination of the TM service.
User=<non-root user>
PIDFile=<FILEDRIVEHOME>/var/run/admin/tomcat.pid
ExecStart=<FILEDRIVEHOME>/bin/<service_start_script>
ExecStop=<FILEDRIVEHOME>/bin/<service_stop_script>
TimeoutStartSec=200
[Install]
WantedBy=securetransport.target
```

The different locations of the PID files are as follows:

- ADMIN: `<FILEDRIVEHOME>/var/run/admin/tomcat.pid`
- TM : `<FILEDRIVEHOME>/var/run/tm.pid`
- AS2: `<FILEDRIVEHOME>/var/run/as2d.pid`
- DB (if embedded): `<FILEDRIVEHOME>/var/run/db.pid`
- FTP: `<FILEDRIVEHOME>/var/run/ftpd.pid`
- HTTP: `<FILEDRIVEHOME>/var/run/httpd.pid`

- MONITORD: `<FILEDRIVEHOME>/var/run/monitord.pid`
- PESIT: `<FILEDRIVEHOME>/var/run/pesitd.pid`
- SSH: `<FILEDRIVEHOME>/var/run/sshd.pid`

Target Unit File Example

```
[Unit]
Description=Target for all SecureTransport services
After=multi-user.target
[Install]
WantedBy=multi-user.target
```

Install SecureTransport on Windows

This topic explains how to install SecureTransport on Windows. SecureTransport 5.5 uses the Axway Installer in console mode or dialogs for new installations. The pages of the Axway Installer wizard on Windows are the same as the dialogs of the Axway Installer in console mode on UNIX-based platforms.

You cannot install more than one instance of SecureTransport on a Windows server. You can install one instance of either SecureTransport Server or SecureTransport Edge.

The following topics provide how to instructions for installing SecureTransport on Windows:

- *Installing SecureTransport with the embedded database on Windows* on page 44
- *Install SecureTransport Server with an external database on Windows* on page 47

Cancel the installation

You can cancel the installation by clicking **Quit**.The installer opens a confirmation window and proceeds according to your response.

Note If you quit while the installation is in progress, no rollback will be performed and the already installed content must be manually deleted.

Installing SecureTransport with the embedded database on Windows

Before you install SecureTransport, check the pre-installation information and prerequisites for Windows deployments here.

This section provides instructions for installing SecureTransport on Windows in all cases where it uses the embedded database server:

- Stand-alone SecureTransport Server
- SecureTransport Server in a Standard Cluster
- SecureTransport Edge

This procedure assumes that SecureTransport is not installed on the system. Only one instance of SecureTransport Server or SecureTransport Edge is supported in a production environment.

During the installation, do not close any console windows that are opened.

1. Download the SecureTransport installation package for Windows from Axway Support.

   The name of the package is `SecureTransport_5.5_Install_win-x86-64_<BuildNumber>.zip`.

2. Extract the zip file into a directory on the same drive where you are going to install SecureTransport. The name of the extracted folder is `SecureTransport_5.5_Install_win-x86-64`.

3. In the extracted folder, run the `setup64.exe` executable to begin the installation process.

4. The installer loads and displays the *Welcome* page. Click **Next** to proceed.

5. Read and accept the terms of the license agreement to continue.

   (Optional) Click **Print** to print out a copy of the license agreement.

6. Specify the **Installation Directory** to which the installer files to be deployed. It must reside on the same drive as the installation files. You can enter a custom location by using its absolute path.

   Note The directory path must not contain the tilde (~) character. Also, we recommend using a directory path without special characters and spaces.

   This is the location where the installer installs its files, including files required to update and uninstall SecureTransport 5.5. It is also used as a parent directory of the SecureTransport default installation location.

   Click **Next** to continue.

7. Select the module or modules to install, **Server** or **Edge** and click **Next**.

8. Specify a location to install SecureTransport. By default, SecureTransport is installed in a sub-directory of the Axway Installer installation directory (specified at step 6). You can either accept the default or specify a new location following the requirements:

   - The SecureTransport installation directory must be specified using an absolute path. It must not contain the tilde (~) character; letters and digits are acceptable.
   - It must reside on the same drive as the SecureTransport installation files.
   - The SecureTransport installation directory and the Axway Installer components must never be in the same directory.

   SecureTransport Server is installed in the `STServer` sub-directory of the specified installation directory. The SecureTransport installation directory is referred to as `<FILEDRIVEHOME>` throughout the product documentation.

   Click **Next** to continue.

9. If you selected **Edge**, the installer displays another *Database settings* page. Continue with step 11.
10. If you selected **Server**, to install SecureTransport Server in a Standard Cluster or as a stand-alone Server using an embedded database, select **Embedded Database (*MariaDB or MySQL*)** and click **Next**.
11. The following step depends on whether you are using MySQL or MariaDB

    For MySQL, set the port for the embedded database, and click **Next**.

    For MariaDB, provide answers to the following:

    - **Port**: select a new port number for the embedded database installed by SecureTransport, or accept the default setting.
    - **Use secure connection**: When selected (default), the database connection will be established over a secure connection.
        - **Auto generate certificates**: When selected, the certificates required for a secure connection to the database will be automatically generated by SecureTransport with a validity period of 365 days. In this case, any value in **Certificate Authority File Path**, **Server Private key File Path** or **Server Certificate File Path** is ignored.
    - **Certificate Authority File Path**: The absolute file path to the CA certificate in PEM format, used to create and sign the Server Private key and Server Certificate files.
      Only applicable when **Use secure connection** is enabled and **Auto generate certificates** is disabled.
    - **Server Private key File Path**: The absolute file path to the private key in PEM format, used by the MariaDB Server.
      Only applicable when **Use secure connection** is enabled and **Auto generate certificates** is disabled.
    - **Server Certificate File Path**: The absolute file path to the X.509 certificate in PEM format used by the MariaDB Server.
      Only applicable when **Use secure connection** is enabled and **Auto generate certificates** is disabled.
12. Accept or modify the configuration settings.
    - **SSL Admin UI Port** – default 444
    - **Tomcat Shutdown Port** – default 8005
    - **Enable Nightly Log Rotation** – Select if you want the system to perform automatic backup and purging of log files on a nightly basis. When this feature is enabled, SecureTransport Server backups log files, generated on the respective day, and creates a new one for the subsequent day. The server takes a back up and creates a new log file at 23:59 or 00:00 hours, depending on the log file type. This option is enabled by default. You can enable or disable the nightly log rotation after installation - see the *SecureTransport Administrator's Guide* for more information.
    - **Secret File Path** – The path to either the secret (`taeh`) file or the vault `.properties` file, depending on where the secret is stored - internally in SecureTransport or externally in a vault. If you leave it blank, the installer creates a new

secret `taeh` file. See *Secret file and vault integration on page 19*.

For the second and subsequent servers of the cluster, you must copy the respective file from the first cluster node before you install SecureTransport.

Click **Next** to continue.

13. On the *Ready to install* page, click **Install** to start the installation.

    The installation process can take several minutes to complete.

14. When the installation is complete, the installer displays the *Installation completed* page. Click **Next** to see summary information about your SecureTransport installation.

15. Click **Finish** to exit the installer.

    You can click **Update** to install patches or service packs without leaving the installer. Refer to the Readme files for the patches or service packs.

The installer also creates a log file, `<AxwayHome>/install.log`.

After successfully installing SecureTransport, you must perform several post-installation steps, such as updating your SecureTransport license, enabling, configuring, and starting the SecureTransport services. For more information, see *SecureTransport Getting Started Guide*.

- *Requirements for installing on Windows on page 18*
- *Requirements for clustered deployments on page 19*
- *Silent installation on page 52*

Install SecureTransport Server with an external database on Windows

Before you install SecureTransport, check the pre-installation information and prerequisites for Windows deployments here.

This section provides instructions for installing SecureTransport on Windows in all cases where the installation will use an external database server:

- SecureTransport Server in an Enterprise Cluster. See *Requirements for clustered deployments* on page 19.
- Stand-alone SecureTransport Server when an external database server is otherwise required

To use an external database, you need a license for the Enterprise Clustering (EC) option.

Note In Enterprise Cluster deployments, all your SecureTransport Server nodes must have the same version as your first installed Server node. Do not attempt to add Server nodes to your cluster if the SecureTransport version does not match the one on already installed nodes.

Installation procedure

This procedure assumes that SecureTransport is not installed on the system. Only one instance of SecureTransport Server or SecureTransport Edge is supported in a production environment.

During the installation, do not close any console windows that are opened.

1. Download the SecureTransport install package for Windows from Axway Support.

2. Extract the ZIP file to a location on the same drive where you are going to install SecureTransport.
3. In the extracted folder, run the `setup64.exe` executable to begin the installation process.
4. The installer loads and displays the *Welcome* page. Click **Next** to proceed.
5. Read and accept the terms of the license agreement to continue.

   (Optional) Click **Print** to print out a copy of the license agreement.
6. Specify the **Installation Directory** to which the installer files to be deployed. It must reside on the same drive as the installation files. You can enter a custom location by using its absolute path.

   Note The directory path must not contain the tilde (~) character. Also, we recommend using a directory path without special characters and spaces.

   In the directory that you specify in this step, the installer installs its files, including the files required to update and uninstall SecureTransport 5.5. The directory is used as the parent directory of the SecureTransport default installation location.

   Click **Next** to continue.
7. On the *Modules* page, click **Next** to install the default Server module.
8. Specify a location to install SecureTransport. By default, SecureTransport is installed in a sub-directory of the Axway Installer installation directory (specified at step 6). You can either accept the default or specify a new location following the requirements:
   - The SecureTransport installation directory must be specified using an absolute path. It must not contain the tilde (~) character; letters and digits are acceptable.
   - It must reside on the same drive as the SecureTransport installation files.
   - The SecureTransport installation directory and the Axway Installer components must never be in the same directory.

   The installer installs SecureTransport into the `STServer` directory in this installation directory. The SecureTransport installation directory is referred to as `<FILEDRIVEHOME>` throughout this document and other SecureTransport documents.

   Click **Next** to continue.
9. Select an external database: **External Oracle Database**, **External Microsoft SQL Server Database**, or **External PostgreSQL Database**. Click **Next**.
10. Supply the required database connection parameters. The database settings must be the same for all SecureTransport Servers in a cluster.

    Depending on the type of database you are connecting to, the required information might vary.
    - **Host** – The FQDN or IP address for your on-premises database server; the server name of your Azure Database for PostgreSQL server.
    - **Port** – The number of the port used to access the server or cluster. Defaults: 1521 for Oracle, 1433 for Microsoft SQL Server, 5432 for PostgreSQL.
    - **Login 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
- **Database Name** – Used to connect to the Microsoft SQL Server or the PostgreSQL
- **Use secure connection** - When selected, the database connection will be established using SSL. The default value is: **true**.
- **Server Certificate DN** (Optional) - This is the Server certificate DN value. If provided, the installer will explicitly match the provided value against the certificate provided by the database server.
- **TrustStore File Path** - PEM or DER file, or JKS (Java Key Store) keystore containing the trusted certificates needed by the installer to establish a chain of trust.

  Note Make sure that all potential certificates for connections between SecureTransport and your database either have no key usage specified or include a Digital Signature key usage extension.

- **Certificate File** - the public key certificate file.
- **Use Proxy Authentication** - Set this option to `true` to use the native Oracle proxy authentication feature. You also need to provide the user name of the proxied account.
- **Use Kerberos mode** - This option is supported only for Oracle databases. It enables Kerberos authentication for secure communication between SecureTransport and the Oracle database server. Check the prerequisites.

  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.

  When configuring Kerberos mode:

  - Do not enter **Password** and **Login name**.
  - Specify the location of the **Kerberos credentials cache file**.
  - **Use Kerberos configuration file** - When unchecked, SecureTransport copies the file to the `<FDH>/conf` directory and synchronizes it between the nodes. The file can be updated via the *Use the Server Configuration Files* page.
    When checked, SecureTransport references the Kerberos configuration file directly by specified path. In this case, the file is not synchronized between nodes. You specify it per node. SecureTransport automatically reloads the Kerberos configuration when there's a change in the file.

11. Select **Use existing database schema** depending on whether you are installing the SecureTransport on a stand-alone server or the first server of an Enterprise Cluster, or on another clustered server:
    - If you are installing the first server in a cluster or a stand-alone server, do not select **Use existing database schema**. The installer creates the database schema and the `taeh` file.

- If you are installing the second or a subsequent server in the cluster, select **Use existing database schema** to use the database schema created when you installed the first server.

Click **Next**.

The installer tests the connection to the database. If the installer cannot connect, it displays an error message and you must correct the database settings.

12. When the installer verifies the database connection, it displays the *Configurations* page.

    If you selected **Use existing database schema** for the second or subsequent server in an Enterprise Cluster, you cannot changes the values of the following three fields.

    - **SSL Administration Tool Port** – default 444
    - **Tomcat Shutdown Port** – default 8005

    The following field is always available:

    - **Enable Nightly Log Rotation** – Select if you want the system to perform automatic backup and purging of log files on a nightly basis. When this feature is enabled, Server backups log files, generated on the respective day, and creates a new one for the subsequent day. The server takes a back up and creates a new log file at 23:59 or 00:00 hours, depending on the log file type. This option is enabled by default. You can enable or disable the nightly log rotation after installation - see the *SecureTransport Administrator Guide* for more information.
    - **Secret File Path** – The path to either the secret (`taeh`) file or the vault `.properties` file, depending on where the secret is stored - internally in SecureTransport or externally in a vault. If you leave it blank, the installer creates a new secret `taeh` file. See *Secret file and vault integration on page 19*.
      For the second and subsequent servers of the cluster, you must copy the respective file from the first cluster node before you install SecureTransport.
    - **ClusterAuto-Register IP/FQDN** – To automatically register a node in an Enterprise Cluster, specify it by its IP address or FQDN. Otherwise, leave the field empty. You can add a SecureTransport Server to the cluster at a later stage using the Administration Tool. For more information, see Add a server to a cluster.

    Click **Next** to continue.

13. On the *Ready to install* page, click **Install** to start the installation. The installer displays the *Installation in progress* page which shows the progress of the installation.

    The installation process can take several minutes to complete.

14. Once the SecureTransport installation has completed, the installer displays the *Installation completed* page. Click **Next** to see summary information about your SecureTransport installation.

15. Click **Finish** to exit the installer.

    You can click **Update** to install patches and service packs without leaving the installer. Refer to the Readme files for the patches or service packs.

The installer also creates a log file, `<AxwayHome>/install.log`.

After successfully installing SecureTransport, you must perform several post-installation steps, such as updating your SecureTransport license, enabling, configuring, and starting the SecureTransport services. For more information, see *SecureTransport Getting Started Guide*.

Silent installation 4

Silent installation aka "unattended" requires no user interaction during the setup process. You can install SecureTransport Server or Edge in a Windows or Linux environment using a silent installation file. The purpose of using a silent file is to quickly duplicate an installation on multiple machines without running the Installer and entering the same parameters over and over again. The Installer's silent mode takes these values from existing or generated silent files. Before you can use this procedure, you must have the necessary silent files available.

1. Generate the files needed for a silent installation - the Installer silent file and the SecureTransport configuration file.
2. Edit the SecureTransport configuration file to specify the options for your installation.
3. Install SecureTransport on the new environment using the Installer silent file.

Generate the files needed for a silent installation

1. Start the SecureTransport installation in console or GUI mode and perform configuration until just before you click **Install**, and quit the installation.
   This adds two `.properties` files under `<installation_root_directory>\SilentFile\<date_and_time>_install\`:
   - `Install_Axway_Installer_<installer-version>.properties` - the Installer silent file that you use to run a silent installation.
   - `Install_SecureTransport_<st-version>.properties` - the SecureTransport property file that defines the options for installing SecureTransport.
2. Open the `Axway_Installer` properties file and scroll to the end. You will see `IncludeFiles` specifying the product for installation; in the case of SecureTransport 5.5, the following declaration must be present:

```
IncludeFiles.SecureTransport = Install_SecureTransport_V5.5.properties
```

Do not modify anything else in the Axway_Installer properties file except the `InstallDir` (the location of the installer files, used as a parent directory for the SecureTransport installation) and list of `IncludeFiles` (the Axway products to install).

Modify the SecureTransport .properties file

The SecureTransport properties file, `Install_SecureTransport_<st-version>.properties`, is used to feed in the installation parameters, such as the module to install (Edge or Server), the database to use (embedded or external), the type of installation (root or non-root). The file consists of a list of key-value pairs, which are described in the *Configurable properties for silent installation* on page 56 topic. Sensitive information, such as passwords, must be supplied encrypted.

The SecureTransport properties file can be edited via any text editor but if you want to change the value of an encrypted property, you must use the Silent File Editor tool.

Silent File Editor

The Silent File Editor is a tool dedicated to editing your SecureTransport `.properties` file. It is in the directory where the SecureTransport installation files are: `Tools/SilentFileEditor`.

- For Windows, there are two batch files: `SilentFileEditor.bat` and `SilentFileEditorGUI.bat`.
  Before you run any of these files, edit it to replace "`$JAVA_HOME`" with "`%JAVA_HOME%`" or the full path to the supported Java 11 version.
- For Unix, the script file is called `SilentFileEditor.sh`.
  Before you run it, edit it to replace "`$JAVA_HOME`" with the full path to the supported Java 11 version (for example, /opt/java) or set the `JAVA_HOME` variable.

The supported Java version for the current SecureTransport release is listed in the Third-party software and licenses document (requires login).

Usage

To modify a silent file using the command line, run the script file at `<installation directory>\Tools\SilentFileEditor`.

The parameters for the Silent File Editor are:

- The path to the silent file that you want to modify
- Three arguments in this format:
  - The first argument is the name of the property that you want to modify (for example, DB_ADMIN_PASSWORD). Each property name given must exist in the silent file.
  - The second argument is the value that you want to assign to the property given as the first argument.
  - The third argument is `-c` if the value is to be encrypted first and then saved in the silent file, or `-u` if the value does not need to be encrypted. Note that the encryption can only happen at the server on which SecureTransport will be installed.

Here is typical example of Silent File Editor usage:

```
./SilentFileEditor.sh $ST_SILENT_FILE_PATH InstallDir $ST_INSTALLDIR -u
mssqlPort $DB_PORT_NUMBER -u mssqlLoginName $DB_USER -u mssqlPassword $DB_
USER_PASSWORD -c mssqlDatabaseName $DB_NAME -u mssqlHost $DB_HOSTNAME -u
```

Where:

- `$ST_SILENT_FILE_PATH` is the path to silent file
- `$ST_INSTALLDIR` is the path where SecureTransport should be installed
- `$DB_PORT_NUMBER` is the database port number
- `$DB_USER` is the database username
- `$DB_USER_PASSWORD` is the database password in plaintext.
- `$DB_NAME` is the database name
- `$DB_HOSTNAME` is the database hostname

Notes

- Silent File Editor is only setting plain or encrypted values to keys.
- Silent File Editor cannot add new keys, you should add them manually.
- Silent File Editor does not check if the name of the key is valid.

If needed, you can edit this file to specify different options for your installation.

Install SecureTransport in silent mode

Before you start the silent installation on a cluster node, review the information in *Recommendations for Clustered SecureTransport deployments* on page 55.

Follow the instructions to install from a command line using the Installer silent file.

1. Copy the silent files and the installation package that you used for the first installation to the new environment.
2. Unzip the package in an empty folder.
3. Go to the installation package directory.
4. Run the Installer in Silent mode using the following commands:
    - UNIX:

      ```
      # ./setup.sh -s <the absolute path to the Installer silent
      file Install_Axway_Installer_<version>.properties>
      ```

    - Windows:

      ```
      setup64.exe -s <the absolute path to the Installer silent
      file Install_Axway_Installer_<version>.properties>
      ```

Recommendations for Clustered SecureTransport deployments

This subtopic contains instructions and tips to aid you to silently install SecureTransport (Server and Edge) in clustered deployments:

- Standard Cluster
- Enterprise Cluster (EC) with external databases

Caution In a cluster deployment, all your SecureTransport Server nodes must have the same version as your first installed Server node. Do not attempt to add Server nodes to your cluster if the SecureTransport version does not match the one on already installed nodes.

Silent installation on Standard Cluster nodes

With Standard Cluster, after you perform silent installation on your first node, you can re-use the

`Install_SecureTransport_<st-version>.properties` file for silent installation of all other nodes.

Standard Cluster works with an embedded MariaDB database. No additional edits are required. See *MariaDB- and MySQL-specific configurable properties* on page 58

Silent installation on Enterprise Cluster nodes

Enterprise Cluster (EC) deployments offer the use of external databases which adds some additional steps in performing successful silent installation on all nodes. To simplify this process, you can separate the SecureTransport Server deployments apart from the SecureTransport Edge deployments. The big difference is that Server nodes are in an EC deployment (using an external database) while Edge nodes are in Standard Cluster deployment (using an embedded database).

When using external databases (Oracle, MSSQL, or PostgreSQL) you must make sure the correct values are added to the respective properties, as listed in the table.

Related topics:

- *Configurable properties for silent installation* on page 56
- *Sample silent files: SecureTransport Server with MariaDB database* on page 64
- *Sample silent files: SecureTransport Server with Oracle database* on page 68

Configurable properties for silent installation

The following table presents some useful configuration properties you can edit in the `Install_SecureTransport_<st-version>.properties` file to perform silent installation of SecureTransport. Make sure to remove any empty valued properties as leaving these properties blank might adversely affect your silent installation!

Property	Description / Example
`Component.Version` *string*	The SecureTransport version to install example: `5.5`
`Component.InstallerVersion` *string*	The version of the Axway installer example: `4.10.0`
`Component.PreferredJavaVersion` *Integer*	The Java version used for the installation. It must be the same as the one from the installation package, for SecureTransport 5.5 it's 11.
`Server` *boolean*	A flag that specifies if you're installing SecureTransport Server. Must be set to `false` when installing SecureTransport Edge.
`Server.LogicalName` *string*	The name of the SecureTransport installation. It must be unique.
`Edge` *boolean*	A flag that specifies if you're installing SecureTransport Edge. Must be set to `false` when installing SecureTransport Server.
`Edge.LogicalName` *string*	The name of the SecureTransport installation. It must be unique.
`InstallDir` *string*	The directory where SecureTransport to be installed
`userName` *string*	The name of the user authorized to install SecureTransport
isNonRootInstall *boolean*	A flag specifying a root or non-root user installation.

Property	Description / Example
`dbType` *string*	The database to install. Possible values: • `useDBLocal` for the embedded database (MySQL or Mariadb) • `useOracleExternal`, `useMSSQLExternal`, `usePostgreSQLExternal` for an external database Must be set to `useDBLocal` when installing SecureTransport Edge. For more information, see *Database-specific configurable properties* on page 58.
`externalDBUseExistingSchema` *boolean*	A flag that specifies if you're using an external database. Not applicable to SecureTransport Edge installation. **Important!** In a EC deployment using an external database (Oracle, PostgreSQL, or MSSQL), you must configure this value to `false` with the first Server node and to `true` all subsequent Server deployments in your EC.
`sslAdminPort` *Integer*	The default value is 444
`tomcatShutdownPort` Integer	The default value is 8005
`enableLogRotation` *boolean*	A flag that specifies if the system should perform automatic backup and purging of log files. When true, SecureTransport Server backups log files, generated on the respective day, and creates a new one for the subsequent day.
`SecretFilePath`	**Secret File Path** – The path to either the secret (`taeh`) file or the vault `.properties` file, depending on where the secret is stored - internally in SecureTransport or externally in a vault. If you leave it blank, the installer creates a new secret `taeh` file. See *Secret file and vault integration on page 19*. For the second and subsequent servers of the cluster, you must copy the respective file from the first cluster node before you install SecureTransport.

Property	Description / Example
`clusterNodeValue`	The FQDN or IP address of the SecureTransport node to add to the cluster.

Database-specific configurable properties

The following table presents the database properties you must configure, depending on your selected database implementation.

MariaDB- and MySQL-specific configurable properties

Property	Description / Example
`dbType` *enum*	The database to install: `useDBLocal` with embedded databases
`internalDBPort` *integer*	Listener port of your embedded database: `33060` by default
`internalDBPort.Type` *enum*	A flag identifying the embedded listener port type: set to `IPPortOwner`
`internalDBPort.Max` *integer*	Upper threshold of embedded port range example: `65535`
`internalDBPort.Min` *integer*	Lower threshold of embedded port range example: `1024`
The following entries concern MariaDB only	
`internalDBUseSecureConnection` *boolean* MariaDB only	A flag that specifies whether or not the connection to the database is encrypted: `true` or `false`. The default value is `true`.
`internalDBAutoGenerateCertificates` *boolean* MariaDB only	A flag that specifies whether or not SecureTransport automatically generates the required certificates for a secure connection to the embedded MariaDB database. The default value is `true`. When set to `false`, you need to provide the certificates by specifying the following options: `internalDBCaPath`, `internalDBServerKeyPath`, and `internalDBServerCertPath`.

Property	Description / Example
`internalDBCaPath` *string* MariaDB only	The absolute file path to the CA certificate in PEM format, used to create and sign X.509 certificates. Only applicable when `internalDBUseSecureConnection` is set to `true` and `internalDBAutoGenerateCertificates` is set to `false`.
`internalDBServerKeyPath` *string* MariaDB only	The absolute file path to the private key in PEM format, used by the MariaDB Server. Only applicable when `internalDBUseSecureConnection` is set to `true` and `internalDBAutoGenerateCertificates` is set to `false`.
`internalDBServerCertPath` *string* MariaDB only	The absolute file path to the X.509 certificate in PEM format, used by the MariaDB Server. Only applicable when `internalDBUseSecureConnection` is set to `true` and `internalDBAutoGenerateCertificates` is set to `false`.

Oracle-specific configurable properties

Property	Description / Example
`dbType` *enum*	The database to install: `useOracleExternal` with Oracle databases
`oracleHost` *string*	Database hostname or IP address: for example `oracle.localdomain.com`
`oracleHost.Type` *enum*	A flag identifying if you are using HostName or IP address for your database: either `HostName` or `IPaddress`
`oraclePort` *integer*	Listener port of your Oracle database: `1521` by default
`oraclePort.Type` *enum*	A flag identifying the listener port number format Example: `Integer`

Property	Description / Example
`oracleUserName` *string*	The name of the database user account Example: `st_user`
`oraclePassword` *string*	The Oracle database encrypted password.
`oraclePassword.Format` *enum*	The encryption format of your database password: `DefenceV1` . Do not change this value!
`oracleServiceName` *enum*	Service name of Oracle database: `orcl`
`externalDBUseExistingSchema` *boolean*	A flag specifying whether to use existing external DB schema or not. Its value must be the same as the one with the `oracleUseExistingSchema` property: either `true` or `false`
`oracleUseExistingSchema` *boolean*	A flag specifying whether to use existing Oracle DB schema or not: `true` or `false`
`oracleUseSecureConnection` *boolean*	A flag specifying whether to use SSL encryption for connection to the Database: `true` or `false`
`oracleCertificateDN` *string*	Distinguished name as specified in your SSL certificate
`OracleTrustStoreFilePath` *string*	Path to DB public certificate in X.509 format
`OracleTrustStoreFilePath.Type` *enum*	Type of SSL certificate: set this value to `File`
`externalDBUseSecureConnection` *boolean*	A flag that specifies whether to use SSL encrypted communication to your external database. Its value must be the same as the one of the `oracleUseSecureConnection` property: either `true` or `false`
`externalDBCertificateName` *string*	Distinguished name as specified in your SSL certificate. Its value must be the same as the one with the `oracleCertificateDN`
`externalDBTrustStore` *string*	Path to DB public certificate in X.509 format

MSSQL-specific configurable properties

Property	Description / Example
`dbType` *enum*	The database to install: `useMSSQLExternal` with MSSQL databases
`mssqlHost` *enum*	Database hostname or IP address Example: `sqlserver.localdomain.com`
`mssqlHost.Type` *enum*	A flag that specifies if you are using HostName or IP address for your database: either `HostName` or `IPaddress`
`mssqlPort` *integer*	Listener port of your MSSQL database: `1433` by default
`mssqlPort.Type` *enum*	A flag identifying the listener port number format Example: `Integer`
`mssqlLoginName` *string*	The name of the database user account Example: `st_user`
`mssqlPassword` *string*	The MSSQL database encrypted password
`mssqlPassword.Format` *enum*	The encryption format of your database password: `DefenceV1`. Do not change this value!
`mssqlDatabaseName` *string*	Name of MSSQL database: `mssql_db4`
`mssqlUseExistingSchema` *boolean*	A flag specifying whether to use existing MSSQL DB schema or not: `true` or `false`
`externalDBUseExistingSchema` *boolean*	A flag specifying whether to use existing external DB schema or not. Its value must be the same as the one with the `mssqlUseExistingSchema` property: either `true` or `false`
`mssqlUseSecureConnection` *boolean*	A flag specifying whether to use SSL encryption for connection to the database: `true` or `false`
`mssqlCertificateDN` *string*	Distinguished name as specified in your SSL certificate

Property	Description / Example
`MssqlTrustStoreFilePath` *string*	Path to DB public certificate in X.509 format
`MssqlTrustStoreFilePath.Type` *enum*	Type of SSL certificate: set this value to `File`
`externalDBUseSecureConnection` *boolean*	A flag that specifies whether to use SSL encrypted communication to your external database. Its value must be the same as the one of the `mssqlUseSecureConnection` property: either `true` or `false`
`externalDBCertificateName` *string*	Distinguished name as specified in your SSL certificate. Its value must be the same as the one with the `mssqlCertificateDN`
`externalDBTrustStore` *string*	Path to DB public certificate in X.509 format

PostgreSQL-specific configurable properties

Property	Description / Example
`dbType` *enum*	The database to install: `usePostgreSQLExternal` with PostgreSQL databases
`postgreHost` *enum*	Database hostname or IP address Example: `postgresqlserver.localdomain.com`
`postgreHost.Type` *enum*	A flag identifying if you are using HostName or IP address for your database: either `HostName` or `IPaddress`
`postgrePort` *integer*	Listener port of your PostgreSQL database: `5432` by default
`postgrePort.Type` *enum*	A flag identifying the listener port number format Example: `Integer`
`postgreLoginName` *string*	The name of the database user account Example: `st_user`

Property	Description / Example
`postgrePassword` *string*	The PostgreSQL database encrypted password
`postgrePassword.Format` *enum*	The encryption format of your database password: `DefenceV1`. Do not change this value!
`postgreDatabaseName` *string*	Name of PostgreSQL database: `st_db4`
`postgreUseExistingSchema` *boolean*	A flag specifying whether to use existing PostgreSQL DB schema or not : `true` or `false`
`externalDBUseExistingSchema` *boolean*	A flag specifying whether to use existing external DB schema or not. Its value must be the same as the one with the `postgreUseExistingSchema` property: either `true` or `false`
`postgreUseSecureConnection` *boolean*	A flag specifying whether to use SSL encryption for connection to the Database: `true` or `false`
`postgreCertificateDN` *string*	Distinguished name as specified in your SSL certificate
`postgreTrustStoreFilePath` *string*	Path to DB public certificate in X.509 format
`postgreTrustStoreFilePath.Type` *enum*	Type of SSL certificate: set this value to `File`
`postgreCertificateCN` *[string]*	SSL certificate common name
`externalDBUseSecureConnection` *boolean*	A flag that specifies whether to use SSL encrypted communication to your external database. Its value must be the same as the one of the `postgreUseSecureConnection` property: either `true` or `false`
`externalDBCertificateName` *string*	Distinguished name as specified in your SSL certificate. Its value must be the same as the one with the `mssqlCertificateDN`
`externalDBTrustStore` *string*	Path to DB public certificate in X.509 format

Configurable database parameters for Clustered SecureTransport deployments

Standard Cluster works with an embedded MariaDB database. No additional edits are required. All nodes must use the same taeh file.

Enterprise Cluster (EC) deployments offer the use of external databases which adds some additional steps in performing successful silent installation on all nodes. To simplify this process, we can separate the SecureTransport Server deployments apart from the SecureTransport Edge deployments. The big difference is in the fact that Server nodes are in EC deployment (using an external database) while Edge nodes are in Standard Cluster deployment (using an embedded database).

When using external databases (Oracle, MSSQL or PostgreSQL) you must make sure the correct values are added to the respective properties, as listed in the following table:

Database	First node	Every following node

Oracle	`oracleUseExistingSchema = false`
	`externalDBUseExistingSchema = false`
		`oracleUseExistingSchema = true`
		`externalDBUseExistingSchema = true`
		`SecretFilePath = <path_to_taeh_file>`

MSSQL	`mssqlUseExistingSchema = false`
	`externalDBUseExistingSchema = false`
		`mssqlUseExistingSchema = true`
		`externalDBUseExistingSchema = true`
		`SecretFilePath = <path_to_taeh_file>`

PostgreSQL	`postgreUseExistingSchema = false`
	`externalDBUseExistingSchema = false`
		`postgreUseExistingSchema = true`
		`externalDBUseExistingSchema = true`
		`SecretFilePath = <path_to_taeh_file>`

Sample silent files: SecureTransport Server with MariaDB database

This topic contains examples of silent installation files for SecureTransport 5.5 with MariaDB database on Windows:

- Axway Installer configuration file: `Install_Axway_Installer_V4.10.13.properties`
- SecureTransport silent installation configuration file: `Install_SecureTransport_V5.5.properties`

Install_Axway_Installer_V4.10.13.properties

The following example provides an example of the Axway Installer configuration file:

```
Component = Axway_Installer
Component.ComponentType = ComponentPack
Component.SourceDiskNumber = 1
Component.Parent =
Component.Version = 4.10.13
Component.LongName = Axway

CreationDate = 27-10-2009 11:23
CreationDate.Type = Date
CreationDate.Format = dd-MM-yyyy HH:mm

IntegrationDir = Axway_Installer_V4.10.13
IntegrationDir.Type = Directory

LevelOfExpertise = 1
LevelOfExpertise.Level = 3
LevelOfExpertise.Show = true

InstallMode = Standard
DVDLocation =
DocumentationIndexRelativePath = Installer_4.5.x_
InstallationPrerequisitesGuide_allOS_en/index.htm

InstMode = Install
InstMode.Default = Install

AxwaySupportURL = https://support.axway.com/

InstallDir = C:\\Axway\\

InstallationLogicalName = SecureTransport1

AllAxwayComps32 = false
AllAxwayComps64 = true

IncludeFiles =
IncludeFiles.SecureTransport = Install_SecureTransport_V5.5.properties
```

Install_SecureTransport_V5.5.properties

The following example provides an example of the SecureTransport silent installation configuration file:

```
Component = SecureTransport
Component.SupportedOS = linux-x86-64;win-x86-64
Component.SourceDiskNumber = 1
Component.Parent = Axway_Installer
Component.Implementation = Java
Component.Version = 5.5
Component.ConfigureMode = false
Component.RootUser = Indifferent
Component.ComponentType = ComponentPack
Component.FileIdent = ST
Component.PreferredJavaVersion = 11
Component.AllowSpaceInDirectoryName = true
Component.InstallerVersion = 4.10.13
Component.MinorVersion = 20211028
Component.LimitedCluster = NoCluster
Component.LongName = Axway SecureTransport
Component.JavaHome = C:\\Axway\\Java\\win-x86\\jre11_u11.0.12_7_64

CreationDate = 27-10-2021 19:07
CreationDate.Type = Date
CreationDate.Format = dd-MM-yyyy HH:mm

IntegrationDir = SecureTransport_V5.5
IntegrationDir.Type = Directory

mssqlPassword =
mssqlPassword.Format = AES128

oraclePassword =
oraclePassword.Format = AES128

SelectedBitArchitecture = 64

InstMode = Install

SecureTransport = true
SecureTransport.Type = Module
SecureTransport.ModuleType = Installed
SecureTransport.LogicalName = SecureTransport
SecureTransport.ParentName = null
SecureTransport.Title = Axway SecureTransport V5.5

Server = true
```

```
Server.Type = Module
Server.ModuleType = Installed
Server.LogicalName = Server
Server.ParentName = SecureTransport
Server.Title = Server

Edge = false
Edge.Type = Module
Edge.ModuleType = NotInstalled
Edge.LogicalName = Edge
Edge.ParentName = SecureTransport
Edge.Title = Edge

InstallDir = C:\\Axway\\SecureTransport\\

userName = root
isNonRootInstall = false

dbType = useDBLocal

internalDBPort = 33060
internalDBPort.Type = IPPortOwner
internalDBPort.Max = 65535
internalDBPort.Min = 1024

internalDBUseSecureConnection = true
internalDBAutoGenerateCertificates = true
internalDBCaPath =
internalDBServerKeyPath =
internalDBServerCertPath =
externalDBUseExistingSchema = false
externalDBUseSecureConnection = false

sslAdminPort = 444
sslAdminPort.Type = Integer

tomcatShutdownPort = 8005
tomcatShutdownPort.Type = Integer

enableLogRotation = true
SecretFilePath =
```

Sample silent files: SecureTransport Server with Oracle database

This topic shows examples of silent installation files for SecureTransport Server with an external Oracle database in a Linux cluster:

- Axway Installer configuration file: `Install_Axway_Installer_V4.10.13.properties`
- SecureTransport silent installation configuration file: `Install_SecureTransport_V5.5.properties`

Install_Axway_Installer_V4.10.13.properties

The following example provides an example of the Axway Installer configuration file:

```
Component = Axway_Installer
Component.ComponentType = ComponentPack
Component.SourceDiskNumber = 1
Component.Parent =
Component.Version = 4.10.13
Component.LongName = Axway
CreationDate = 27-10-2009 11:23

CreationDate.Type = Date
CreationDate.Format = dd-MM-yyyy HH:mm

IntegrationDir = Axway_Installer_V4.10.13
IntegrationDir.Type = Directory

LevelOfExpertise = 1
LevelOfExpertise.Level = 3
LevelOfExpertise.Show = true

InstallMode = Standard

DVDLocation =
DocumentationIndexRelativePath = Installer_4.5.x_
InstallationPrerequisitesGuide_allOS_en/index.htm

InstMode = Install
InstMode.Default = Install

AxwaySupportURL = https://support.axway.com/

InstallDir = /root/Axway/
```

```
InstallationLogicalName = SecureTransport2

AllAxwayComps32 = false

AllAxwayComps64 = true

IncludeFiles =
IncludeFiles.SecureTransport = Install_SecureTransport_V5.5.properties
```

Install_SecureTransport_V5.5.properties

The following example provides an example of the SecureTransport silent installation configuration file:

```
Component = SecureTransport
Component.SupportedOS = linux-x86-64;win-x86-64
Component.SourceDiskNumber = 1
Component.Parent = Axway_Installer
Component.Implementation = Java
Component.Version = 5.5
Component.ConfigureMode = false
Component.RootUser = Indifferent
Component.ComponentType = ComponentPack
Component.FileIdent = ST
Component.PreferredJavaVersion = 11
Component.AllowSpaceInDirectoryName = true
Component.InstallerVersion = 4.10.13
Component.MinorVersion = 20211125
Component.LimitedCluster = NoCluster
Component.LongName = Axway SecureTransport
Component.JavaHome = /root/Axway/Java/linux-x86/jre11_u11.0.12_7_64

CreationDate = 09-11-2021 11:58
CreationDate.Type = Date
CreationDate.Format = dd-MM-yyyy HH:mm

IntegrationDir = SecureTransport_V5.5
IntegrationDir.Type = Directory

mssqlPassword =
mssqlPassword.Format = AES128

oraclePassword = YourEncryptedPassword
oraclePassword.Format = DefenceV1

SelectedBitArchitecture = 64
```

```
InstMode = Install

SecureTransport = true
SecureTransport.Type = Module
SecureTransport.ModuleType = Installed
SecureTransport.LogicalName = SecureTransport
SecureTransport.ParentName = null
SecureTransport.Title = Axway SecureTransport V5.5

Server = true
Server.Type = Module
Server.ModuleType = Installed
Server.LogicalName = Server
Server.ParentName = SecureTransport
Server.Title = Server

Edge = false
Edge.Type = Module
Edge.ModuleType = NotInstalled
Edge.LogicalName = Edge
Edge.ParentName = SecureTransport
Edge.Title = Edge

ServerDocker = false
ServerDocker.Type = Module
ServerDocker.ModuleType = NotInstalled
ServerDocker.LogicalName = ServerDocker
ServerDocker.ParentName = SecureTransport
ServerDocker.Title = ServerDocker

EdgeDocker = false
EdgeDocker.Type = Module
EdgeDocker.ModuleType = NotInstalled
EdgeDocker.LogicalName = EdgeDocker
EdgeDocker.ParentName = SecureTransport
EdgeDocker.Title = EdgeDocker

InstallDir = /root/Axway/SecureTransport/

userName = root

isNonRootInstall = false

dbType = useOracleExternal

oracleHost = 10.222.2.22
oracleHost.Type = HostName
```

```
oraclePort = 1521
oraclePort.Type = Integer

oracleUserName = gblagovwin

oracleServiceName = oracle19c

oracleUseExistingSchema = true

oracleUseSecureConnection = false

oracleCertificateDN =
OracleTrustStoreFilePath =

oracleKerberosMode = false
oracleUseKerberosFile = false
OracleKrbConfPath =
OracleKrbCachePath =

oracleUseProxy = false

oracleProxyUser =

externalDBTrustStore =

oracleTlsVersion =

externalDBUseExistingSchema = true
externalDBUseSecureConnection = false

sslAdminPort =

tomcatShutdownPort =

enableLogRotation = true

SecretFilePath = /opt/taeh
SecretFilePath.Type = File

clusterNodeValue = 10.164.64.64
```

FAQ and Troubleshooting 4

This section provides answers to frequently asked questions as well as troubleshooting information about installing and starting up SecureTransport.

How to check and set user-level limits

The */etc/security/limits.conf* file defines process resource limits per user. On the user level, SecureTransport has minimum requirements for the soft and hard limits for open files, processes, and locked memory. Here is how you can check the current user level limits and raise them if necessary:

1. Check the current values for soft and hard limits as shown below:

```
#Check user limits for open files
ulimit -aH |grep "open files"
ulimit -aS |grep "open files"
ulimit -a |grep "open files"

#Check user limits for max user processes
ulimit -aH |grep "max user processes"
ulimit -aS |grep "max user processes"
ulimit -a |grep "max user processes"

#Check user limit for max locked memory
ulimit -aH |grep "max locked memory"
ulimit -aS |grep "max locked memory"
ulimit -a |grep "max locked memory"
```

2. To raise a limit, set the new value in the */etc/security/limits.conf* file, being logged with the appropriate user. The example below shows how to set new limits for all users, the root user only, or for a non-root user:

```
#Apply the new limits for all users
* soft nofile 65536
* hard nofile 65536
* soft nproc 65536
* hard nproc 65536
* hard memlock 4194304

#Apply limits for the root user only (root installation)
root soft nofile 65536
root hard nofile 65536
root soft nproc 65536
root hard nproc 65536
root hard memlock 4194304
```

```
#Apply the new limits for the non-root user used to run SecureTransport (non-
root installation)
<ST non-root Linux user> soft nofile 65536
<ST non-root Linux user> hard nofile 65536
<ST non-root Linux user> soft nproc 65536
<ST non-root Linux user> hard nproc 65536
<ST non-root Linux user> hard memlock 4194304
```

3. For the changes to take effect, exit the existing session and open a new one.
4. Verify that the new limits are effective with the *ulimit* command. See *Step 1*.
5. Restart all services with the *stop_all* and *start_all* scripts.
6. Check the SecureTransport processes limits to confirm the new values.

How to check current limits for a SecureTransport process

You can check the current limits at runtime using the following command, where <PID> is the process ID of the currently running SecureTransport process:

```
cat /proc/<PID>/limits
```

This gives you the values of all limits currently in place for the specified process.
If you want to check only a specific limit, you can use the *grep* command, as shown in the example below:

```
#Check the limits for open files, processes, and locked memory for the TM service
cat /proc/`cat /FILEDRIVEHOME/var/run/tm-java.pid`/limits |grep "Max open files"
cat /proc/`cat /FILEDRIVEHOME/var/run/tm-java.pid`/limits |grep "Max processes"
cat /proc/`cat /FILEDRIVEHOME/var/run/tm-java.pid`/limits |grep "Max locked memory"
```

How to set resource limits over systemd

When SecureTransport starts as a service at boot time, the limits set in the */etc/security/limits.conf* file are ignored. In this case, the limits specified in service unit files have the highest priority and override the system defaults configured in the *system.conf* file. If the latter does not set limits, then the Linux kernel defaults will be taken.
The unit files for the SecureTransport services are located in the */etc/systemd/system/* directory, but upon non-root installation, they are not created automatically and you need to add a unit file for each SecureTransport service following the instructions *FAQ and Troubleshooting* on page 72.

This is an example of what the TM service unit file looks like.

```
[Unit]
Description=Start point for SecureTransport service unit

[Service]
Type=forking
PIDFile=/FILEDRIVEHOME/var/run/tm.pid
ExecStart=/FILEDRIVEHOME/bin/start_tm
ExecStop=/FILEDRIVEHOME/bin/stop_tm

[Install]
WantedBy=securetransport.target
```

Set the new limits

A new limit for SecureTransport service can be set in two ways:

- Option 1: To extend the existing service definition file by editing the unit file to add the required parameters with their new values in the [Service] section, as shown below:

```
[Service]
Type=forking
PIDFile=/FILEDRIVEHOME/var/run/tm.pid
ExecStart=/FILEDRIVEHOME/bin/start_tm
ExecStop=/FILEDRIVEHOME/bin/stop_tm

LimitNOFILE=65536
LimitNPROC=65536
LimitMEMLOCK=4194304
TasksMax=65536
```

- Option 2: To create an override file in which you specify the new resource values.
  The following example demonstrates how to raise the limits for the HTTP daemon (`securetransport_httpd` service) by creating an override file. The procedure is applicable for all protocol services as well as the database service.

  1. Under `/etc/systemd/system`, create a directory named `securetransport_httpd.service.d` (unless it already exists).
  2. In that directory, create a file named `override.conf`.
  3. Add the following in the file:

```
[Service]

LimitNOFILE=65536
LimitNPROC=65536
LimitMEMLOCK=4194304
TasksMax=65536
```

Apply the new limits

For the changes to take effect, reload the daemon configuration with the command `systemctl daemon-reload`.
Then, restart the service via systemd:

```
systemctl stop securetransport_tm.service
systemctl start securetransport_tm.service
```

How to fix service failures after installing SecureTransport on systems with SELinux enabled

After installing SecureTransport on a system with SELinux in Enforcing mode, certain services may fail to start automatically. To resolve the issue, run the following commands:

```
chcon -t bin_t ../SecureTransport/bin/start_*
chcon -t bin_t ../SecureTransport/bin/stop_*
chcon -t init_var_run_t ../SecureTransport/var/run/
chcon -t init_var_run_t ../SecureTransport/var/run/admin
```

Uninstallation 5

The following topics describe how to uninstall SecureTransport on all platforms.

- *Uninstall SecureTransport from Linux systems* on page 76 - Provides how to instructions for uninstalling SecureTransport on UNIX-based systems.
- *Uninstall SecureTransport from Windows* on page 78 - Provides how to instructions for uninstalling SecureTransport on Windows-based platforms.

Uninstall SecureTransport from Linux systems

This section explains how to uninstall SecureTransport from the Axway appliance or any of the supported Linux platforms.

The following error messages may occur and be placed in the `uninstall.log` during the uninstall of SecureTransport:

- `<Axway installer folder>/synInstall/scripts/utils.sh: line 743: [: -gt: unary operator expected`
- `<Axway installer folder>/synInstall/scripts/utils.sh: line 746: [: too many arguments`

They are expected and will not cause an uninstall failure.

If you are uninstalling from the Axway appliance, you can use the Appliance Console Menu to proceed. Refer to the *SecureTransport Appliance Guide*.

Note In a cluster environment, stop all of the protocol servers and services on the node you want to uninstall and remove this node from the cluster before you uninstall it.

1. Log in to the system as the user who installed and runs SecureTransport.
2. Use the `<FILEDRIVEHOME>/bin/stop_all` command to stop all SecureTransport services.
3. Navigate to the Axway Installer directory of your installation and run the uninstaller script by typing the following on the command line:

```
./uninstall.sh
```

Note If you want to run the uninstallation procedure in non-interactive mode, you should run `./uninstall.sh -a`.

The Axway Installer initializes and displays a welcome message and a prompt.

```
Initialization in progress .......

----------------------------------------
Welcome
----------------------------------------
Welcome to the Axway Installer wizard for SecureTransport
Server and SecureTransport Edge.
This wizard will install SecureTransport Server or
SecureTransport Edge on your computer.
Next (type Next or N or n): to go to next Dialog
Previous (type Previous or P or p): to go to previous Dialog
Quit (type Quit or Q or q): to abort
If you want to delete a field value, use the Space bar or
the Tab key. During installation, all values or choices must
be validated by pressing Enter.
Enter (Next, Quit).
>Next
```

4. Press Enter to continue.

   The installer prepares the uninstallation execution and displays the following prompt:

```
Please wait while execution process is being prepared!

----------------------------------------
Uninstallation execution
----------------------------------------
All selected products are ready to uninstall. Type Next to start
uninstalling. If not, type Previous to make changes.
Enter (Next, Previous, Quit).
>Next
```

5. Press Enter to continue.

   The installer displays the following confirmation prompt:

```
Uninstall in progress...
Confirmation
Warning: Before proceeding, ensure that the products you want to
uninstall are stopped.
Do you want to continue?
Confirm this operation [y/n]
```

6. Type `y` and press Enter to continue.

The installer displays progress messages as it completes the uninstallation tasks.

When the installer has uninstalled SecureTransport and the Axway Installer, it displays:

```
Uninstallation successful

---------------------------------------
Summary
---------------------------------------
The information below summarizes the uninstallation status. Refer to
install.log for more details.

---------------------------------------------------------------------
Product: SecureTransport_V5.5 Uninstalled from <FILEDRIVEHOME>
---------------------------------------------------------------------
```

7. If you were running SecureTransport as a service, as described in *Run SecureTransport as a service after non-root installation on page 41*:
   - On Oracle Linux and RHEL – Remove `/etc/init.d/rc.stransport`.
   - On SLES – Remove `/etc/rc.d/rc.stransport`.

Uninstall SecureTransport from Windows

This section explains how to uninstall SecureTransport from Windows.

When uninstalling SecureTransport, there could be Axway registry entries left behind. It is safe to remove them following the procedure:

Run `regedit.exe` to start the Microsoft Windows registry and delete the following registry entries:

```
HKEY_LOCAL_MACHINE\SOFTWARE\Axway Software
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\
    Uninstall\Axway_Installer_4.10.6 SecureTransport01
```

Note In a cluster environment, stop all of the protocol servers and services on the node you want to uninstall and remove this node from the cluster before you uninstall it.

1. Prior to uninstallation, stop all SecureTransport processes and make sure that no SecureTransport files or directories are in use and that the Cygwin console and all Cygwin tools and services installed with your previous SecureTransport installation are closed. If necessary, close the Cygwin console and tools manually.
2. Select **Start > Programs > Axway Software > Uninstall**.

   The installer loads and displays the *Welcome* page.
3. Click **Next** to proceed. The installer displays the *Ready to uninstall* page.
4. Click **Uninstall** to start the uninstallation. The installer displays a confirmation dialog.

5. If you have stopped all SecureTransport and related services as described in step 1, click **Yes**. The installer displays the *Uninstall in progress* page which shows the progress of the uninstallation.
6. When uninstallation is complete, the installer displays the *Uninstall completed* page.
7. Click **Next**. The installer displays the *Summary* page.
8. Click **Finish** to close the installer.

Note You can also use the Add/Remove Programs option in the Control Panel to uninstall SecureTransport Server or Edge or navigate to the Axway Installer installation folder and start `uninstall64.exe`.

Installer reference 6

This section provides information about the Axway Installer.

About Axway Installer

The Axway Installer is an installation program for uniform and consistent installation of Axway products. The installer is used to install, configure, update, and uninstall SecureTransport.

Installer modes

The installer has the following installation modes:

- GUI mode is supported on Windows, UNIX and Linux. However, to use on UNIX platforms, the installer requires an X-Window environment. To use an X-Window distributed environment, you must export the DISPLAY environment variable: export DISPLAY=myhost.mydomain:0.0
- Console mode displays a series of prompts requiring user responses or actions.

Installer functions

The installer command files are for invoking installer functions in GUI or console mode.

Before installing, Install is the only available function, invoked with the setup file in the root directory of the installation package.

After installing, the Update, and Uninstall functions are available. The scripts for those functions are in the root installation directory.

Function	Mode	UNIX/Linux	Windows
Install	GUI	`setup.sh`	`setup64.exe`
	Console	`setup.sh -m console`	`setup64 -m console`
Update	GUI	`update.sh`	`update64.exe`
	Console	`update.sh -m console`	`update64 -m console`

Function	Mode	UNIX/Linux	Windows
Uninstall	GUI	`uninstall.sh`	`uninstall64.exe`
	Console	`uninstall.sh -m console`	`uninstall64.exe -m console`

The Configure function enables you to change settings that were applied during installation.

The Update function enables you to apply or remove service packs and patches.

The following sections provide more details on how the installer functions are used with SecureTransport.

- *Install product* on page 84
- *Update product* on page 84
- *Remove updates* on page 86
- *Uninstall product* on page 88

Display command

The `display` command lists information about all installed products. The command is named *display.bat* on Windows and *display.sh* on UNIX and Linux. Run it from the root installation directory.

If you enter the command without parameters, the command lists:

- All products and versions that are installed
- All the service packs that have been applied

Use the `name` parameter to display the installation history of a single product. For example:

```
display -n <product name>
```

Installed directories

The Axway Installer creates the following sub-directories:

- **Configuration**

  Includes the configuration file for each installed product
- **Documentation**

  User documentation
- **Installer**

  Files used by the installer
- **Java**

  The deployed JRE used by the installer and Axway products

- **SilentFile**

  Includes the silent file for each installed product

- **synInstall**

  Installer internal files that are used to manage the installed infrastructure

- **Tools**

  Tools used by the installer to manage infrastructure instances. You can use some of these tools. For example, XDBM and SilentFileEditor.

Hostname

Hostname corresponds to the object assigned to a physical server. In the installer, hostname is required for the following reasons:

- In a page where you configure which network interface the product is going to listen for an incoming connection. In this case, enter one of the following values:
  - Hostname
  - Fully qualified domain name
  - IP address of the machine
  - Specific string (0.0.0.0 or *) indicating that you want the product to listen on all network interfaces if your machine has more than one
- In a page where you configure how your product is going to connect to another product. In this case, it is strongly recommended to use either the fully qualified name or the IP address of the remote machine.

Installer prerequisite checks

This article details the prerequisite checks that Axway Installer performs during the installation of SecureTransport on a Linux machine. It applies to SecureTransport Update 5.5-20230629.

Temporary directory

Previously, the temporary directory was selected and passed to the Java code, silently, without any message in the console/GUI. Information about the process was only recorded in the installation.log file.

Starting with Update 5.5-20230629, there is a new version of the installer that validates and assigns a temporary directory for the installation process. There are three requirements for the temporary directory and all of them must be met to install SecureTransport:

- It must be a valid directory.
- It must have at least 2 GB of free space.

- The user performing the installation must have rwx (read, write, and execute) permission on the temp directory.

You can explicitly specify a temporary directory by passing the argument `-temporary-dir` to the *setup.sh* script. For example, with the following command, the installer will try to use /newtmp as a temporary directory

```
./setup.sh --temporary-dir /newtmp
```

If the specified directory meets all the requirements, it would be set as a temporary directory.

If no temporary directory was passed via the `-temporary-dir` argument or it does not meet the requirements, the installer will use the directory, specified using the TMPDIR environment variable. Again, it is checked for validity, available space, and permissions, and if all requirements are met, it used as a temporary directory for the installation files.

If a temp directory is not specified via `TMPDIR` or the installer argument, or the specified one does not pass the prerequisite checks, then the installer will check the following directories */tmp*, */var/tmp*, */usr/tmp* in this exact order and will use the first one that passes the validation check for usage.

Installation directory

The installer checks the permissions on the SecureTransport installation directory and the available space. You need 2 GB (fixed) of free space and full permissions on the selected folder. If the specified directory does not meet a requirement, the installer will abort the operation and display an appropriate message.

Temporary directory and installation directory on update

During the installation of SecureTransport updates, the installer performs the following health checks:

- Installation directory checks:
    - Permission to the installation location: you need Full access (Read, Write, and Execute)
- Temp directory checks:
    - Available space: the needed space is calculated with a prediction of the size of the update jar file in unzipped form-dynamic requirement. A little buffer zone is also estimated and included in the calculation.
    - Permissions on the directory: the required ones are Read, Write, and Execute.

Considerations

The new refined and estimated requirements for installation sum up to approximately 6 GB of total free space on the Linux machine before downloading the installer:

- 2 GB of free space in the temporary directory
- 2 GB of free space in the installation directory
- 2 GB of free space for the installer

The stated requirements are independent of drives distribution since now users can specify a temporary directory. They are only valid for the installation process. After installation, SecureTransport may need more space especially if it is installed with an embedded database or is connected to a locally hosted database.

Install product

This section describes how to start the installer to install a product.

Start the installer

Prerequisites

- You have downloaded the installation package from Axway Support
- You have uncompressed or unzipped the package

GUI mode

Locate and run the setup file in the root folder of the installation package as follows:

- UNIX/Linux: `setup.sh`
- Windows: `setup64.exe`

Console mode

Locate and run the setup file in the root folder of the installation package as follows:

- UNIX/Linux: `setup.sh -m console`
- Windows: `setup64.exe -m console`

Update product

This section explains how to apply updates using the Axway Installer.

Prerequisites

- Download the product updates from Axway Support to the machine you want to update.
- Stop the servers that you want to update.
- Before you install an Axway Installer update, make sure that all Windows services created by the installer are stopped.
- Before you install a SecureTransport update, make sure that all services are stopped.
- Before starting the update procedure, it is strongly recommended you change the location of the temporary directory by setting the TEMPORARY_DIR environment variable.
- On Windows platforms, the same user that performed the initial installation (or at least the same type of user) must start the update.

Install updates in interactive mode

1. Use the Update function of the installer:
    - UNIX: Go to the installation folder you want to update and run `./update.sh`.
    - Windows: In the Windows Start menu, select **Axway Software > Axway [installation name] > Update**.
    - Using the console: Change to the installation directory you want to update and run `update.exe`.
2. Select your updates. Under Updates Management, specify the following:
    - **Select a directory**: Select the directory containing all the updates you want to install.
    - **Select file**: Select the update file you want to install.

      The file can be a JAR file or a ZIP archive of JAR files.

      The Installer allows the ZIP file format containing the updates to apply to more than one product in the same installation package.
    - **Information**: Click to open the Readme file.
    - Click **Next** to continue.
3. Review the updates you want to install.
4. Click **Update**.
5. On the warning message, click **Yes** to continue.
6. After the updates are installed, click **Next** to view the summary.
7. Click **OK** to exit the installer.

Install updates in non-interactive mode

To install a single update file, run the following command:

- UNIX: `update.sh -i <full_path_to_update_file>`
- Windows: `update.exe -i <full_path_to_update_file>`

This command applies to JAR and ZIP files.

The update progresses without user interaction. When the update process is complete, the summary is displayed and you can check the log files.

View installation and update log files

The installer creates a log file during the installation, `install.log`, located in the installation root directory. For each product, Axway Installer or SecureTransport, there is an `update.log` file that contains the update history. Those update logs can be found in the following locations:

- `<FILEDRIVEHOME>/synInstall/synPatch/` - for SecureTransport
- `/<userHome>/Axway/synInstall/synPatch/` - for the installer

Remove updates

This section describes how SecureTransport updates are stored as backup, and how you can uninstall them in GUI or in console mode based on your operating system.

Update backups

Whenever an instance of SecureTransport is updated, the current version binary and configuration files are automatically backed up to allow you to revert back. Those per-update backups are saved in two locations:

- `<FILEDRIVEHOME>/synInstall/synPatch/` - for product updates
- `/<userHome>/Axway/synInstall/synPatch/` - for the installer

The first backups of the Axway Installer and SecureTransport binary files are created upon updating the system for the first time, and put in directories named `001` inside the above-mentioned locations. For each subsequent update the system creates individual backup directories and names them with sequential numbers – `002`, `003`, and so on. To check which update the folders correspond to, you can view the `update.log` file in the `synPatch` directory.

Here's an example:
If you install SecureTransport 5.5-20210429 and SecureTransport 5.5-20210930 on top of SecureTransport 5.5 GA, the `<FILEDRIVEHOME>/synInstall/synPatch/` folder will have two subfolders, named `001` and `002`, where

- `001` holds the backup files of SecureTransport 5.5 GA
- `002` holds the backup files of SecureTransport 5.5-20210429

  If a subsequent update is installed, subfolder `003`, containing the backup files of SecureTransport 5.5-20210930, would be created.

SecureTransport will continue adding those backup folders as long as you install updates. Currently, there is no maintenance application to clean up the `synPatch` directories, and if you update frequently, there is a chance the space will run out. It is recommended that you delete the oldest backup folders first and always keep the last one or two to make it possible to remove the current update.

To delete old backups safely, remove the directories of the old updates under `synPatch`, for example, `001`. The files in the parent `synPatch` directory should stay intact. Keep only the directories of the updates you may want to roll back to but make sure you keep the latest update directory. If you delete all subfolders inside `synPatch`, SecureTransport would continue to function. However, reverting to an earlier version would not be possible.

How to remove updates

You can uninstall updates in GUI or in console mode based on your operating system.

Remove updates on UNIX/Linux

1. From the installation root directory, enter the command: `./update.sh -m gui`
2. In the Welcome section, click **Enter** to continue.
3. In the Updates Management section, select option *[2] Manage updates, patches and/or service packs installed*.
4. A tree of products is displayed that represents all the installed products, each of them with their latest updates.
5. Select the update you want to uninstall.
6. When you move to the next section Updates Management is displayed again, where you can perform another update action if you need to. If you do not have any more updates to do, move to the next section.
7. Start the update execution.
8. Review the summary and exit.

Remove updates on Windows

You can remove an update via GUI or the console.

Console mode

1. From the installation root directory, enter the command `update64.exe -m gui`
2. Follow the steps 2 to 8 from the procedure above.

GUI mode

1. Use the Update function of the installer:
2. In the Windows **Start** menu, select

   Axway Software > Axway [installation name] > Update
3. On the Updates Management page, select the update you want to uninstall and click **Remove**.
4. Click **Next** to continue.
5. Review the updates you want to uninstall. To remove the update, click **Update**.
6. After the updates have been uninstalled, click **Next** to view the summary. It displays the list of updates that were removed

Uninstall product

This section describes how to uninstall a product.

Windows installations

The same user (or at least the same type of user) who performed the initial installation must start the installer.

Services modification

Some products support an installation in service mode with a user other than the default (local system account).

If the domain field is not shown in the product's service configuration dialog, then it must be introduced in the user name field, using this format:

```
<domain>\<user name>
```

If it is a local user (a user that was created on the local machine) the `<domain>` field can be . or the `<hostname>`.

Example

```
Local user: user1

.\user1

<hostname>\user1

Network user: user2

<domain_name>\user2
```

Use the Uninstall function

Before you begin uninstalling, you must stop the servers you want to uninstall.

You can uninstall products in GUI or console mode. The command to use depends on your operating system.

If products were installed on Windows in service mode, the installer removes the service.

GUI mode

- UNIX/Linux: `uninstall.sh -m gui`
- Windows:In the Windows **Start** menu, select

  Axway Software > Axway [installation name] > Uninstall

Console mode

- UNIX/Linux: `uninstall.sh -m console`
- Windows: `uninstall64.exe -m console`