API Gateway

Version 7.6.2

14 July 2020

Kerberos Integration Guide

This documentation describes the following Axway software:

Axway API Gateway 7.6.2

Contents

Preface

This guide describes how to use the Kerberos authentication protocol in API Gateway using *Kerberos SPNEGO authentication* on page 12.

Who should read this guide

The intended audience for this guide is personnel in charge of the technical integration of a Kerberos solution.

How to use this guide

This guide should be used in conjunction with the other documents in the API Gateway documentation set.

Before configuring API Gateway to work with other applications, you should understand exactly what filters are, and how they are chained together to create a policy. These concepts are documented in detail in the *API Gateway Policy Developer Guide*.

Before you begin configuring Kerberos authentication in API Gateway, you must have a basic understanding of Kerberos authentication.

Review this guide thoroughly. The following is a brief description of the contents of each section:

- *Kerberos authentication* on page 10 – Provides a short introduction on Kerberos authentication, the key concepts, and the authentication flow.
- *Demo setup: API Gateway as both Kerberos client and service* on page 13 – Describes how to configure a Kerberos setup for demonstration purposes.
- *API Gateway as a Kerberos client* on page 22 – Describes how to configure API Gateway as a Kerberos client to mediate authentication between a non-Kerberos client application and a Kerberos back-end service.
- *API Gateway as a Kerberos service* on page 29 – Describes how to configure API Gateway as Kerberos service to mediate authentication between a Kerberos client application and a non-Kerberos back-end service.
- *API Gateway in Kerberos constrained delegation* on page 43 – Describes how to configure API Gateway to perform Kerberos constrained delegation (KCD).
- *API Gateway in unconstrained credentials delegation* on page 54 – Describes how to configure API Gateway for unconstrained credentials delegation.
- *Use KPS to store passwords for Kerberos authentication* on page 64 – Describes how to utilize Key Property Store (KPS) to store Kerberos passwords.

- *Wireshark tracing for Kerberos authentication* on page 67 – Describes how to use Wireshark to trace Kerberos authentication.

The example Kerberos realm name `AXWAY.COM` is specific to the examples in this guide. Replace the example realm name with your own realm name.

Related documentation

The AMPLIFY API Management solution enables you to create, publish, promote, and manage Application Programming Interfaces (APIs) in a secure and scalable environment. For more information, see the *AMPLIFY API Management Getting Started Guide*.

The following reference documents are also available on the Axway Documentation portal at https://docs.axway.com:

- *Supported Platforms*

  Lists the different operating systems, databases, browsers, and thick client platforms supported by each Axway product.

- *Interoperability Matrix*

  Provides product version and interoperability information for Axway products.

Support services

The Axway Global Support team provides worldwide 24 x 7 support for customers with active support agreements.

Email support@axway.com or visit Axway Support at https://support.axway.com.

See "Get help with API Gateway" in the *API Gateway Administrator Guide* for the information that you should be prepared to provide when you contact Axway Support.

Training services

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

Accessibility

Axway strives to create accessible products and documentation for users.

This documentation provides the following accessibility features:

Screen reader support

- Alternative text is provided for images whenever necessary.
- The PDF documents are tagged to provide a logical reading order.

Support for high contrast and accessible use of colors

- The documentation can be used in high-contrast mode.
- There is sufficient contrast between the text and the background color.
- The graphics have the right level of contrast and take into account the way color-blind people perceive colors.

Updates and revisions

This guide includes the following documentation changes.

Changes in version 7.6.2

- Added information on how to configure a delegation in Kerberos service principal. For more information, see *Configure Active Directory* on page 55

Changes in version 7.6.1

No changes.

Changes in version 7.6.0

No changes.

Kerberos authentication 1

Kerberos is an authentication protocol that is used to verify the identity of a user or host. The authentication is based on tickets used as credentials, allowing communication and proving identity in a secure manner even over a non-secure network. For further security, the Kerberos protocol messages are also protected against eavesdropping and replay attacks.

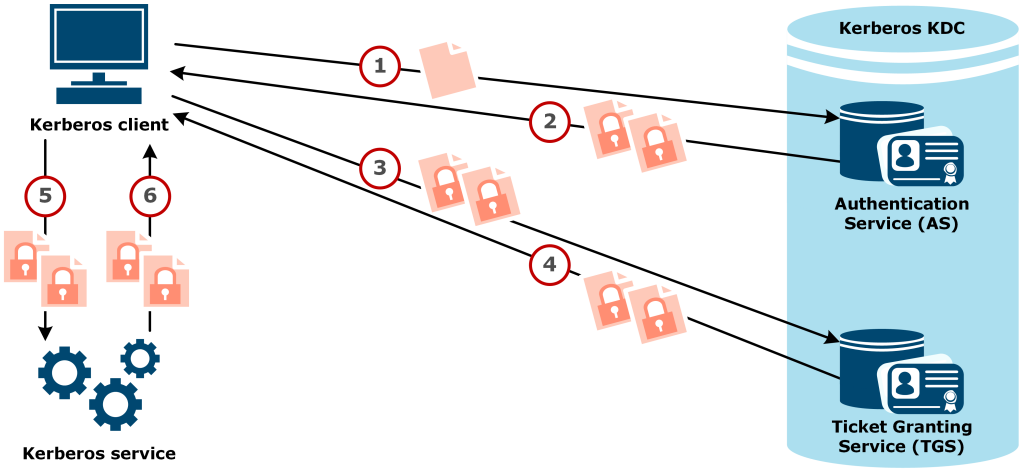
Kerberos authentication is aimed primarily at a client–server model: a Kerberos client sends an authentication request to a Kerberos service. The authentication builds on symmetric-key cryptography, and it requires a trusted third party to facilitate the interactions between two parties. In API Gateway, the authentication is by default mutual: both the user and the server verify each other's identity.

On Windows 2000 and later, Kerberos authentication is the default authentication method when authenticating within an Active Directory domain. Linux operating systems include software for Kerberos authentication of users or services.

Key concepts

- **Kerberos client** – a client (an application or even an end user) requiring access to a Kerberos service.
- **Kerberos service** – a server or an application providing something (a resource, a service, a process, data) a Kerberos client wants to access.
- **Key Distribution Center (KDC)** – a domain service located on a domain controller (such as Active Directory on Windows). In Kerberos, the KDC is a single process providing two services:
    - **Authentication Service (AS)** – authenticates the Kerberos client against the user database, and grants a Ticket Granting Ticket (TGT) for the client.
    - **Ticket Granting Service (TGS)** – validates the client is allowed to access the requested Kerberos service and issues a service ticket for that service. The TGS acts as the trusted third party in the Kerberos protocol.
- **Ticket Granting Ticket (TGT)** – an encrypted identification ticket with a limited validity period used for data traffic protection. The TGT is used to obtain a service ticket from the TGS. The TGT contains the the client/TGS session key, its expiration date, and the client's IP address protecting the client from man-in-the-middle attacks. The TGT is encrypted with the secret key of the TGS.
- **Service ticket** – an encrypted client-to-server ticket containing the client ID, client network address, validity period and client/server session key. A Kerberos client obtains this ticket from TGS after presenting a valid TGT. The service ticket is encrypted with the secret key of the Kerberos service.

Flow description



1. A Kerberos client sends its user ID in a cleartext message to the AS. The message does not include the client's password, nor its secret key based on the password.
2. The AS checks if the client is in the user database, and if found, generates the secret key for the client by hashing the client's password. The AS then sends a client/TGS session key and a TGT to the Kerberos client. The session key is encrypted with the secret key of the client.
3. The Kerberos client decrypts the client/TGS session key, and sends a request message – containing the TGT and the ID of the Kerberos service to be accessed – and an authenticator message – containing the client ID and the timestamp and encrypted with the client/TGS session key – to the TGS.
4. The TGS decrypts the TGT in the request message to retrieve client/TGS session key, and decrypts the authenticator message. The TGS verifies the Kerberos client is authorized to access the Kerberos service requested, and sends a service ticket and a client/server session key encrypted with the client/TGS session key to the Kerberos client.
5. The Kerberos client sends the service ticket and a new authenticator message encrypted with the client/server session key to the Kerberos service to be accessed.
6. The Kerberos service decrypts the service ticket to retrieve the client/server session key, then decrypts the authenticator message to retrieve the client's timestamp. The Kerberos service sends a service confirmation message including the timestamp and encrypted with the client/server session key back to the Kerberos client.
7. The Kerberos client decrypts the service confirmation message and verifies the timestamp is correct. The mutual authentication is now complete. The Kerberos client can now start issuing service requests, and the Kerberos service can provide the requested services for the client.

Kerberos authentication relies on a secure user database storing user IDs and passwords. Using secret keys for encryption requires that the password on the Kerberos client or Kerberos service must match the one stored in the database on the KDC. If the passwords do not match, the secret keys hashed from the passwords do not match either, and decrypting messages fails.

Kerberos SPNEGO authentication

Kerberos authentication based on Simple and Protected Negotiation Protocol (SPNEGO) over HTTP refers to the use of the HTTP negotiate protocol to perform Kerberos authentication at the transport layer between a client and a service. The tokens required for the authentication are transmitted in HTTP headers. The SPNEGO specification suggests using SSL/TLS to provide confidentiality with the authentication mechanism. For more details on SPNEGO, see SPNEGO-based Kerberos and NTLM HTTP Authentication.

Kerberos use cases

This guide provides examples of the following Kerberos scenarios:

- *Demo setup: API Gateway as both Kerberos client and service* on page 13
- *API Gateway as a Kerberos client* on page 22
- *API Gateway as a Kerberos service* on page 29
- *API Gateway in Kerberos constrained delegation* on page 43
- *API Gateway in unconstrained credentials delegation* on page 54

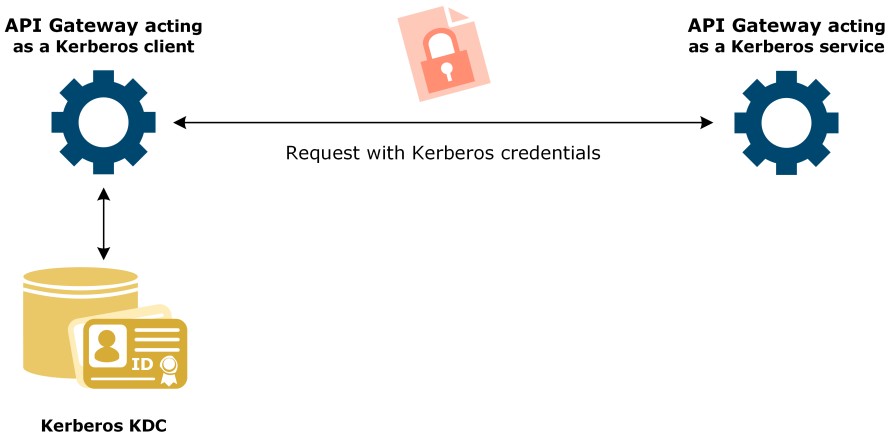
Demo setup: API Gateway as both Kerberos client and service 2

For demonstration purposes, or to test configuring Kerberos authentication, you can configure API Gateway to act both as Kerberos client (`DemoClient`) and Kerberos service (`DemoService`). This configuration is not suitable for production environment.

This is the most straight-forward setup to get started with Kerberos authentication in API Gateway. You configure API Gateway to act as a Kerberos client and authenticate to API Gateway that acts as a Kerberos service.

You can do this configuration using a single API Gateway instance, or two API Gateway instances in different groups. The example in this guide uses a single API Gateway instance.

The Kerberos client and service principals do not use selectors, so the same client principal (`DemoClient@AXWAY.COM`) always authenticates to the same service principal (`DemoService@AXWAY.COM`).



Prerequisites

Before you start configuration, you must have API Gateway installed on any machine with access to the Windows Domain Controller. The machine does *not* have to be a Windows machine that is part of the Windows Domain.

Configuration process

The configuration process has the following steps:

1. *Configure Active Directory* on page 14
2. *Configure Kerberos principals* on page 16
3. *Configure API Gateway to act as the Kerberos client* on page 17
    - *Configure a Kerberos client* on page 17
    - *Configure a Kerberos profile for the Kerberos client* on page 18
    - *Configure a client-side policy* on page 18
    - *Configure Kerberos system settings* on page 19
4. *Configure API Gateway to act as the Kerberos service* on page 19
    - *Configure a Kerberos service* on page 20
    - *Configure a service-side policy* on page 20

Example names

In this example, the Kerberos client `DemoClient@AXWAY.COM` connects to the Kerberos service `DemoService@AXWAY.COM`. You can use the example names, or replace them with names of your own.

The example Kerberos realm name `AXWAY.COM` is specific to the examples in this guide. Replace the example realm name with your own realm name.

Configure Active Directory

This section describes how to configure a Kerberos client principal and Kerberos service principal in Active Directory acting as the Key Distribution Center (KDC). The principals in the KDC are used when configuring the Kerberos principals in Policy Studio. For more details, see *Configure Kerberos principals* on page 16.

- *Configure a user account for the Kerberos client* on page 14
- *Configure a user account for the Kerberos service* on page 15

Configure a user account for the Kerberos client

Configure a user account for the Kerberos client principal. In this example, the client principal is `DemoClient@AXWAY.COM`.

1. On the Windows Domain Controller, click **Control panel > Administrative Tools > Active Directory User and Computers**.

2. Right-click **Users**, and select **New > User**.
3. Enter a name (such as `DemoClient`) in the **First Name** and **User Logon Name** fields, ensure the Active Directory domain is set to your domain, and click **Next**.
4. Enter the password, and do the following:
   - **User must change password at next logon**: Deselect this.
   - **User cannot change password**: Select this.
   - **Password never expires**: Select this.

   This ensures that a working API Gateway configuration does not stop working when a user chooses, or is prompted to change their password. API Gateway does not track these actions.

   If these options are not suitable in your implementation and a user password changes in Active Directory, you must then update the password or keytab of the Kerberos client or service related to the user in Policy Studio, and redeploy the configuration to API Gateway.
   If you cannot deselect **User must change password at next logon**, ensure the user changes the password and that the new password or keytab is deployed to API Gateway *before* API Gateway attempts to connect as this user.

   Tip You can store Kerberos passwords in a KPS table to update a changed password in runtime. For more details, see *Use KPS to store passwords for Kerberos authentication* on page 64.
5. Click **Next > Finish**.

Configure a user account for the Kerberos service

1. Configure a user account for the Kerberos service as in *Configure a user account for the Kerberos client* on page 14. In this example, the name of the service is `DemoService@AXWAY.COM`.
2. Map a Service Principal Name (SPN) to the user account. The Kerberos client uses the SPN to uniquely identify a service. To map the SPN, open a command prompt on the Windows Domain Controller, and enter the following command:

```
> ktpass -princ HTTP/<host>@<Kerberos realm> -mapuser
<user> -pass password -out <user>.keytab -crypto rc4-
hmac-nt -kvno 0
```

The SPN is of the format `HTTP/<host>@<Kerberos realm>`, where `<host>` is the name of the host running the Kerberos service, `DemoService` in this case:

```
> ktpass -princ HTTP/DemoService.axway.com@AXWAY.COM -
mapuser DemoService -pass Axway123 -out
DemoService.keytab -crypto rc4-hmac-nt -kvno 0
```

Substitute the example realm name with your own domain name. Note that the realm name is uppercase.

The command creates an SPN `HTTP/DemoService.axway.com@AXWAY.COM`, which is mapped to the `DemoService` user account. The command also creates a keytab file for the account that you can use later when configuring a Kerberos service for API Gateway in Policy Studio. See *Configure a Kerberos service* on page 20.

Tip If you do not want to create a keytab file, you can use the following command:

```
> setspn -A HTTP/<host> <user>
```

Configure Kerberos principals

This section describes how to add Kerberos principals for the Kerberos client and Kerberos service using Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Principals**.
2. Add a new Kerberos principal for the Kerberos client account (`DemoClient@AXWAY.COM`) as follows:
   - **Name**: `DemoClient`
   - **Principal Name**: `DemoClient@AXWAY.COM`
   - **Principal Type**: `NT_USER_NAME`
3. Add a new Kerberos principal for the service account (`DemoService`) as follows:
   - **Name**: `DemoService`
   - **Principal Name**: `DemoService@AXWAY.COM`
   - **Principal Type**: `NT_USER_NAME`

For more details on the fields and options in this configuration window, see "Configure Kerberos principals" in the *API Gateway Policy Developer Guide*.

For next steps, see *Configure API Gateway to act as the Kerberos client* on page 17

Configure API Gateway to act as the Kerberos client

This section describes how to configure API Gateway to act as a Kerberos client (`DemoClient@AXWAY.COM`) in Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

Configure a Kerberos client

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Clients**.
2. Click **Add a Kerberos Client**, and enter a name for your client (`DemoClient Kerberos Client`).
3. On the **Kerberos Endpoint** tab, set the following:
    - **Load via JAAS Login**: Select this option and the **Request from KDC** option.
    - **Kerberos Principal**: `DemoClient`.
    - **Enter Password**: Enter the password.
    - **Enabled**: Make sure this option is selected.
4. On the **Advanced** tab, set the following:
    - **Mechanism**: `SPNEGO_MECHANISM`.
    - **Context Settings**: Select the following options:
        - **Mutual authentication**
        - **Integrity**
        - **Confidentiality**
        - **Anonymity**
        - **Replay Detection**
        - **Sequence Checking**
    - **Synchronize to Avoid Replay Errors at Service**: Deselect this option to improve performance.

For more details on the fields and options in this configuration window, see "Configure Kerberos clients" in the *API Gateway Policy Developer Guide*.

Configure a Kerberos profile for the Kerberos client

1. In the node tree, click **Environment Configuration > External Connections > Client Credentials > Kerberos**.
2. Add a Kerberos profile as follows:
   - **Profile Name**: `Authenticate to DemoService`.
   - **Kerberos Client**: `DemoClient Kerberos Client`.
   - **Kerberos Service Principal**: `DemoService`.
   - **Send token with first request**: Select this option.

For more details on the fields and options in this configuration window, see "Configure Kerberos client credential profiles" in the *API Gateway Policy Developer Guide*.

Configure a client-side policy

1. Add a new policy named, for example, `Kerberos Demo Client-Side`.
2. Open the **Routing** category in the filter palette, and drag a **Connect to URL** filter onto the policy canvas.
3. Enter the **URL** used to invoke the Kerberos service-side policy in the Kerberos service. In this example, `DemoClient@AXWAY.COM` calls out and back to the same API Gateway instance on `http://localhost:8080/service` to call `DemoService@AXWAY.COM`.
4. On the **Authentication** tab, choose the Kerberos credential profile configured earlier (`Authenticate to DemoService`), and click **Finish**.
   For more details on the fields and options in this configuration window, see "Connect to URL" in the *API Gateway Policy Developer Filter Reference*.
5. Right-click the **Connect to URL** filter, and select **Set as Start**.
6. Click on the **Add Relative Path** icon to create a new relative path `/client` that links to this Kerberos client-side policy.

The policy looks like this:

The client-side policy has the following flow:

- Send a request with a SPNEGO Kerberos token to the Kerberos service on URL `http://localhost:8080/service`.
- Pass the response from Kerberos service back to the calling application.

Configure Kerberos system settings

1. In the node tree, click **Environment Configuration > Server Settings > Security > Kerberos**, and click **Add Kerberos Configuration**.
2. On the **krb5.conf** tab, change the following settings:

```
[libdefaults]
default_realm = AXWAY.COM
[realms]
AXWAY.COM = {
kdc = dc.axway.com
}
```

   Replace the realm settings in the example code with your Kerberos realm, and set the `kdc` setting to the host name of your Windows Domain Controller.

For more details on the fields and options in this configuration window, see "Kerberos configuration" in the *API Gateway Policy Developer Guide*.

Deploy the configuration

To deploy the configuration to API Gateway, click the **Deploy** icon.

You have now configured and deployed a simple client-side policy for Kerberos authentication using SPNEGO. You still need to configure the Kerberos service-side policy that runs when the above policy calls `http://localhost:8080/service`. See *Configure API Gateway to act as the Kerberos service* on page 19.

Configure API Gateway to act as the Kerberos service

This section describes how to configure API Gateway to act as a Kerberos client (`DemoService@AXWAY.COM`) in Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

Configure a Kerberos service

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Services**.
2. Click **Add a Kerberos Service**, and enter a name for your service (`DemoService Kerberos Service`).
3. On the **Kerberos Endpoint** tab, set the following:
   - **Kerberos Principal**:`DemoService@AXWAY.COM`.
   - **Enter Password**: Enter the password you configured for the user account in Active Directory.
   - **Enabled**: Select this option.
4. On the **Advanced** tab, set **Mechanism** to `SPNEGO_MECHANISM`, and click **OK**.

For more details on the fields and options in this configuration window, see "Configure Kerberos services" in the *API Gateway Policy Developer Guide*.

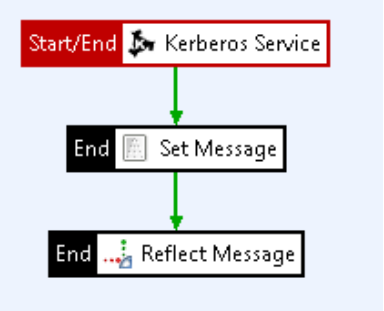
Configure a service-side policy

1. Add a new Policy named, for example, `Kerberos Demo Service-Side`.
2. Open the **Authentication** category in the filter palette, and drag a **Kerberos Service** filter onto the policy canvas.
3. Set **Kerberos Service** to the Kerberos service you created (`DemoService Kerberos Service`), change **Kerberos Standard** to **SPNEGO Over HTTP**, and click **Finish**.
   For more details on the fields and options in this configuration window, see "Kerberos service authentication" in the *API Gateway Policy Developer Filter Reference*.
4. Right-click the **Kerberos Service** filter, and select **Set as Start**.
5. Open the **Conversion** category in the palette, and drag a **Set Message** filter onto the policy canvas.
6. Set **Content type** as `text/xml`, copy the following code to **Message Body**, and click **Finish**:

```
<Response>Kerberos client
'${authentication.subject.id} authenticated'
successfully.</Response>
```

7. Open the **Utility** category in the palette, and drag a **Reflect Message** filter onto the policy canvas.
8. Click **Add Relative Path**, and create a new relative path `/service` that links to this Kerberos service-side policy.

9. Connect the filters with success paths.



The policy has the following flow:

- Authenticate the client.
- Return a response with a HTTP status `200` if the authentication passes.

Deploy the configuration

To deploy the configuration to your Kerberos service, click the **Deploy** icon.

You have now configured a simple service-side policy for SPNEGO authentication. The Kerberos client invokes this policy on `http://localhost:8080/service`.

To test that your Kerberos authentication works as expected, see *Test the policies* on page 21.

Test the policies

Use a third-party client application, such as Postman, to call the client-side policy in API Gateway.

The response should be:

```
<Response>
Kerberos client 'DemoClient@AXWAY.COM authenticated'
successfully.
</Response>
```

The **Traffic Monitor** tab on the API Gateway Manager (`https://localhost:8090`) is an excellent place to view and troubleshoot the message flows. For more details, see "Monitor services in API Gateway Manager" in the *API Gateway Administrator Guide*.

You can now move on to configuring use cases suitable for your implementation. For the list of use cases covered in this guide, see *Kerberos use cases* on page 12.

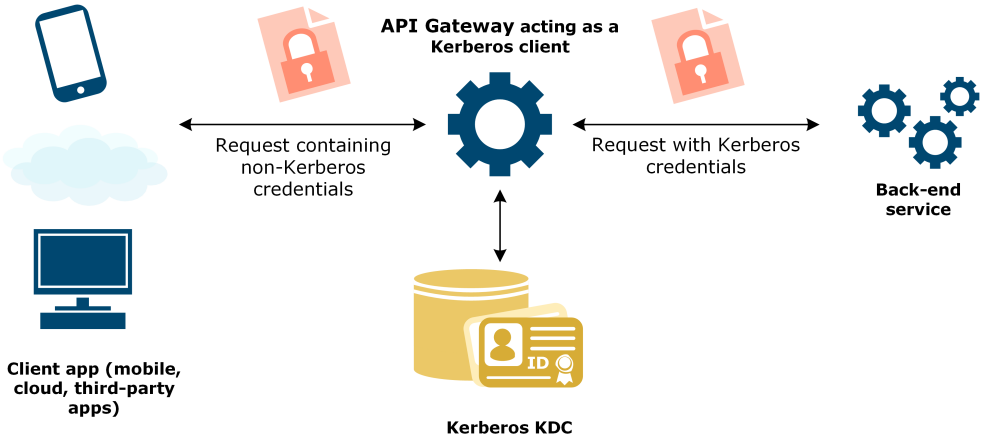
API Gateway as a Kerberos client

3

If a non-Kerberos client application must connect to a back-end service requiring Kerberos authentication, API Gateway can act as a Kerberos client and mediate the authentication.

- **Client application**: Does not support Kerberos authentication.
- **Back-end service**: Requires Kerberos authentication, but not the end user's credential.
- **API Gateway**: Acts as a Kerberos client and authenticates to the back-end service as itself.

The client application can only authenticate using some non-Kerberos authentication mechanism. The back-end service requires Kerberos authentication, but does not need to authenticate the real user associated with the client application. API Gateway sits between the client application and back-end service, and authenticates the client using a non-Kerberos authentication mechanism. The back-end service authenticates API Gateway as the Kerberos client, and trusts that API Gateway has authenticated the real user.



Prerequisites

Before you start configuration, you must have API Gateway installed on any machine with access to the Windows Domain Controller. The machine does *not* have to be a Windows machine that is part of the Windows Domain.

Configuration process

The configuration process has the following steps:

1. *Configure a user account in Active Directory* on page 23
2. *Configure Kerberos principals* on page 24
3. *Configure API Gateway policy* on page 24
    - *Configure a Kerberos client* on page 25
    - *Configure a Kerberos profile for the Kerberos client* on page 26
    - *Configure a Kerberos policy* on page 26
    - *Configure Kerberos system settings* on page 27

Example names

In this example, the Kerberos client `ClientGateway` connects to an existing back-end service. You can use the example names, or replace them with names of your own.

The example Kerberos realm name `AXWAY.COM` is specific to the examples in this guide. Replace the example realm name with your own realm name.

Configure a user account in Active Directory

This section describes how to configure a Kerberos client principal for API Gateway in Active Directory acting as the Key Distribution Centre (KDC).

1. On the Windows Domain Controller, click **Control panel > Administrative Tools > Active Directory User and Computers**.
2. Right-click **Users**, and select **New > User**.
3. Enter a name for the user (such as `ClientGateway`) in the **First Name** and **User Logon Name** fields, ensure the Active Directory domain is set to your domain, and click **Next**.
4. Enter the password, and do the following:

- **User must change password at next logon**: Deselect this.
- **User cannot change password**: Select this.
- **Password never expires**: Select this.

This ensures that a working API Gateway configuration does not stop working when a user chooses, or is prompted to change their password. API Gateway does not track these actions.

If these options are not suitable in your implementation and a user password changes in Active Directory, you must then update the password or keytab of the Kerberos client or service related to the user in Policy Studio, and redeploy the configuration to API Gateway.

If you cannot deselect **User must change password at next logon**, ensure the user changes the password and that the new password or keytab is deployed to API Gateway *before* API Gateway attempts to connect as this user.

Tip You can store Kerberos passwords in a KPS table to update a changed password in runtime. For more details, see *Use KPS to store passwords for Kerberos authentication* on page 64.

5. Click **Next > Finish**.

As a Kerberos client, API Gateway authenticates to an existing back-end Kerberos service. For the authentication to succeed, the back-end Kerberos service must have an account and any SPNs configured in your Active Directory. For an example configuration, see *Configure a user account in Active Directory* on page 30.

For next steps, see *Configure Kerberos principals* on page 24.

Configure Kerberos principals

This section describes how to add Kerberos principals using Policy Studio when API Gateway is the Kerberos client. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Principals**.
2. Add a Kerberos principal for API Gateway acting as the Kerberos client as follows:
   - **Name**: `ClientGateway`
   - **Principal Name**: `ClientGateway@AXWAY.COM`
   - **Principal Type**: `NT_USER_NAME`
3. Add a Kerberos principal for the back-end service as follows:
   - **Name**: Enter the name of the back-end service.
   - **Principal Name**: Enter the Service Principal Name (SPN) for the back-end service (for example, `HTTP/backend.axway.com@AXWAY.COM`).
   - **Principal Type**: `NT_USER_NAME`.

For more details on the fields and options in this configuration window, see "Configure Kerberos principals" in the *API Gateway Policy Developer Guide*.

For next steps, see *Configure API Gateway policy* on page 24.

Configure API Gateway policy

This section describes how to configure API Gateway as the Kerberos client using Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

Configure a Kerberos client

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Clients**.
2. Click **Add a Kerberos Client**, and enter a name for your client (such as `ClientGateway Kerberos Client`).
3. On the **Kerberos Endpoint** tab, set the following:
   - **Load via JAAS Login**: Select this option and the **Request from KDC** option.
   - **Kerberos Principal**: Your API Gateway principal (`ClientGateway`).
   - **Enter Password**: Enter the password for `ClientGateway@AXWAY.COM`.
   - **Enabled**: Make sure this option is selected.
4. On the **Advanced** tab, set the following:
   - **Mechanism**: `SPNEGO_MECHANISM`.
   - **Context Settings**: Select the following options:
     - **Mutual authentication**
     - **Integrity**
     - **Confidentiality**
     - **Anonymity**
     - **Replay Detection**
     - **Sequence Checking**
   - **Synchronize to Avoid Replay Errors at Service**: Deselect this option to improve performance.

For more details on the fields and options in this configuration window, see "Configure Kerberos clients" in the *API Gateway Policy Developer Guide*.

Configure a Kerberos profile for the Kerberos client

1. In the node tree, click **Environment Configuration > External Connections > Client Credentials > Kerberos**.
2. Add a Kerberos profile as follows:
    - **Profile Name**: `Authenticate to back-end service`.
    - **Kerberos Client**: Your Kerberos client (`ClientGateway Kerberos Client`).
    - **Kerberos Service Principal**: Your back-end service Kerberos service principal.
    - **Send token with first request**: Select this option.

For more details on the fields and options in this configuration window, see "Configure Kerberos client credential profiles" in the *API Gateway Policy Developer Guide*.

Configure a Kerberos policy

The following section describes how to configure the Kerberos policy for API Gateway as the Kerberos client.

To start, add a new policy named, for example, `Kerberos Client SPNEGO`.

Configure the end user authentication method

In this example, API Gateway authenticates the end user application using HTTP Basic. Depending on your environment, you may want to use a different authentication mechanism. For more details on the authentication filters available in API Gateway, see Authentication filters" in the *API Gateway Policy Developer Filter Reference*.

1. Open the **Authentication** category, and drag a **HTTP Basic** filter onto the policy canvas.
2. Set **Credential Format** to **User Name**, and select **Allow client challenge**.
3. Set **Repository Name** to `Local User Store`, and click **Finish**.
   For more details on the fields and options in this configuration window, see "HTTP basic authentication" in the *API Gateway Policy Developer Filter Reference*.
4. Right-click the **HTTP Basic** filter, and select **Set as Start**.

API Gateway does not authenticate the end user to the back end. The back-end service only sees API Gateway's Kerberos credentials regardless of how the end user authenticates to API Gateway.

Configure connection to the back-end service

1. Open the **Routing** category in the filter palette, and drag a **Connect to URL** filter onto the policy canvas.
2. Enter the **URL** used that invokes the back-end service.

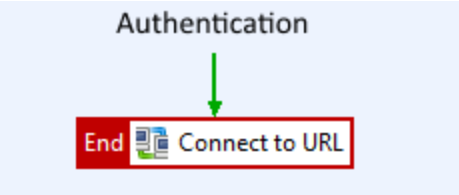
3. On the **Authentication** tab, choose the Kerberos credential profile configured earlier (`Authenticate to back-end service`), and click **Finish**.

If the back-end service requires an SSL/TLS connection, SSL must be configured on the Connect to URL filter. For more details on the fields and options in this configuration window, see "Connect to URL" in the *API Gateway Policy Developer Filter Reference*.

Build the policy

1. Click on the **Add Relative Path** icon to create a new relative path (for example, `/gw-client-to-back-end`) that links to this Kerberos policy.
2. Connect the filters with a success path.

The policy looks like this:



The policy has the following flow:

- Client application authenticates to API Gateway.
- API Gateway sends a request to the back-end service. The request contains a Kerberos SPNEGO token where the client principal is API Gateway.
- The back-end service authenticates API Gateway and returns a response to API Gateway.

Configure Kerberos system settings

1. In the node tree, click **Environment Configuration > Server Settings > Security > Kerberos**, and click **Add Kerberos Configuration**.
2. On the **krb5.conf** tab, change the following settings:

```
[libdefaults]
default_realm = AXWAY.COM
[realms]
AXWAY.COM = {
kdc = dc.axway.com
}
```

Replace the realm settings in the example code with your Kerberos realm, and set the `kdc` setting to the host name of your Windows Domain Controller.

For more details on the fields and options in this configuration window, see "Kerberos configuration" in the *API Gateway Policy Developer Guide*.

Deploy the configuration

To deploy the configuration to API Gateway, click the **Deploy** icon.

You have now configured and deployed a simple Kerberos policy for SPNEGO authentication.

For a list of other use cases covered in this guide, see *Kerberos use cases* on page 12.

Test the configuration

Use a third-party application, such as POSTMan or similar, to call the Kerberos policy in API Gateway.

The back-end Kerberos service should send a confirmation on a successful authentication.

The **Traffic Monitor** tab on the API Gateway Manager (`https://localhost:8090`) is an excellent place to view and troubleshoot the message flows. For more details, see "Monitor services in API Gateway Manager" in the *API Gateway Administrator Guide*.

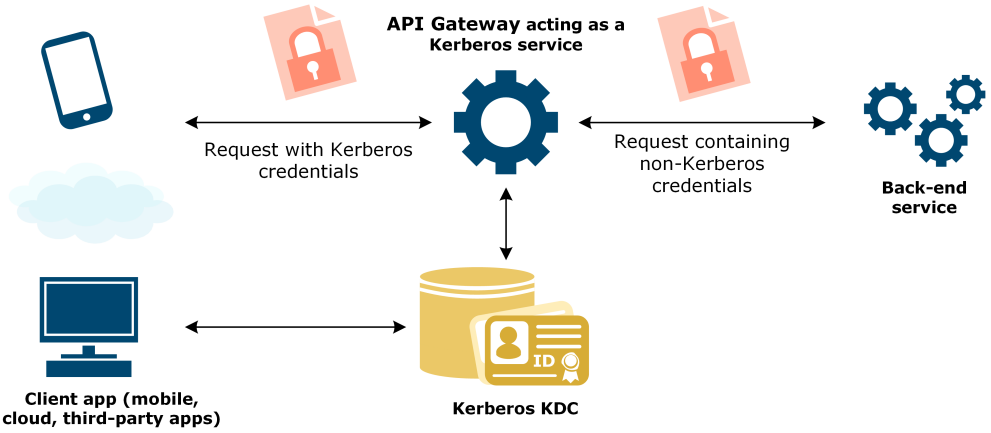
API Gateway as a Kerberos service

4

If the back-end service requires a non-Kerberos authentication, but the client application supports only Kerberos authentication, API Gateway can act as a Kerberos service, and mediate the authentication to the back-end.

- **Client application**: Supports Kerberos authentication.
- **Back-end service**: Requires non-Kerberos authentication (for example, OAuth or SAML).
- **API Gateway**: Acts as a Kerberos Service authenticating the client application, then authenticates to back-end service.

A Kerberos client app, such as a standard browser, authenticates to API Gateway using Kerberos authentication. API Gateway then authenticates to the back-end service using a non-Kerberos authentication mechanism.



Prerequisites

Before you start configuration, you must have API Gateway installed on any machine with access to the Windows Domain Controller. The machine does *not* have to be a Windows machine that is part of the Windows Domain.

Configuration process

The configuration process for API Gateway has the following steps:

1. *Configure a user account in Active Directory* on page 30
2. *Configure Kerberos principal* on page 34
3. *Configure API Gateway policy* on page 34
    - *Configure a Kerberos service* on page 35
    - *Configure a Kerberos policy* on page 35

The most common client application in this scenario is a browser, so this example focuses on that. For instructions on how to configure your browser, see *Configure your browser to authenticate to API Gateway* on page 37

The connection between the client application and API Gateway acting as the Kerberos service is by default unsecured. For security reasons, it is recommended to enable SSL/TLS connection in the Kerberos service. SSL/TLS is configured in the SSL port on the Kerberos service, but you must also configure your browser separately to use SSL/TLS connection. For more details, see *Configure browser authentication over SSL/TLS* on page 38.

Example names

In this example, a client application supporting Kerberos connects to the Kerberos service `ServiceGateway` running on the host machine `gateway.axway.com` connects to an existing back-end service. You can use the example names, or replace them with names of your own.

The example Kerberos realm name `AXWAY.COM` is specific to the examples in this guide. Replace the example realm name with your own realm name.

Configure a user account in Active Directory

This section describes how to configure a Kerberos service principal for API Gateway in Active Directory acting as the Key Distribution Centre (KDC).

- *Configure a user account for API Gateway* on page 31
- *Map an SPN to the user account* on page 31
    - *DNS considerations* on page 32
    - *Map additional SPNs to the user account* on page 33

Configure a user account for API Gateway

1. On the Windows Domain Controller, click **Control panel > Administrative Tools > Active Directory User and Computers**.
2. Right-click **Users**, and select **New > User**.
3. Enter the host name of the machine running API Gateway in the **First Name** and **User Logon Name** fields. For example, if the host name is `gateway.axway.com`, enter `gateway`.
4. Click **Next**.
5. Enter the password, and do the following:
   - **User must change password at next logon**: Deselect this.
   - **User cannot change password**: Select this.
   - **Password never expires**: Select this.

     This ensures that a working API Gateway configuration does not stop working when a user chooses, or is prompted to change their password. API Gateway does not track these actions.

     If these options are not suitable in your implementation and a user password changes in Active Directory, you must then update the password or keytab of the Kerberos client or service related to the user in Policy Studio, and redeploy the configuration to API Gateway.
     If you cannot deselect **User must change password at next logon**, ensure the user changes the password and that the new password or keytab is deployed to API Gateway *before* API Gateway attempts to connect as this user.

Tip You can store Kerberos passwords in a KPS table to update a changed password in runtime. For more details, see *Use KPS to store passwords for Kerberos authentication* on page 64.

6. Click **Next > Finish**.
7. Right-click the new user **gateway**, and select **Properties**.
8. On the **Account** tab, select **Use DES encryption types for this account**.
9. Click **Apply > OK**.

As a Kerberos service, API Gateway authenticates the client application using Kerberos authentication. For the authentication to succeed, the client application must have an account configured in your Active Directory. For an example configuration, see *Configure a user account in Active Directory* on page 23.

Map an SPN to the user account

You must map a Service Principal Name (SPN) to the user account you created (`gateway@AXWAY.COM`).

1. On the Windows Domain Controller, open a command prompt.
2. Enter the following `ktpass` command:

```
> ktpass -princ HTTP/<host>@<Kerberos realm> -mapuser
<user> -pass password -out <user>.keytab -crypto rc4-
hmac-nt -kvno 0
```

The SPN is of the format `HTTP/<host>@<Kerberos realm>`, where `<host>` is the name of the host running the Kerberos service, `gateway.axway.com` in this case:

```
> ktpass -princ HTTP/gateway.axway.com@AXWAY.COM -
mapuser gateway -pass Axway123 -out gateway.keytab -
crypto des-cbc-md5 -kvno 0
```

Replace `gateway.axway.com` with the full host name your browser will use when connecting to API Gateway.
Replace `AXWAY.COM` with your Kerberos realm name. Note that the realm name should be uppercase.

This command creates an SPN `HTTP/gateway.axway.com@AXWAY.COM`, which is mapped to the user account (`gateway`).

The command also creates a keytab file for the account that you can use later when configuring the Kerberos service in Policy Studio. See *Configure Kerberos principal* on page 34.

Tip If you do not want to create a keytab file, you can use the following command:

```
> setspn -A HTTP/<host> <user>
```

If you view the user properties, you see that the user logon name has changed.

DNS considerations

A web browser is the most likely client application authenticating to API Gateway acting as the Kerberos service.

When a browser requests a service ticket from the Kerberos KDC, the browser presents the SPN for the service it wants to connect to. The SPN is based on the host name in the URL entered in the browser.

For example, if the user enters `http://gateway.axway.com:8080/kerberos`, the SPN that the browser passes to the Kerberos KDC to acquire a service ticket is `HTTP/gateway.axway.com@AXWAY.COM`.

If the host name is defined in the DNS as a host (A-name), the SPN is directly resolved from the host:

- The DNS server has the following DNS record defined:

  `HOST(A): gateway.axway.com`

- The following URL is entered in the client browser:

```
URL: http://gateway.axway.com:8080/kerberos
```

- The requested SPN is:

  ```
  HTTP/gateway.axway.com
  ```

If DNS aliases (C-names) are used as host names, the SPN is resolved by mapping the C-name to a DNS A-name:

- The DNS server has the following records defined:

  ```
  HOST(A): gateway.axway.com
  ```

  ```
  Alias (CNAME): test -> gateway.axway.com
  ```

- The following URL is entered in the client browser:

  ```
  URL: http://test.axway.com:8080/kerberos
  ```

- The requested SPN is:

  ```
  HTTP/gateway.axway.com
  ```

If all host names are defined as hosts (A-names) in the DNS, you must map separate SPNs for the hosts to the user account you configured.

If the DNS uses aliases (C-names), it is not necessary to map additional SPNs.

Map additional SPNs to the user account

You can map more than one SPN to a user account, if many browser users need to refer to the API Gateway machine using different host names, for example, `http://gateway.axway.com:8080/kerberos` and `http://test.axway.com:8080/kerberos`.

1. To map additional SPNs to the user account, enter the following command:

   ```
   > setspn -A HTTP/test.com gateway
   ```

   The output looks like this:

   ```
   Registering ServicePrincipalNames for
   CN=gateway,CN=Users,DC=axway,DC=com
   HTTP/test.axway.com
   Updated object
   ```

2. To list the SPNs mapped to a user account, enter the following command:

```
> setspn -L gateway
```

The output looks like this:

```
Registered ServicePrincipalNames for
CN=gateway,CN=Users,DC=axway,DC=com:
HTTP/test.axway.com
HTTP/gateway.axway.com
```

Configure Kerberos principal

This section describes how to add Kerberos principal for API Gateway acting as a Kerberos service using Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

The client application can provide Kerberos token to API Gateway using Kerberos, and the back-end service supports some other authentication mechanism, so you do not need to configure Kerberos principals for them.

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Principals**.
2. Click **Add a Kerberos principal**.
3. Enter a name, such as `ServiceGateway`.
4. In **Principal Name**, enter the SPN you mapped to the user account on Active Directory (`HTTP/gateway.axway.com@AXWAY.COM`).
5. Ensure **Principal Type** is set to `NT_USER_NAME`, and click **OK**.

For more details on the fields and options in this configuration window, see "Configure Kerberos principals" in the *API Gateway Policy Developer Guide*.

For next steps, see *Configure API Gateway policy* on page 34.

Configure API Gateway policy

This section describes how to configure API Gateway as the Kerberos service using Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

Configure a Kerberos service

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Services**, and add a new Kerberos service.
2. Enter a name for the service, such as `ServiceGateway Kerberos Service`, and select the **Kerberos Principal** you configured (`ServiceGateway`).
3. Click **Enter Password**, and enter the password for `gateway@AXWAY.COM`.
4. On the **Advanced** tab, set **Mechanism** to **SPNEGO_MECHANISM**, and click **OK**.

For more details on the fields and options in this configuration window, see "Configure Kerberos services" in the *API Gateway Policy Developer Guide*.

Configure a Kerberos policy

The following section describes how to configure the Kerberos policy for API Gateway as the Kerberos service.

To start, add a new policy named, for example, `Kerberos Service SPNEGO`.

Configure the Kerberos authentication

1. Open the **Authentication** category in the palette, and drag a **Kerberos Service** filter onto the policy canvas.
2. Select the Kerberos service you configured (`ServiceGateway Kerberos Service`), select **SPNEGO Over HTTP**, and click **OK**.
   For more details on the fields and options in this configuration window, see "Kerberos service authentication" in the *API Gateway Policy Developer Filter Reference*.
3. Right-click the **Kerberos Service** filter on the policy canvas, and select **Set as Start**.

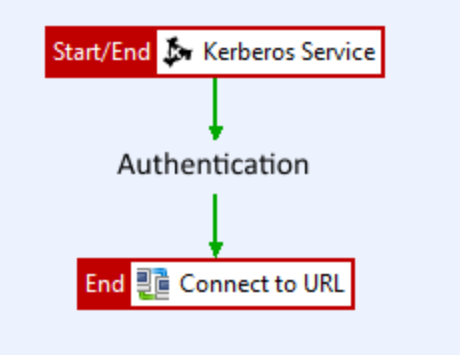
Configure connection to the back-end service

1. Configure the authentication mechanism the back-end service requires. The required filters and configuration details depend on the type of authentication. For more details on different authentication methods, see *API Gateway Policy Developer Guide*.

2. Open the **Routing** category in the filter palette, and drag a **Connect to URL** filter onto the policy canvas.
3. Enter the **URL** used that invokes the back-end service, and click **Finish**.
   For more details on the fields and options in this configuration window, see "Connect to URL" in the *API Gateway Policy Developer Filter Reference*.

Build the policy

1. Click **Add Relative Path** icon, create a new relative path `/gw-service-to-back-end` that links to the Kerberos policy, and click **OK**.
2. Connect the filters with success paths.



The policy has the following flow:

- API Gateway authenticates the client application, such as a browser, using Kerberos authentication.
- API Gateway creates the authentication tokens the back-end service requires.
- API Gateway connects and authenticates to the back-end service.

If API Gateway can map the Kerberos credentials received from the client app to the end-user-specific credentials in the non-Kerberos authentication mechanism, API Gateway can authenticate the actual end user to the back-end service.

Deploy the configuration

To deploy the configuration to API Gateway, click **Deploy** icon.

You have now configured and deployed a simple Kerberos policy for SPNEGO authentication.

The most common client application in this scenario is a standard browser. In addition to configuring API Gateway, you must also configure your browser to authenticate to API Gateway. For more details, see *Configure your browser to authenticate to API Gateway* on page 37.

By default, the connection between the browser and API Gateway acting as the Kerberos service is by default unsecured. For details how to change to a secured connection, see *Configure browser authentication over SSL/TLS* on page 38.

Configure your browser to authenticate to API Gateway

This section describes how to configure your browser to authenticate to API Gateway acting as the Kerberos service.

Configure Internet Explorer

1. Open Internet Explorer on a machine connected to your Kerberos realm (`AXWAY.COM`).
2. Click **Tools > Internet Options**.
3. On the **Advanced** tab, select **Enable Integrated Windows Authentication**, and click **OK**.
4. Restart Internet Explorer, and click **Tools > Internet Options**.
5. On the **Security** tab, click **Local Intranet > Sites > Advanced**.
6. In **Add this website to the zone**, enter your host name (`http://gateway.axway.com`), and click **Add**.You do not have to enter the port or the relative path, and you can use wildcards for the host name (`http://*.axway.com`).

   Note Ensure the host name matches the SPN you mapped to the user account in Active Directory. The SPN must also match the host name the client browser authenticating to API Gateway specifies in the URL. See *Map an SPN to the user account* on page 31.
7. Click **Close**, and on the **Security** tab, click **Local Intranet > Custom Level**.
8. Select **Automatic login only in Intranet zone**, and click **OK**.

Configure Firefox

1. Open Firefox on a machine connected to your Kerberos realm (`AXWAY.COM`).
2. In the address bar, enter `about:config`. If you get a warning prompt on changing the advanced settings, accept it.
3. Right-click the preference name **network.negotiate-auth.trusted-uris**, and select **Modify**.
4. Enter your host name (`http://gateway.axway.com`). If you have multiple entries, separate them with a comma.

   Note Ensure the host name matches the SPN you mapped to the user account in Active Directory. The SPN must also match the host name the client browser authenticating to API Gateway specifies in the URL. See *Map an SPN to the*

user account on page 31.

5. Click **OK**.

You are now ready to test your configuration works as expected. See *Test the configuration* on page 38.

Test the configuration

To test the policy, you need a client app that can get a Kerberos token, such as a standard browser.

1. On a machine connected to your Kerberos realm (`AXWAY.COM`), start the browser you want to test.
2. Enter the URL of API Gateway (`http://gateway.axway.com:8080/gw-service-to-back-end`).

API Gateway authenticates the Kerberos user, passes the request to the back-end service, and passes the response from the back-end service back to the browser.

The **Traffic Monitor** tab on the API Gateway Manager (`https://localhost:8090`) is an excellent place to view and troubleshoot the message flows. For more details, see "Monitor services in API Gateway Manager" in the *API Gateway Administrator Guide*.

For details how to change the configuration to use a secure connection, see *Configure browser authentication over SSL/TLS* on page 38.

Configure browser authentication over SSL/TLS

The connection between the browser and API Gateway acting as the Kerberos service is by default unsecured. For security reasons, it is recommended to use a secure SSL/TLS connection when connecting to the API Gateway.

This section describes the additional configuration steps required to enable a browser to authenticate to the API Gateway using SPNEGO over a secure SSL connection.

The configuration of Kerberos principal, Kerberos service, and Kerberos service policy remains the same as with unsecure connection. For details, see *Configure Kerberos principal* on page 34 and *Configure API Gateway policy* on page 34.

To enable SSL connection, you must do the following:

1. *Configure API Gateway for SSL/TLS connection* on page 39
    - *Import an SSL certificate* on page 39
    - *Configure an HTTPS interface* on page 39
2. *Configure your browser to use SSL/TLS connection* on page 40

Configure API Gateway for SSL/TLS connection

This section describes how to import an existing SSL certificate for API Gateway and configure an HTTPS interface in Policy Studio. For more information on Policy Studio, see the *API Gateway Policy Developer Guide* .

Import an SSL certificate

To enable SSL/TLS connection, you must have a valid SSL certificate.

1. In the node tree, click **Environment Configuration > Certificates and Keys > Certificates**.
2. Click **Create/Import**.
3. Click **Import Certificate...**, and select the certificate file you want to use.
4. Enter an alias for the certificate or click **Use Subject**, and click **OK**.

For more details on the fields and options in this configuration window, see "Manage X.509 certificates and keys" in the *API Gateway Policy Developer Guide*.

Configure an HTTPS interface

1. In the node tree, click **Environment Configuration > Listeners > API Gateway > Default Services > Ports**.
2. Click **Add > HTTPS Interface**, and enter a name for the interface.
3. On the **Network** tab, set **Port** to `8081`.
4. Click **X.509 Certificate**, select your SSL certificate, and click **OK**.

For more details on the fields and options in this configuration window, see "Configure HTTP services" in the *API Gateway Policy Developer Guide*.

Deploy the configuration

To deploy the configuration to API Gateway, click **Deploy** icon.

You must configure your browser to use secure connection as well. For more details, see *Configure your browser to use SSL/TLS connection* on page 40.

Configure your browser to use SSL/TLS connection

This section describes how to configure your browser to authenticate with SPNEGO over SSL/TLS connection.

- *Configure Internet Explorer to use SSL/TLS connection* on page 40
- *Configure Firefox to use SSL/TLS connection* on page 40

For details how to test your configuration, see *Test the SSL/TLS configuration* on page 41.

Configure Internet Explorer to use SSL/TLS connection

1. Open Internet Explorer on a machine connected to your Kerberos realm (`AXWAY.COM`).
2. Click **Tools > Internet Options**.
3. On the **Security** tab, click **Local Intranet > Sites > Advanced**.
4. In **Add this website to the zone**, enter your host name (`https://gateway.axway.com`), and click **Add**. You do not have to enter the port or the relative path, and you can use wildcards for the host name (`https://*.axway.com`).

   Note Ensure the host name matches the SPN you mapped to the user account in Active Directory. The SPN must also match the host name the client browser authenticating to API Gateway specifies in the URL. See *Map an SPN to the user account* on page 31.

5. Click **Close**.

Configure Firefox to use SSL/TLS connection

1. Open Firefox on a machine connected to your Kerberos realm (`AXWAY.COM`).
2. In the address bar, enter `about:config`. If you get a warning prompt on changing the advanced settings, accept it.
3. Right-click the preference name **network.negotiate-auth.trusted-uris**, and select **Modify**.
4. Enter your host name (`https://gateway.axway.com`). If you have multiple entries, separate them with a comma.

   Note Ensure the host name matches the SPN you mapped to the user account in Active Directory. The SPN must also match the host name the client browser authenticating to API Gateway specifies in the URL. See *Map an SPN to the user account* on page 31.

5. Click **OK**.

Test the SSL/TLS configuration

This section describes how to test the configuration for the browser SPNEGO authentication over SSL.

1. On a machine connected to your Kerberos realm (`AXWAY.COM`), start the browser you want to test.
2. Enter the URL of API Gateway (`https://gateway.axway.com:8081/gw-service-to-back-end`).

When you start a new session, the browser might show some security alerts. Once the security alerts have been handled, the back-end Kerberos service should send a confirmation on a successful authentication. For more information, see *Internet Explorer security alerts* on page 41 and *Firefox security alerts* on page 41.

Once the security alerts have been handled, API Gateway sends a response after authenticating the Kerberos user and passing the request to the back-end service.

The **Traffic Monitor** tab on the API Gateway Manager (`https://localhost:8090`) is an excellent place to view and troubleshoot the message flows. For more details, see "Monitor services in API Gateway Manager" in the *API Gateway Administrator Guide*.

For list of other use cases covered in this guide, see *Kerberos use cases* on page 12.

Internet Explorer security alerts

When you start a new HTTPS session to API Gateway, Internet Explorer might show a security alert on the security certificate.

If you want to view the certificate details, click **View Certificate**. To proceed, click **Yes**.

To avoid the security alerts, you can import API Gateway's SSL certificate to the trusted certificate store in the browser:

1. In Internet Explorer, click **Tools > Internet Options**.
2. On the **Content** tab, click the **Certificates...**.
3. Import the certificate, and close the dialog window.

Firefox security alerts

When you start a new HTTPS session to the API Gateway, Firefox may show a security alert for an untrusted connection.

If the security alert is displayed, follow these steps:

1. Click **I Understand the Risks**.
2. Click **Add Exception > Get Certificate**, and import the certificate.

3. Click **Close**.
4. Click **Confirm Security Exception**.

To avoid the security alerts, you can import API Gateway's SSL certificate to the trusted certificate store in the browser:

1. Open Firefox, and select **Tools > Options > Advanced**.
2. On the **Certificates** tab, click **View Certificates**.
3. Import the SSL certificate, and close the dialog window.

API Gateway in Kerberos constrained delegation

5

Kerberos *constrained* delegation (KCD) enables API Gateway to act as a trusted Kerberos service principal, acquire a Kerberos service ticket in the name of the requesting end user, and authenticate to a constrained set of Kerberos back-end services as the end user.

- **Client application**: Does not support Kerberos authentication, or cannot provide Kerberos credentials.
- **Back-end service**: Requires Kerberos authentication with end user's credentials. Multiple back-end services may exist.
- **API Gateway**: Authenticates the client application, then acts as a Kerberos client and authenticates to the back-end service as the end user.

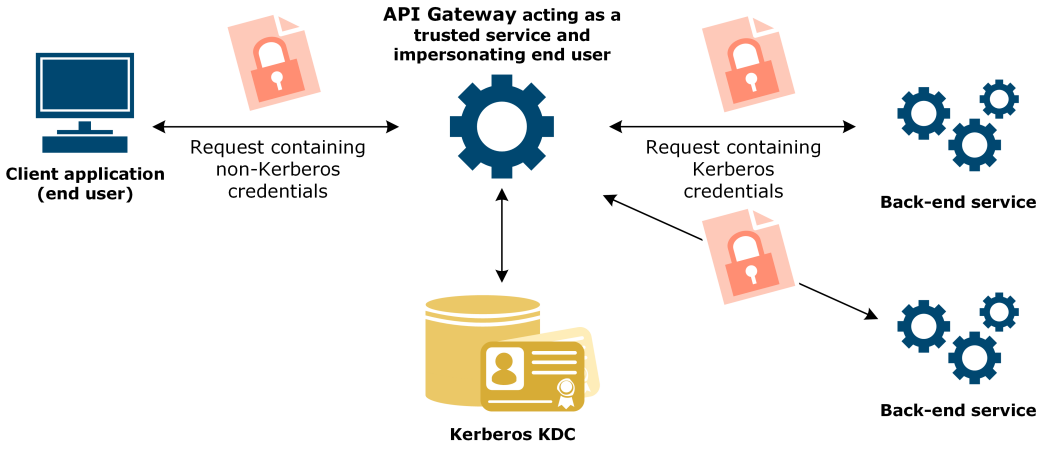
The client application can only authenticate using a non-Kerberos authentication mechanism. The back-end service requires Kerberos authentication, and must authenticate the real user associated with the client application.

API Gateway authenticates the client using a non-Kerberos authentication mechanism. API Gateway has no access to the end user's Kerberos secret key or keytab. API Gateway must map the incoming user credential to the Kerberos client principal name of the user to be impersonated. To do this, API Gateway uses a selector to generate the end user's Kerberos principal name.

As a trusted proxy, API Gateway impersonates the end user's credentials and authenticates to the back-end service as the end user. The back-end service sees the request originating directly from the original end user and can authenticate the end user with Kerberos credentials. API Gateway can request Kerberos service tickets on behalf of the client to more than one Kerberos service and authenticate to multiple back-end services as the end user in a single policy.

Even when using a client application that supports Kerberos authentication, an end user may not be able to provide the Kerberos credentials, for example, when working remotely on a browser and trying to access the back-end service from outside of the Kerberos realm. Configuring API Gateway for KCD helps solve this authentication problem as well.

Note Cross-domain authentication is not supported for Kerberos constrained delegation. However, you can configure a chain of policies with a separate Kerberos service account for each domain to overcome this.



KCD uses two Microsoft Kerberos extensions, Protocol Transition and Constrained Delegation:

- **Protocol Transition (S4U2Self)** – A Kerberos service can obtain a Kerberos service ticket to itself on behalf of a Kerberos principal (the end user) without requiring the end user to initially authenticate using Kerberos. The end user can authenticate using some other authentication mechanism.
- **Constrained Delegation (S4U2Proxy)** – A Kerberos service can request and obtain further Kerberos service tickets to other services on behalf of an end user after receiving the first Kerberos service ticket for that end user. The further Kerberos service tickets can only be requested to a constrained set of services configured in the KDC.

In API Gateway, Protocol Transition and Constrained Delegation must be used in combination. Constrained Delegation is not possible using a ticket obtained by API Gateway when authenticating the client using Kerberos. An API Gateway policy can enforce the authentication of the client to API Gateway to use Kerberos authentication. However, API Gateway does not support forcing this within Active Directory. A policy that forces the incoming authentication to use Kerberos authentication still does both Protocol Transition and Constrained Delegation.

Note Kerberos Constrained Delegation is not supported out-of-the-box in API Gateway v7.5.1 or earlier.

In addition to constrained delegation, API Gateway also supports *unconstrained* or *open* credentials delegation. Constrained delegation is considered to be more secure than unconstrained delegation because the KDC administrator can constrain the set of back-end services that the trusted Kerberos service can request tickets for as the end user they are impersonating. Restricting the delegation reduces the number of potential targets for attacks, so that if one part of the system is compromised, the whole system is not. In unconstrained delegation, the Kerberos service ticket can be requested for any valid service. For more details, see *API Gateway in unconstrained credentials delegation* on page 54.

Prerequisites

Before you start configuration, you must have API Gateway installed on any machine with access to the Windows Domain Controller. The machine does *not* have to be a Windows machine that is part of the Windows Domain.

Configuration process

The configuration process has the following steps:

1. *Configure Active Directory* on page 45
2. *Configure Kerberos principals* on page 47
3. *Configure API Gateway policy* on page 48
    - *Configure the Kerberos client* on page 48
    - *Configure a Kerberos profile for the Kerberos client* on page 49
    - *Configure a Kerberos policy* on page 50
    - *Configure the Kerberos system settings* on page 51

Example names

In this example, the trusted Kerberos principal `TrustedAPIGateway` can impersonate valid users in Active Directory and request service tickets in their name to the back-end service principals `HTTP/BackEndService.axway.com@AXWAY.COM` and `HOST/BackEndService.axway.com@AXWAY.COM`. You can use the example names, or replace them with names of your own.

The example Kerberos realm name `AXWAY.COM` is specific to the examples in this guide. Replace the example realm name with your own realm name.

Configure Active Directory

This section describes how to configure a trusted Kerberos client principal for API Gateway in Active Directory acting as the Key Distribution Centre (KDC) .

Before you configure the user account for the trusted Kerberos principal, you must have configured the user account and Service Principal Names (SPN) for the back-end services you want API Gateway to request service tickets for. For an example configuration, see *Configure a user account for the Kerberos service* on page 15.

Configure user account for the trusted Kerberos principal

1. On the Windows Domain Controller, click **Control panel > Administrative Tools > Active Directory User and Computers**
2. Right-click **Users**, and select **New > User**.
3. Enter a name for the Kerberos principal (`TrustedAPIGateway`) in the **First Name** and **User Logon Name** fields, select your Active Directory domain from the drop-down menu (`@axway.com`), and click **Next**.
4. Enter the password, and do the following:
   - **User must change password at next logon**: Deselect this.
   - **User cannot change password**: Select this.
   - **Password never expires**: Select this.

     This ensures that a working API Gateway configuration does not stop working when a user chooses, or is prompted to change their password. API Gateway does not track these actions.

     If these options are not suitable in your implementation and a user password changes in Active Directory, you must then update the password or keytab of the Kerberos client or service related to the user in Policy Studio, and redeploy the configuration to API Gateway.
     If you cannot deselect **User must change password at next logon**, ensure the user changes the password and that the new password or keytab is deployed to API Gateway *before* API Gateway attempts to connect as this user.

Tip You can store Kerberos passwords in a KPS table to update a changed password in runtime. For more details, see *Use KPS to store passwords for Kerberos authentication* on page 64.

5. Click **Next > Finish**.
6. Open a command prompt on the Windows Domain Controller, and enter the following command to set the Service Principal Name (SPN):

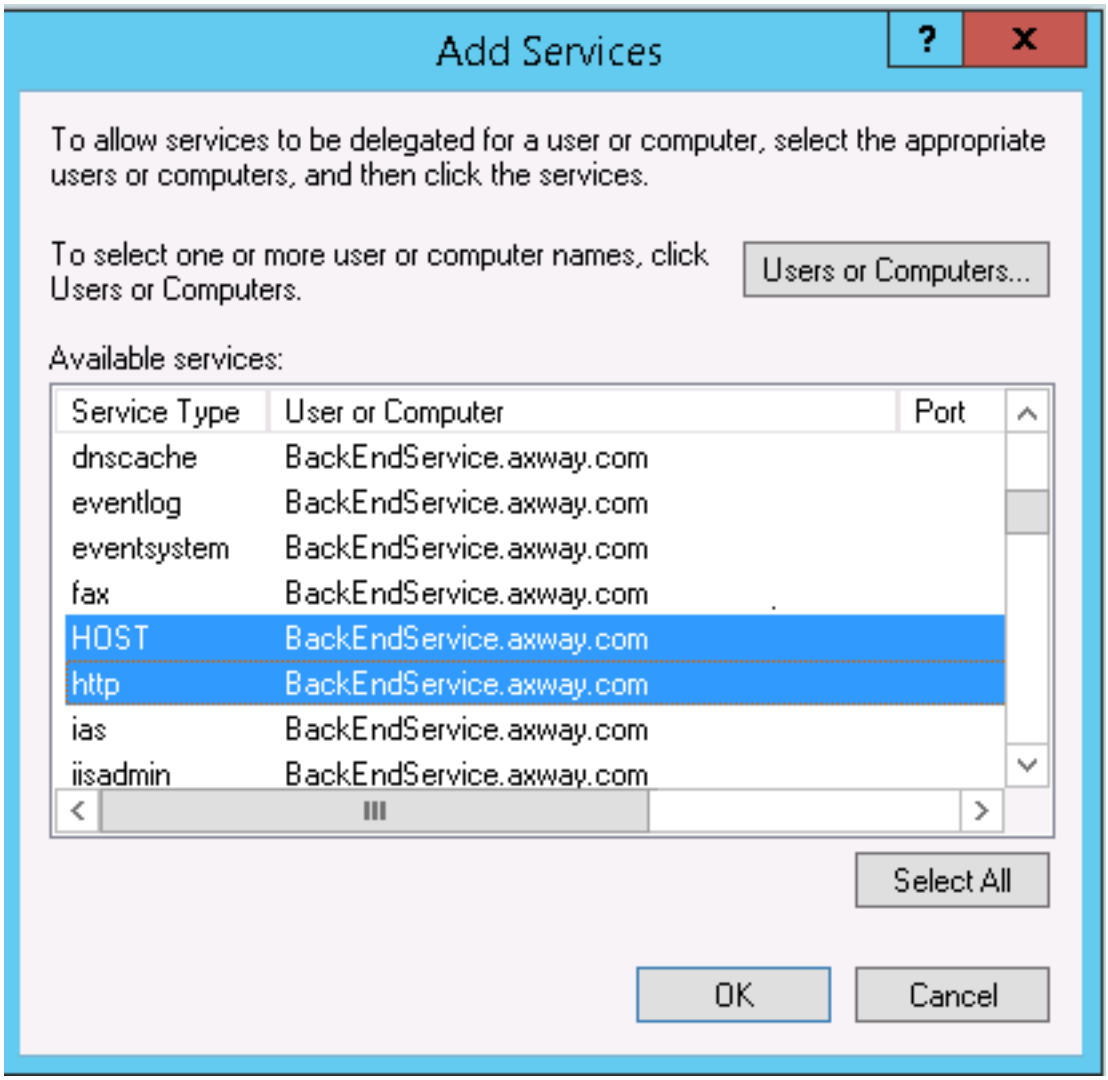
   ```
   > setspn -A <service class>/<host> <service name>
   ```

   For example:

   ```
   > setspn -A HTTP/TrustedAPIGateway.axway.com
   TrustedAPIGateway
   ```

   This command creates the SPN, but does not create a keytab file.
7. Right-click on the new user, and select **Properties > Delegation**, and select **Trust this user for delegation to specified services only** and **Use any authentication protocol**. API Gateway does not support the option **Use Kerberos only**.

8. Add the back-end services (here `HTTP/BackEndService.axway.com` and `HOST/BackEndService.axway.com`), then click **OK**. The trusted Kerberos principal can request service tickets for these back-end services on behalf of the impersonated end users.



For the next steps, see *Configure Kerberos principals* on page 47.

Configure Kerberos principals

This section describes how to add Kerberos principals for the end user, trusted Kerberos principal, and back-end Kerberos service for KCD using Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Principals**.
2. Add a new Kerberos principal for the end user:

- **Name**: `End User Principal to Impersonate in KCD`
- **Principal Name**: `${authentication.subject.id}@AXWAY.COM`
- **Principal Type**: `NT_USER_NAME`

Using a selector here enables you to impersonate multiple end users.

3. Add a new Kerberos principal for the trusted Kerberos principal account as follows:
   - **Name**: `TrustedAPIGateway for KCD`
   - **Principal Name**: `TrustedAPIGateway@AXWAY.COM`
   - **Principal Type**: `NT_USER_NAME`
4. Add a new Kerberos principal for the back-end service account as follows:
   - **Name**: `<Back-end service name>` (for example, `Back-end Kerberos Service`)
   - **Principal Name**: `<Service Principal Name for the back-end service>` (for example, `HOST/BackEndService.axway.com@AXWAY.COM`)
   - **Principal Type**: `NT_USER_NAME`

For more details on the fields and options in this configuration window, see "Configure Kerberos principals" in the *API Gateway Policy Developer Guide*.

For the next steps, see *Configure API Gateway policy* on page 48.

Configure API Gateway policy

This section describes how to configure API Gateway for the KCD using Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

Configure the Kerberos client

Although the trusted Kerberos principal can be referred to as a Kerberos service principal, it is acting on the client-side of the Kerberos authentication transaction, and needs a Kerberos client for KCD.

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Clients**.
2. Click **Add a Kerberos Client**, and enter a name for your client (`Trusted Kerberos Client for KCD`).
3. On the **Kerberos Endpoint** tab, set the following:
    - **Load via JAAS Login**: Select this option and the **Request from KDC** option.
    - **Kerberos Principal**: `TrustedAPIGateway for KCD`.
    - **Enter Password**: Enter the password for `TrustedAPIGateway@AXWAY.COM`.
    - **Enabled**: Ensure this option is selected.
4. On the **Kerberos Constrained Delegation** tab, set the following:
    - **Kerberos Principal to Impersonate**: `End User Principal to Impersonate in KCD`
    - **Select Cache for Impersonated Subjects**: `Kerberos Constrained Delegation Impersonated Subject Cache`
5. On the **Advanced** tab, set the following:
    - **Mechanism**: `SPNEGO_MECHANISM`.
    - **Context Settings**: Select the following options:
        - **Mutual authentication**
        - **Integrity**
        - **Confidentiality**
        - **Anonymity**
        - **Replay Detection**
        - **Sequence Checking**
    - **Synchronize to Avoid Replay Errors at Service**: Deselect this option to improve performance.
    - **Refresh when remaining validity is <value> seconds**: Set to `300`.

For more details on the fields and options in this configuration window, see "Configure Kerberos clients" in the *API Gateway Policy Developer Guide*.

Configure a Kerberos profile for the Kerberos client

1. In the node tree, click **Environment Configuration > External Connections > Client Credentials > Kerberos**.
2. Add a Kerberos profile as follows:
    - **Profile Name**: `Authenticate to BackEndService using KCD`.

- **Kerberos Client**: `Trusted Kerberos Client for KCD`.
- **Kerberos Service Principal**: `<Back-end Kerberos Service>`.
- **Send token with first request**: Select this option.

For more details on the fields and options in this configuration window, see "Configure Kerberos client credential profiles" in the *API Gateway Policy Developer Guide*.

Configure a Kerberos policy

The following section describes how to configure the Kerberos policy for KCD.

To start, add a new policy named, for example, `Kerberos KCD SPNEGO Client-Side`.

Configure the end user authentication method

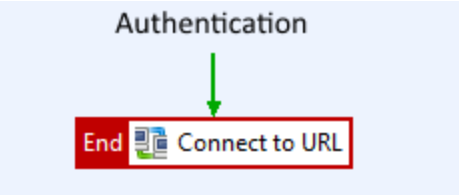
1. Configure the authentication mechanism the end user application requires. The required filters and configuration details depend on the type of authentication, for more details, see *API Gateway Policy Developer Guide*. For an example configuration, see *Configure a KCD demo setup* on page 52.
2. Right-click the first filter in your policy, and select **Set as Start**.

Configure connection to the back-end service

1. Open the **Routing** category in the filter palette, and drag a **Connect to URL** filter onto the policy canvas.
2. Enter the **URL** used to invoke the back-end service.
3. On the **Authentication** tab, select the Kerberos credential profile configured earlier (`Authenticate to BackEndService using KCD`), and click **Finish**.
   For more details on the fields and options in this configuration window, see "Connect to URL" in the *API Gateway Policy Developer Filter Reference*.

Build the policy

1. Click the **Add Relative Path** icon to create a new relative path `/kcd` that links to this Kerberos client-side policy.
2. Connect the filters with a success path.



The policy has the following flow:

- API Gateway authenticates the end user using a non-Kerberos authentication mechanism, and sets the message attribute `authentication.subject.id` to the user name of the end user.
- API Gateway generates a Kerberos principal name for the end user using the selector `${authentication.subject.id}@AXWAY.COM`.
- API Gateway checks the cache of impersonated subjects for valid credentials for the end user Kerberos principal name.
- If valid credentials are not found, API Gateway impersonates the end user, and sends a service ticket request in the name of the end user to the KDC.
- API Gateway sends the Kerberos token containing the service ticket in the name of the end user to the back-end Kerberos service.
- The back-end Kerberos service authenticates the end user and returns its response.

Configure the Kerberos system settings

1. In the node tree, click **Environment Configuration > Server Settings > Security > Kerberos**, and click **Add Kerberos Configuration**.
2. On the **krb5.conf** tab, add the Kerberos settings as follows:

```
[libdefaults]
default_realm = AXWAY.COM
renewable=true
proxiable=true
forwardable=true

[realms]
AXWAY.COM = {
kdc = dc.axway.com
}
```

   For KCD, the setting `forwardable` must be `true`.

   Replace the realm settings in the example with your Kerberos realm, and set the `kdc` setting to the host name of your Windows Domain Controller.

For more details on the fields and options in this configuration window, see "Kerberos configuration" in the *API Gateway Policy Developer Guide*.

Deploy the configuration

To deploy the configuration to API Gateway, click the **Deploy** icon.

You have now configured and deployed a simple KCD policy for SPNEGO authentication where API Gateway acts as the trusted Kerberos principal for KCD. The end user application that invokes this policy in API Gateway must provide authentication credentials to satisfy the chosen non-Kerberos authentication mechanism.

For demonstration purposes, you can add API Gateway as the back-end service as well as sample users. See *Configure a KCD demo setup* on page 52.

For other use cases covered in this guide, see *Kerberos use cases* on page 12.

Test the configuration

Use a client application to call the KCD policy in API Gateway.

The back-end Kerberos service should send a confirmation on a successful authentication.

The **Traffic Monitor** tab on the API Gateway Manager (`https://localhost:8090`) is an excellent place to view and troubleshoot the message flows. For more details, see "Monitor services in API Gateway Manager" in the *API Gateway Administrator Guide*.

Configure a KCD demo setup

To test or demonstrate KCD, you may want to configure a test back-end service as well as sample users.

Configure a back-end service for testing

For demonstration purposes, you can use another API Gateway instance as the back-end Kerberos service. API Gateway is configured as the Kerberos service for the most part the same way for both KCD and standard Kerberos authentication in the client-side transaction. For more details, see *Configure API Gateway to act as the Kerberos service* on page 19.

The difference between KCD and standard SPNEGO configuration is that for KCD, the back-end service must have a Service Principal Name (SPN). For more details, see *Map an SPN to the user account* on page 31.

Configure sample authentication

For demonstration purposes, you can configure HTTP Basic authentication against a local user repository as the incoming authentication mechanism on API Gateway for the end user.

Configure sample users

You can quickly add some sample users to a local repository in Policy Studio.

The user identity in the local repository must be mappable to an end user Kerberos principal name, so that when the trusted Kerberos principal impersonates an end user, the original end user can be identified in Active Directory. The setup in this guide uses a selector expression `${authentication.subject.id}@AXWAY.COM` for the mapping. For more details, see *Configure Kerberos principals* on page 47.

For example, if your end user Kerberos principal names were `Bob@AXWAY.COM`, and `Bill@AWAY.COM`, then add users named Bob and Bill to the local user repository.

1. In the node tree, click **Environment Configuration > Users and Groups > Users**.
2. Click **Add**, and fill in the details for your user. For example:

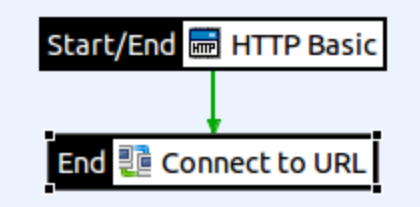
Bob	Bill
User Name: `Bob`	User Name: `Bill`
Password: `changeme`	Password: `changeme`

The passwords in the local user repository do *not* need to match these users' Kerberos passwords in the Key Distribution Center. Here, the user names and passwords configured in the local repository are passed to API Gateway over HTTP Basic.

Configure HTTP Basic authentication

In this example, API Gateway authenticates the end users using HTTP Basic.

1. Open the **Authentication** category, and drag a **HTTP Basic** filter onto the policy canvas.
2. Set **Credential Format** to **User Name**, and select **Allow client challenge**.
3. Set **Repository Name** to **Local User Store**, and click **Finish**.
   For more details on the fields and options in this configuration window, see "HTTP basic authentication" in the *API Gateway Policy Developer Filter Reference*.
4. Right-click the **HTTP Basic** filter, and select **Set as Start**.
5. Connect the filters with a success path.



To test the configuration, see *Test the configuration* on page 52.

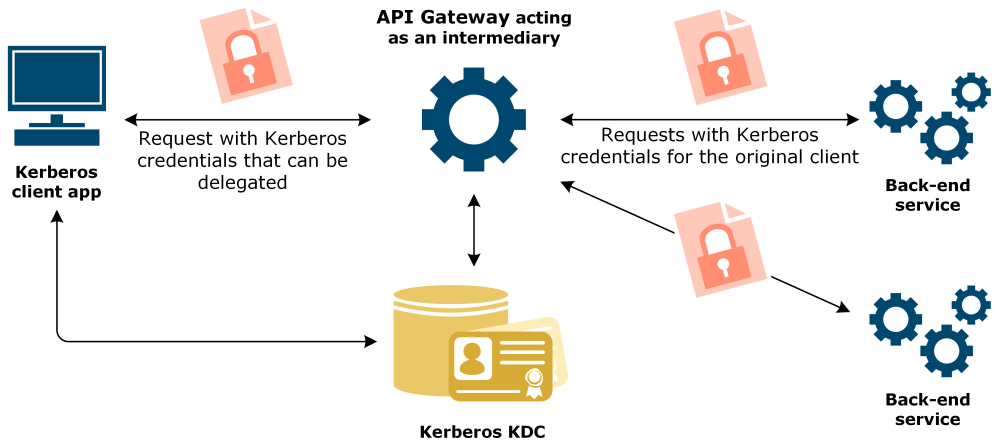
API Gateway in unconstrained credentials delegation 6

When authenticating to API Gateway using Kerberos, a client application can delegate its Kerberos credentials to API Gateway. API Gateway acts as an intermediary between a Kerberos client and Kerberos back-end services, and requests service tickets to *any* other Kerberos service in the same Kerberos realm on behalf of the client. These service tickets can then be used to authenticate the original end user to the other Kerberos services. This type of Kerberos delegation is often called *unconstrained* or *open*.

- **Client application**: Supports Kerberos authentication.
- **Back-end service**: Requires Kerberos authentication with the end user's credentials. Multiple back-end services may exist.
- **API Gateway**: Acts as both a Kerberos service and client, and authenticates to different back-end services as the end user.

API Gateway receives a request from the Kerberos client containing a SPNEGO token with the credentials to be delegated (the Ticket Granting Ticket (TGT)). The token also contains a service ticket that allows the Kerberos client to authenticate to API Gateway that is itself acting as a Kerberos service.

Using Kerberos authentication, API Gateway authenticates the Kerberos client. API Gateway then requests a service ticket to the back-end Kerberos service on behalf of the client, and authenticates to the back-end service as the original end user. The client application itself does not need to have any knowledge of any of the back-end services that API Gateway might invoke on its behalf.



A safer method for credentials delegation in Kerberos authentication is *constrained* delegation. In constrained delegation with protocol transition, the Kerberos service can obtain a Kerberos service ticket to itself on behalf of a Kerberos principal (end user) without requiring the principal to initially authenticate using Kerberos. Constrained delegation also restricts which back-end services the Kerberos service can request Kerberos service tickets on behalf of the client. The credential delegation is only allowed to a constrained set of Kerberos services that are configured in the Kerberos Key Distribution Center (KDC). For more details, see *API Gateway in Kerberos constrained delegation* on page 43.

Prerequisites

Before you start configuration, you must have API Gateway installed on any machine with access to the Windows Domain Controller. The machine does *not* have to be a Windows machine that is part of the Windows Domain.

Configuration process

The configuration process has the following steps:

1. *Configure Active Directory* on page 55
2. *Configure Kerberos principals* on page 57
3. *Configure API Gateway policy* on page 58
    - *Configure an intermediary Kerberos service* on page 58
    - *Configure a Kerberos client for the delegated credentials* on page 59
    - *Configure a Kerberos profile for the intermediary Kerberos service* on page 59
    - *Configure an intermediary policy* on page 60
    - *Configure Kerberos system settings* on page 62

Example names

The example name for the intermediary Kerberos service used in this guide is `IntermediaryGateway`. You can use this name, or replace it with a name of your own.

The example Kerberos realm name `AXWAY.COM` is specific to the examples in this guide. Replace the example realm name with your own realm name.

Configure Active Directory

This section describes how to configure a Kerberos service principal for API Gateway in Active Directory acting as the Key Distribution Center (KDC).

1. On the Windows Domain Controller, click **Control panel > Administrative Tools > Active Directory User and Computers**.
2. Right-click **Users**, and select **New > User**.
3. Enter a name for the Kerberos principal (such as `IntermediaryGateway`) in the **First Name** and **User Logon Name** fields, select your Active Directory domain from the drop-down menu (`@axway.com`), and click **Next**.
4. Enter the password, and do the following:
   - **User must change password at next logon**: Deselect this.
   - **User cannot change password**: Select this.
   - **Password never expires**: Select this.

     This ensures that a working API Gateway configuration does not stop working when a user chooses, or is prompted to change their password. API Gateway does not track these actions.

     If these options are not suitable in your implementation and a user password changes in Active Directory, you must then update the password or keytab of the Kerberos client or service related to the user in Policy Studio, and redeploy the configuration to API Gateway.
     If you cannot deselect **User must change password at next logon**, ensure the user changes the password and that the new password or keytab is deployed to API Gateway *before* API Gateway attempts to connect as this user.

Tip You can store Kerberos passwords in a KPS table to update a changed password in runtime. For more details, see *Use KPS to store passwords for Kerberos authentication* on page 64.

5. Click **Next > Finish**.
6. Map a Service Principal Name (SPN) to the user account. The Kerberos client uses the SPN to uniquely identify a service. To map the SPN, open a command prompt on the Windows Domain Controller, and enter the following command:

```
> ktpass -princ HTTP/<host>@<Kerberos realm> -mapuser
<user> -pass password -out <user>.keytab -crypto rc4-
hmac-nt -kvno 0
```

   The SPN is of the format `HTTP/<host>@<Kerberos realm>`, where `<host>` is the name of the host running the Kerberos service, `IntermediaryGateway` in this case:

```
> ktpass -princ
HTTP/IntermediaryGateway.axway.com@AXWAY.COM -mapuser
IntermediaryGateway -pass Axway123 -out
IntermediaryGateway.keytab -crypto rc4-hmac-nt -kvno 0
```

   Replace the example Kerberos realm name with your own realm name. Note that the realm name is uppercase.

The command creates an SPN `HTTP/IntermediaryGateway.axway.com@AXWAY.COM`, which is mapped to the `IntermediaryGateway` user account. The command also creates a keytab file for the account that you can use later when configuring a Kerberos service for API Gateway in Policy Studio. See *Configure API Gateway policy* on page 58.

Tip If you do not want to create a keytab file, you can use the following command:

```
> setspn -A HTTP/<host> <user>
```

As a Kerberos service, API Gateway authenticates the client application using Kerberos authentication. For the authentication to succeed, the client application or end user must also have an account configured in your Active Directory. For an example configuration for the client account, see *Configure a user account in Active Directory* on page 23. You must also configure user accounts and Service Principal Names (SPN) for the back-end services you want API Gateway to request service tickets for.

7. Right-click on the new user, and select **Properties > Delegation**. Then, select the **Trust this user for delegation to any service (Kerberos only)** option.

This is required for the API Gateway to extract delegated credentials when using unconstrained delegation where the client is the browser.

For the next steps, see *Configure Kerberos principals* on page 57.

Configure Kerberos principals

This section describes how to add Kerberos principals for the intermediary Kerberos service, and back-end Kerberos service for unconstrained credentials delegation using Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Principals**.
2. Add a new Kerberos principal for the intermediary Kerberos service account as follows:
   - **Name**: `IntermediaryGateway`
   - **Principal Name**: `IntermediaryGateway@AXWAY.COM`
   - **Principal Type**: `NT_USER_NAME`
3. Add a new Kerberos principal for the back-end Kerberos service account as follows:
   - **Name**: `<Back-end service name>` (for example, `Back-end Kerberos Service`)
   - **Principal Name**: `<Service Principal Name for the back-end service>` (for example, `HOST/BackEndService.axway.com@AXWAY.COM`)
   - **Principal Type**: `NT_USER_NAME`

For more details on the fields and options in this configuration window, see "Configure Kerberos principals" in the *API Gateway Policy Developer Guide*.

For the next steps, see *Configure API Gateway policy* on page 58.

Configure API Gateway policy

This section describes how to configure API Gateway for unconstrained credential delegation using Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

Configure an intermediary Kerberos service

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Services**.
2. Click **Add a Kerberos Service**, and enter a name for your Kerberos service (`IntermediaryGateway Kerberos Service for Unconstrained Delegation`).
3. On the **Kerberos Endpoint** tab, set the following:
    - **Kerberos Principal**: `IntermediaryGateway`.
    - **Enter Password**: Enter the password for `IntermediaryGateway@AXWAY.COM`.
    - **Enabled**: Select this option.
4. On the **Advanced** tab, set the following:
    - **Mechanism**: `SPNEGO_MECHANISM`.
    - **Extract delegated credentials**: Select this option.

Note Selecting **Extract delegated credentials** means that API Gateway extracts the Kerberos client's TGT from the SPNEGO token after the client has been authenticated. API Gateway can then use the TGT to request service tickets to other Kerberos services on behalf of the Kerberos client.

For more details on the fields and options in this configuration window, see "Configure Kerberos services" in the *API Gateway Policy Developer Guide*.

Configure a Kerberos client for the delegated credentials

To authenticate to the back-end Kerberos services, API Gateway loads the credentials it extracted from the end user to a Kerberos client of its own.

1. In the node tree, click **Environment Configuration > External Connections > Kerberos Clients**.
2. Click **Add a Kerberos Client**, and enter a name for your client (`Kerberos Client for Unconstrained Delegation`).
3. On the **Kerberos Endpoint** tab, set the following:
    - **Load from delegated credentials**: Select this option.
    - **Enabled**: Make sure this option is selected.
4. On the **Advanced** tab, set the following:
    - **Mechanism**: `SPNEGO_MECHANISM`.
    - **Context Settings**: Select the following options:
        - **Mutual authentication**
        - **Integrity**
        - **Confidentiality**
        - **Anonymity**
        - **Replay Detection**
        - **Sequence Checking**
    - **Synchronize to Avoid Replay Errors at Service**: Deselect this option to improve performance.
    - **Refresh when remaining validity is <value> seconds**: Set to `300`.

For more details on the fields and options in this configuration window, see "Configure Kerberos clients" in the *API Gateway Policy Developer Guide*.

Configure a Kerberos profile for the intermediary Kerberos service

1. In the node tree, click **Environment Configuration > External Connections > Client Credentials > Kerberos**.
2. Add a Kerberos profile as follows:
    - **Profile Name**: `Authenticate to Back-End Service using`

`Delegated Credentials`.

- **Kerberos Client**: `Kerberos Client for Unconstrained Delegation`.
- **Kerberos Service Principal**: `<Back-end Kerberos Service>`.
- **Send token with first request**: Select this option.

For more details on the fields and options in this configuration window, see "Configure Kerberos client credential profiles" in the *API Gateway Policy Developer Guide*.

Configure an intermediary policy

The following section describes how to configure the policy for API Gateway delegating the credentials.

To start, add a new policy named, for example, `Kerberos Intermediary for Unconstrained Credentials Delegation`.

Configure a Kerberos service filter

1. Open the **Authentication** category in the filter palette, and drag a **Kerberos Service** filter onto the policy canvas.
2. Set **Kerberos Service** to the intermediary Kerberos service you created (`IntermediaryGateway Kerberos Service for Unconstrained Delegation`).
3. Change **Kerberos Standard** to **SPNEGO Over HTTP**, and click **Finish**.
4. Right-click the **Kerberos Service** filter, and select **Set as Start**.

For more details on the fields and options in this configuration window, see "Kerberos service authentication" in the *API Gateway Policy Developer Filter Reference*.

Configure retrieving the end user credentials

1. Open the **Attributes** category in the palette, and drag a **Retrieve from HTTP Header** filter onto the policy canvas.
2. Set the **HTTP Header name** to `WWW-Authenticate` and **Attribute ID** to `outer.www.authenticate`, and click **Finish**.
   For more details on the fields and options in this configuration window, see "Retrieve attribute from HTTP header" in the *API Gateway Policy Developer Filter Reference*.
3. Open the **Conversion** category in the palette, drag a **Remove HTTP Header** filter onto the policy canvas.
4. Set **HTTP Header Name** to `WWW-Authenticate`.

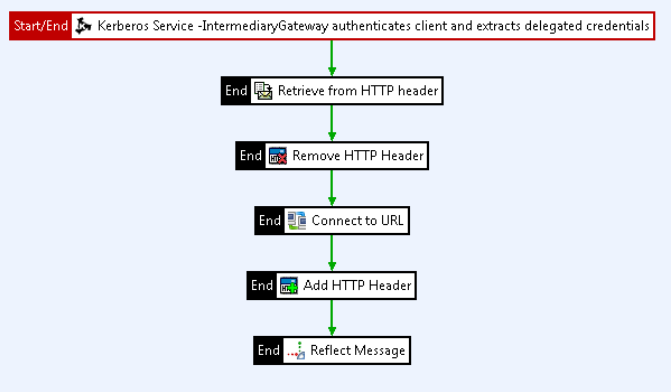
Configure authentication to the back-end service

1. Open the **Routing** category in the palette, and drag a **Connect to URL** filter onto the canvas.
2. Enter the **URL** used to invoke the back-end Kerberos service.
3. On the **Authentication** tab, select the Kerberos profile you configured (`Authenticate to Back-End Service using Delegated Credentials`), and click **Finish**.
   For more details on the fields and options in this configuration window, see "Connect to URL" in the *API Gateway Policy Developer Filter Reference*.
4. Open the **Conversion** category in the palette, and drag an **Add HTTP Header** filter onto the policy canvas.
5. Set the following, and click **Finish**:
   - **HTTP Header Name**: `WWW-Authenticate`.
   - **HTTP Header Value**: `${outer.www.authenticate}`.
   - **Override existing header**: Select this option.
   - **Add header to HTTP headers attribute**: Select this option.

     For more details on the fields and options in this configuration window, see "Add HTTP header" in the *API Gateway Policy Developer Filter Reference*.
6. Open the **Utility** category in the palette, and drag a **Reflect Message** filter onto the canvas.

Build the policy

1. Click on the **Add Relative Path** icon to create a new relative path `/intermediary` that links to this policy.
2. Connect the filters with success paths.



The policy has the following flow:

- API Gateway receives a request from the end user, and uses the Kerberos token in the `Authorization` HTTP header to authenticate the end user.

- API Gateway extracts the value of the `WWW-Authenticate` HTTP header and saves the value to a message attribute, so it can be reinstated later. This token is the response to the original token the end user sent.
- API Gateway retrieves a service ticket for the end user to access the back-end Kerberos service, connects to the back-end Kerberos service, and authenticates using the Kerberos credentials relating to the original end user.
- API Gateway reinstates the value of the `WWW-Authenticate` HTTP Header, overriding the value the back-end Kerberos service set.
- API Gateway sends the response to the Kerberos client.

Configure Kerberos system settings

1. In the node tree, click **Environment Configuration > Server Settings > Security > Kerberos**, and click **Add Kerberos Configuration**.
2. On the **krb5.conf** tab, add the Kerberos settings as follows:

```
[libdefaults]
default_realm = AXWAY.COM

[realms]
AXWAY.COM = {
kdc = dc.axway.com
}
```

Replace the realm settings in the example code with your Kerberos realm, and set the `kdc` setting to the host name of your Windows Domain Controller.

For more details on the fields and options in this configuration window, see "Kerberos configuration" in the *API Gateway Policy Developer Guide*.

Deploy the configuration

To deploy the configuration to API Gateway, click the **Deploy** icon.

You have now configured and deployed a policy for the authenticating Kerberos service on API Gateway that delegates the SPNEGO credentials to the back-end Kerberos service. The client application calls the policy on relative path `/intermediary`.

For demonstration purposes, you may want to add API Gateway as the client application and the back-end service. For example configurations, see *Demo setup: API Gateway as both Kerberos client and service* on page 13. When configuring API Gateway as the client application for credentials delegation, the setting `forwardable` on the `krb5.conf` tab in the Kerberos system settings must be `true`:

```
[libdefaults]
default_realm = AXWAY.COM
forwardable=true

[realms]
AXWAY.COM = {
kdc = dc.axway.com
}
```

For a list of other use cases covered in this guide, see *Kerberos use cases* on page 12.

Test the configuration

Use a client application to call the policy in API Gateway.

The back-end Kerberos service should send a confirmation on a successful authentication.

The **Traffic Monitor** tab on the API Gateway Manager (`https://localhost:8090`) is an excellent place to view and troubleshoot the message flows. For more details, see "Monitor services in API Gateway Manager" in the *API Gateway Administrator Guide*.

Use KPS to store passwords for Kerberos authentication 7

Kerberos authentication in API Gateway relies on keeping API Gateway in sync with Active Directory. If a password changes in Active Directory, it must also be updated in API Gateway.

For example, you might have an Active Directory password policy where all passwords must change every 60 days and passwords that never change are not allowed. In this case, the updates to API Gateway are frequent, so it is important that they can be done easily, quickly, and without downtime.

To achieve this, you can use a Key Property Store (KPS) to store passwords for both Kerberos clients and Kerberos services. A KPS is a table of data that policies running on API Gateway can reference as needed using selectors. You can view, populate, and update the data in KPS tables using API Gateway Manager. When a password is changed in Active Directory, you can update the password in the KPS at runtime, instead of redeploying the API Gateway configuration, or restarting API Gateway.

- *Configure a KPS table for Kerberos passwords* on page 64
- *Populate data to the KPS table* on page 65
- *Update your Kerberos configuration to use the KPS table* on page 65

For more details on KPS tables, see the *API Gateway Key Property Store User Guide*.

Configure a KPS table for Kerberos passwords

This section describes how to configure a KPS table for storing passwords in Policy Studio. For more information on working in Policy Studio, see the *API Gateway Policy Developer Guide*.

KPS tables are stored in KPS collections, so you must have a KPS collection to which you add the new KPS table. You can use an existing KPS collection, or create a new collection called, for example, `Passwords` with no collection alias prefix. For more details on how to configure a KPS collection, see the *API Gateway Key Property Store User Guide*.

1. In the node tree, select the KPS collection you want to use, and click **Add Table**.
2. Enter a name for your table (for example, `Passwords`).
3. Click **Add**, enter an alias (such as `Kerberos`), and click **OK**.
4. In the KPS table you created, go to the **Structure** tab, and click **Add** to add a field to the table.
5. Set the following, and click **OK**:
    - **Name**: `name`
    - **Type**: `java.lang.String`

6. Click **Add**, set the following, and click **OK**:
    - **Name**: `password`
    - **Type**: `java.lang.String`
7. Select **Primary Key** for the field `name` and **Encrypted** for the field `password`.
8. Click **Save** in the top right corner to save the configuration, and deploy the configuration to API Gateway.

Populate data to the KPS table

Use API Gateway Manager to populate the KPS table with entries for Kerberos principals, and to update the data when it changes. For more information on working in API Gateway Manager, see the *API Gateway Administrator Guide*.

1. Log in to the API Gateway Manager, click **Settings > Key Property Stores**.
2. Select the KPS table you created (`Passwords`), and select **Actions > New Entry** to add a Kerberos principal to the KPS table.
3. In the **name** field, enter the Kerberos principal's user name in the Active Directory.
4. In the **password** field, enter the Kerberos principal's password from Active directory, and click **Save**.
5. Create an entry and fill in the user name and password from Active Directory for each of your Kerberos principals.

You now have a KPS table storing the passwords for Kerberos authentication.

To update the details in the table when a Kerberos principal's password changes in Active Directory, log in to API Gateway Manager, select your KPS table, select the principal you want to edit, and update the password to match Active Directory.

Update your Kerberos configuration to use the KPS table

You must update the Kerberos clients and Kerberos services to use selectors in the password field to pick up the actual passwords from the KPS table.

1. In the Policy Studio node tree, click **Environment Configuration > External Connections > Kerberos Clients**.
2. Select the Kerberos client you want, and click **Edit**.
3. Select **Wildcard Password**, and enter the following:

```
${kps.<KPS table alias>['<name>'].password}
```

For example:

```
${kps.Kerberos['TrustedGateway'].password}
```

Here, `TrustedGateway` is the value of the `name` field in the KPS table, and thus the user name of the Kerberos principal in the Active Directory (`TrustedGateway@AXWAY.COM`). The name field is used as the primary key for the KPS table.

4. Repeat these steps for all your Kerberos clients you want to use the KPS table.
5. In the node tree, click **Environment Configuration > External Connections > Kerberos services**, and repeat the steps for all your Kerberos services you want to use the KPS table.
6. Deploy the configuration to API Gateway.

Wireshark tracing for Kerberos authentication 8

You can use Wireshark, a third-party trace tool, to view the SPNEGO token data sent between a Kerberos client and service when the client authenticates to the service. For more details on Wireshark and to download and install the program, go to Wireshark web page.

In SPNEGO Kerberos authentication, Kerberos tokens are sent between the client and service using the `Authorization` HTTP header. Wireshark can parse, decrypt, and view the content of these tokens.

Because Wireshark can trace any application acting either as the Kerberos client or service, the information in this section is applicable for both API Gateway and third-party applications.

Use Wireshark to trace authentication between the client and service

SPNEGO tokens are used only for the Client-Server Authentication Exchange (the `AP_REQ` and `AP_REP` Kerberos messages) between the client and service. The `AP_REP` the Kerberos client sends to the Kerberos service contains a service ticket encrypted with the service's secret key. To view this data decrypted, you must import the service's keytab to Wireshark.

The messages sent between the client and the KDC to acquire TGTs and service tickets are not covered by SPNEGO. For information on how to view these messages in Wireshark, see *Use Wireshark to trace Authentication Service Exchange and Ticket-Granting Service Exchange* on page 71.

Import a Kerberos service keytab file into Wireshark

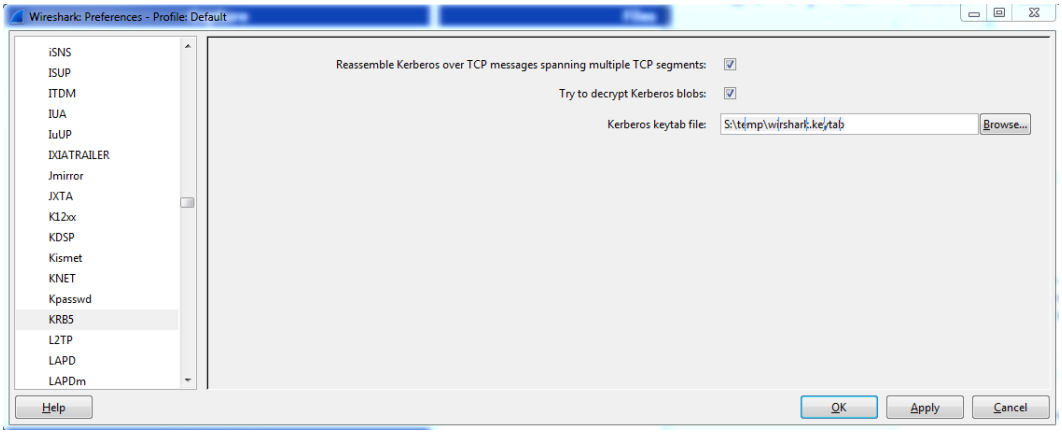
1. If you do not yet have a keytab file for your Kerberos service, create one by running the following command on the Windows Domain Controller:

```
> ktpass -princ <Kerberos service>@<Kerberos realm> -
pass ****** -out wireshark.keytab -ptype KRB5_NT_
PRINCIPAL
```

For example:

```
> ktpass -princ ServiceGateway@AXWAY.COM -pass ******
-out wireshark.keytab -ptype KRB5_NT_PRINCIPAL
```

2. In Wireshark, click **Edit > Preferences > Protocols > KRB5**.
3. Click **Browse**, and select the keytab file of your Kerberos service.



4. Select **Try to decrypt Kerberos blobs**, and click **Apply**.
5. Click **OK**.

Capture and analyze a Wireshark trace

If you have the Kerberos client and Kerberos service running on separate machines, run Wireshark on the same machine as the Kerberos client.

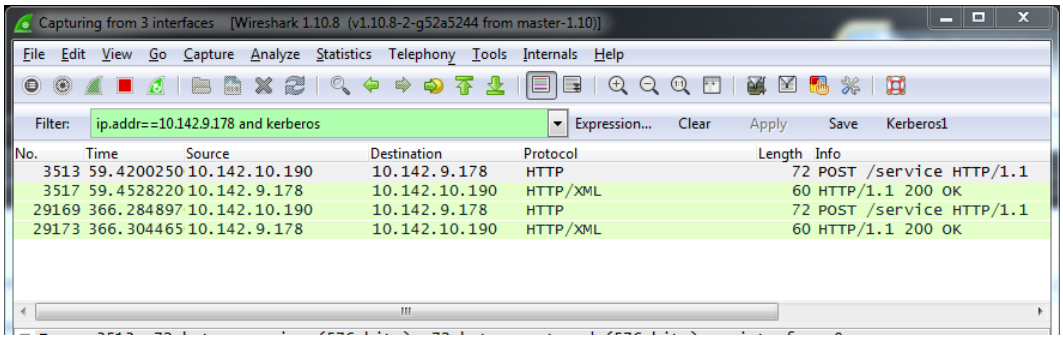
1. Start a Wireshark capture with the following filter:

```
ip.addr==<ip address of the machine running Kerberos
service> and kerberos
```

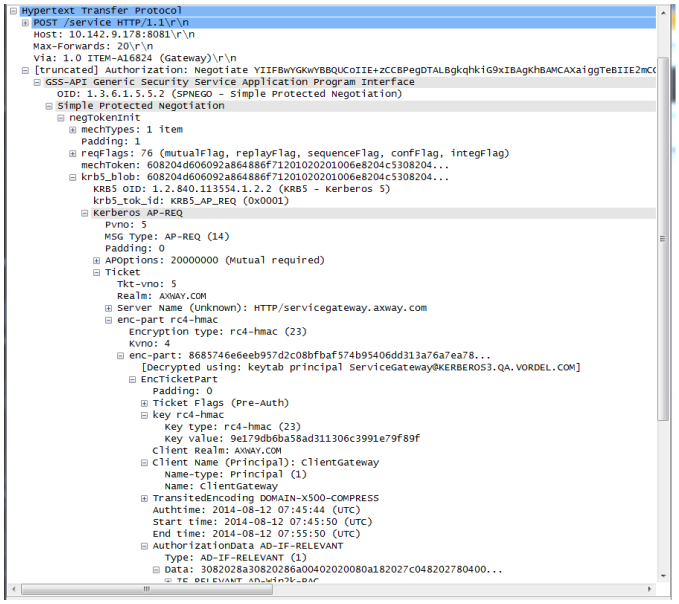
For example:

```
ip.addr==10.142.9.178 and kerberos
```

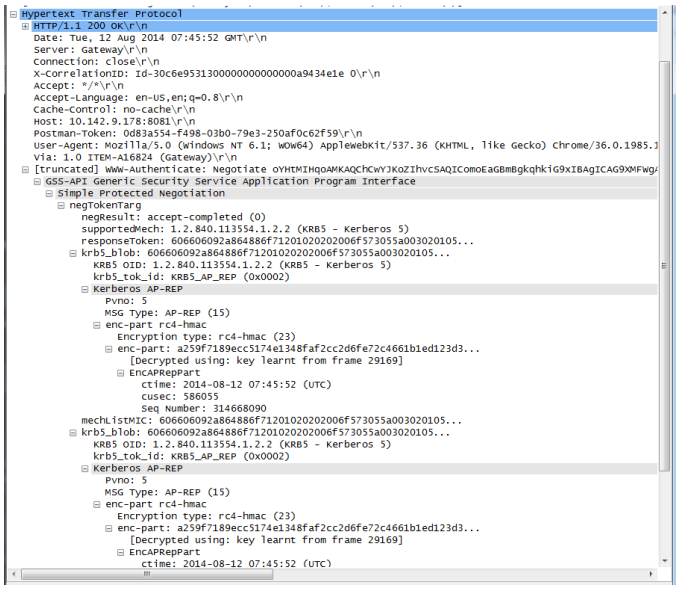
2. Send a message from the Kerberos client to the Kerberos service.
The Kerberos client calls the Kerberos service on the configured path (for example, `POST /service` in the Wireshark trace details).



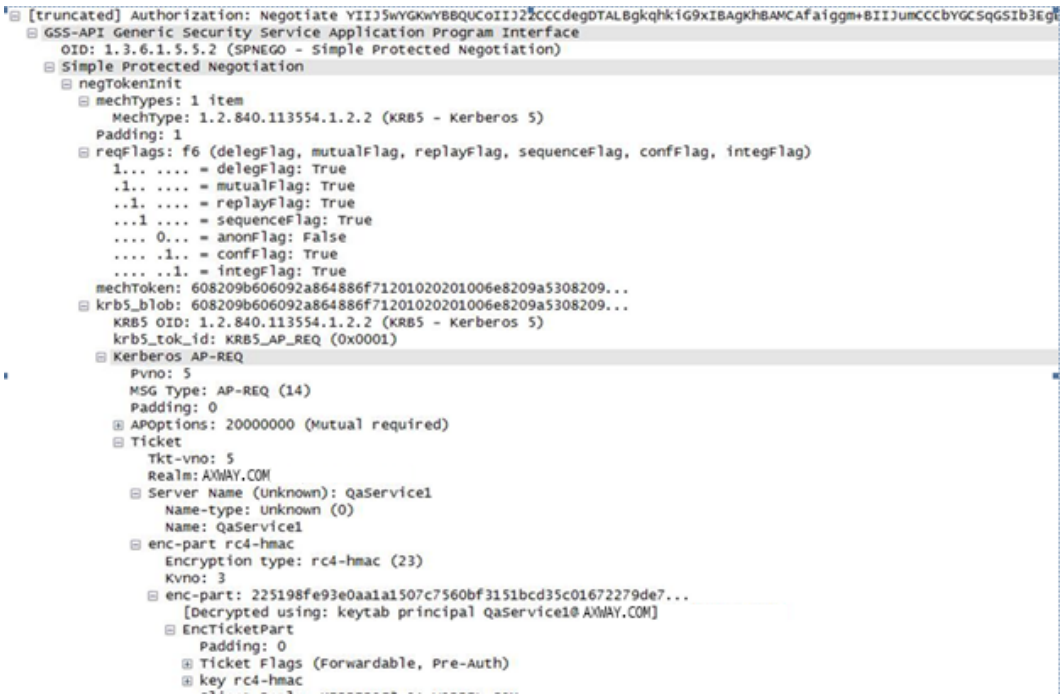
3. Select one of the `POST /<path>` lines in the top panel, and open the `Hypertext Transfer Protocol` in the lower panel.
4. To view the contents of the SPNEGO token sent from the Kerberos client to the Kerberos service, open the `Authorization` HTTP header. For example:



5. To view the response SPNEGO token the Kerberos service sends to the Kerberos client in the `WWW-Authenticate HTTP` header, select one of the `HTTP/1.1 200 OK` lines in the top panel, and expand the `Hypertext Transfer Protocol` in the lower panel. For example:



6. When tracing credential delegation, you must set the `forwardable` flag and the `delegFlag` in the `reqFlag` to `true` in the tickets.



Use Wireshark to trace Authentication Service Exchange and Ticket-Granting Service Exchange

You can use Wireshark to trace the Kerberos traffic between the Kerberos client and the Kerberos KDC (Windows Domain Controller). This traffic relates to the Kerberos Authentication Service Exchange (`AS-REQ` and `AS-REP`) and the Ticket-Granting Service Exchange (`TGS-REQ` and `TGS-REP`) when the client requests the TGT and service ticket.

If you have the Kerberos client and Kerberos service running on separate machines, run Wireshark on the same machine as the Kerberos client.

1. Start a Wireshark capture with the following filter:

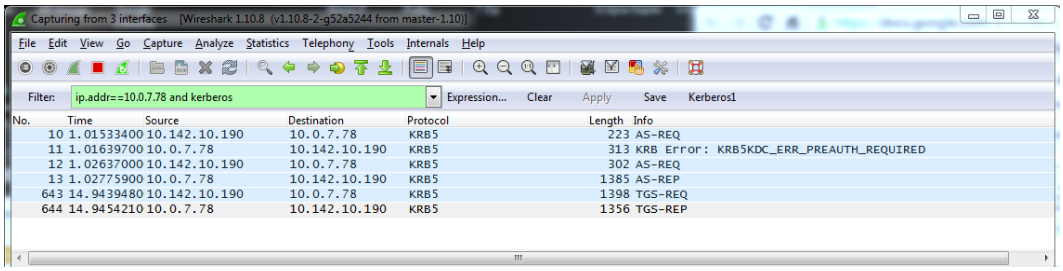
   ```
   ip.addr==<ip address of the Windows Domain Controller>
   and kerberos
   ```

   For example:

   ```
   ip.addr==10.0.7.78 and kerberos
   ```

2. Restart API Gateway running the Kerberos client. If the Kerberos client is a 3rd party application, you most likely need to restart the application as well to ensure that a cached TGT and service ticket are not used.
3. Send a message from the Kerberos client to the Kerberos service.

The Kerberos traffic is as follows:



- The `AS-REQ` and `AS-REP` are generated at the startup of API Gateway, because this is when the TGT for the Kerberos client is requested from the KDC. The TGT is only re-requested when it expires, because the TGT is cached in API Gateway.
- The `TGS-REQ` and `TGS-REP` are created when the Kerberos client sends a message to the Kerberos service to request the service ticket for the Kerberos service. The `TGS-REQ` is only sent from the Kerberos client on the first request, or when the service ticket has expired, because the service ticket is cached in API Gateway.

- The service ticket is encrypted with the secret key of the Kerberos service. To view the content decrypted, you must have the keytab of the Kerberos service imported in the Wireshark. See *Use Wireshark to trace authentication between the client and service* on page 67.