

# API Gateway

Version 7.6.2

14 July 2020

## PassPort Interoperability Guide



Copyright © 2020 Axway. All rights reserved.

This documentation describes the following Axway software:

Axway API Gateway 7.6.2

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

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

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

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

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

---

# Contents

<b>Preface</b>	<b>4</b>
Who should read this guide	4
How to use this guide	4
Related documentation	5
Support services	5
Training services	5
<b>Accessibility</b>	<b>6</b>
Screen reader support	6
Support for high contrast and accessible use of colors	6
<b>1 API Gateway and Axway PassPort</b>	<b>7</b>
Why use the products together	7
Prerequisites	7
API Gateway	8
Axway PassPort	8
Curl testing utility	8
Example LDAP structure	9
Further information	9
<b>2 Configure Axway PassPort</b>	<b>10</b>
Step 1 – Create the CSD	10
Step 2 – Create the product in PassPort (import the CSD)	11
Step 3 – Create the privileges in PassPort	13
Step 4 – Create the roles in PassPort	14
Step 5 – Map the PassPort roles to LDAP roles	15
Step 6 – Export Axway PassPort certificates	16
<b>3 Configure API Gateway</b>	<b>18</b>
Step 1 – Deploy the CSD file to API Gateway	18
Step 2 – Configure the authentication repository profile	18
Step 3 – Create a policy that delegates authentication to PassPort	20
Add a HTTP basic authentication filter	21
Add a PassPort authorization filter	21
Add a success message filter	22
Add a failure message filter	23
Step 4 – Add HTTP services and paths	24
Step 5 – Deploy the configuration	25
<b>4 Test API Gateway and Axway PassPort configuration</b>	<b>26</b>

---

# Preface

This guide describes how to configure API Gateway and Axway PassPort to work together.

## Who should read this guide

The intended audience for this guide is system integrators who are responsible for integrating API Gateway with other applications.

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

## How to use this guide

This guide should be used in conjunction with the other guides in the API Gateway documentation set. Before configuring API Gateway to work with Axway PassPort you should understand exactly what message filters are, and how they are chained together to create a message policy. These concepts are documented in detail in the *API Gateway Policy Developer Guide*.

You should also consult the documentation set for Axway PassPort.

Before you begin, review this guide thoroughly. The following is a brief description of the contents of each section:

[API Gateway and Axway PassPort on page 7](#) – Provides an overview of the interoperability scenario, including prerequisites for API Gateway and Axway PassPort.

[Configure Axway PassPort on page 10](#) – Describes how to configure Axway PassPort.

[Configure API Gateway on page 18](#) – Describes how to configure API Gateway.

[Test API Gateway and Axway PassPort configuration on page 26](#) – Describes how to test the configuration using the Curl testing utility.

## 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](mailto: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 on page 6](#)
- [Support for high contrast and accessible use of colors on page 6](#)

## 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.

# API Gateway and Axway PassPort

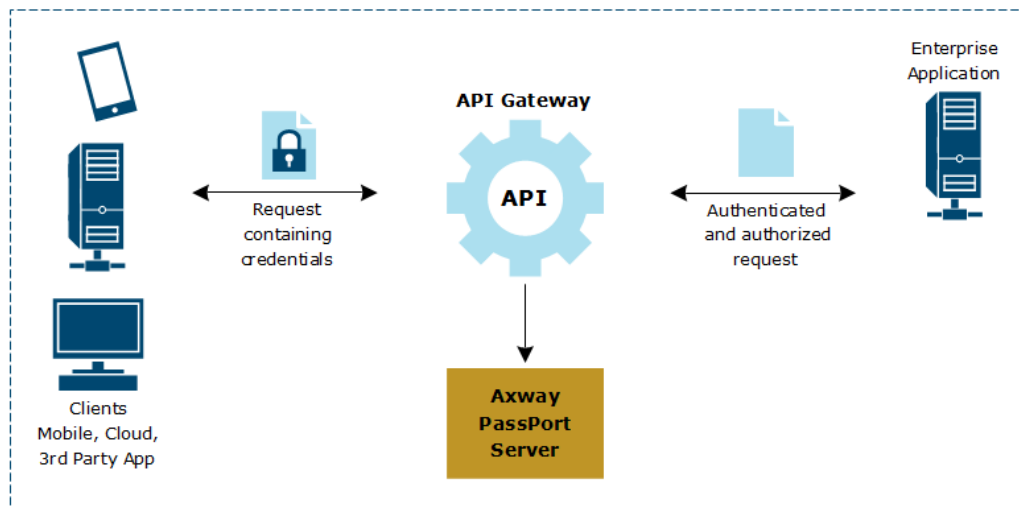
# 1

This guide describes the interoperability between API Gateway and Axway PassPort. For specific information about the installation and general use of either API Gateway or Axway PassPort, refer to their respective documentation.

## Why use the products together

You can use API Gateway to authenticate and authorize users against Axway PassPort.

The following diagram shows the sequence of events that occurs when a client sends a message to API Gateway. API Gateway authenticates the request sender against the authentication repository and then they are authorized against Axway PassPort. If the user is permitted access to the requested resource the request is routed to the Enterprise Application. Otherwise an appropriate fault message is returned to the client.



This guide describes a simple scenario that demonstrates how to configure API Gateway to authenticate a user and authorize their requests for a particular resource against an Axway PassPort server.

## Prerequisites

The prerequisites for API Gateway and Axway PassPort interoperability are as follows.

## API Gateway

You must install Axway API Gateway version 7.6.2 or higher, and have a valid API Gateway license file from Axway.

## Axway PassPort

You must install Axway PassPort version 4.6 Service Pack 1 or higher and have a valid PassPort license file from Axway.

**Note** Axway PassPort 4.4 Service Pack 9 or higher is also supported.

## Curl testing utility

The steps outlined in this guide are tested using the Curl testing utility to send HTTP POST requests to API Gateway. It is available from the following URL:

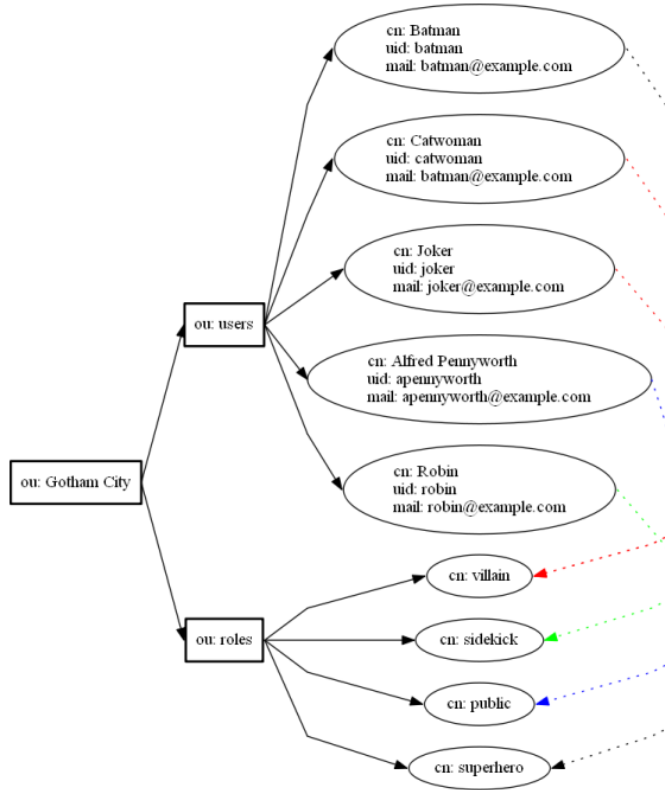
<http://curl.haxx.se/download.html>

Alternatively, you can use any client capable of sending HTTP POST requests with HTTP basic authentication.



## Example LDAP structure

This guide uses the following LDAP structure as an example in the configuration of API Gateway and Axway PassPort.



## Further information

The configuration example described in this guide can be part of a larger policy, including features such as XML threat detection and conditional routing. For more information on these features see the *API Gateway Policy Developer Guide* and the *API Gateway Policy Developer Filter Reference*.

# Configure Axway PassPort

# 2

This section describes how to configure Axway PassPort to work with API Gateway.

The following steps assume that you have installed Axway PassPort, and that you have configured a suitable identity store (for example, an LDAP server). For more information about configuring identity stores in LDAP, see the *Axway PassPort Administrator Guide*.

The steps involved in configuring PassPort are summarized as follows:

- [Step 1 – Create the CSD on page 10](#)
- [Step 2 – Create the product in PassPort \(import the CSD\) on page 11](#)
- [Step 3 – Create the privileges in PassPort on page 13](#)
- [Step 4 – Create the roles in PassPort on page 14](#)
- [Step 5 – Map the PassPort roles to LDAP roles on page 15](#)
- [Step 6 – Export Axway PassPort certificates on page 16](#)

## Step 1 – Create the CSD

In PassPort the resources and actions offered by client applications are described by their Component Security Descriptor (CSD). This is an XML file. For more information on CSD files, see the *Axway PassPort Administrator Guide*.

In this scenario the resource is the `/batcave` and the actions are the supported HTTP verbs `GET`, `POST`, and `DELETE`. In this case the CSD might look like:

```
<?xml version="1.0" encoding="UTF-8"?>
<AMImport CSDVersion="1"
  ComponentGroup="Gotham City" ComponentName="Lairs" ComponentVersion="1.0.0"
  ImportProtocolVersion="1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="importation.xsd">

  <ResourceDefinition>

    <Resource name="/batcave">
      <Description>
        <Content content="Used by: Lairs" lang="en"/>
      </Description>
      <AvailableActions>
```

```
<AvailableAction>
  <Action name="DELETE"/>
  <Description>
    <Content content="HTTP DELETE" lang="en"/>
  </Description>
</AvailableAction>

<AvailableAction>
  <Action name="POST"/>
  <Description>
    <Content content="HTTP POST" lang="en"/>
  </Description>
</AvailableAction>

<AvailableAction>
  <Action name="GET"/>
  <Description>
    <Content content="HTTP GET" lang="en"/>
  </Description>
</AvailableAction>

</AvailableActions>
</Resource>

</ResourceDefinition>
</AMImport>
```

**Tip** A script is available in the following location in your API Gateway installation, which can generate a CSD based on the current configuration of API Gateway:

```
INSTALL_DIR/scripts/passport/generateCSD.py
```

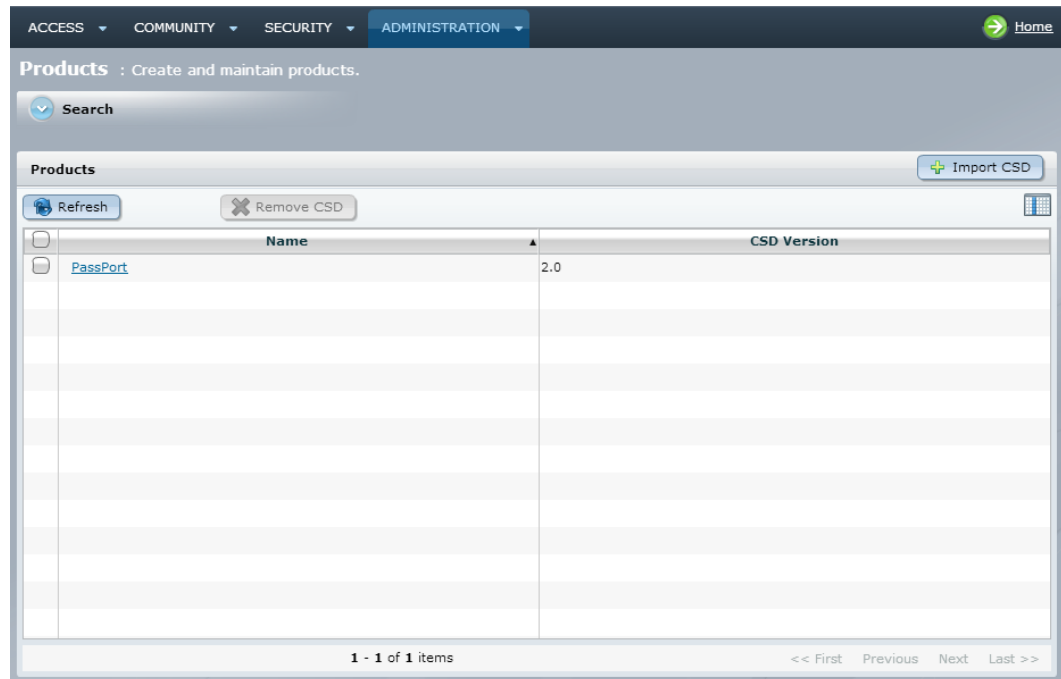
## Step 2 - Create the product in PassPort (import the CSD)

After the CSD is created you must import it into PassPort to create the product.

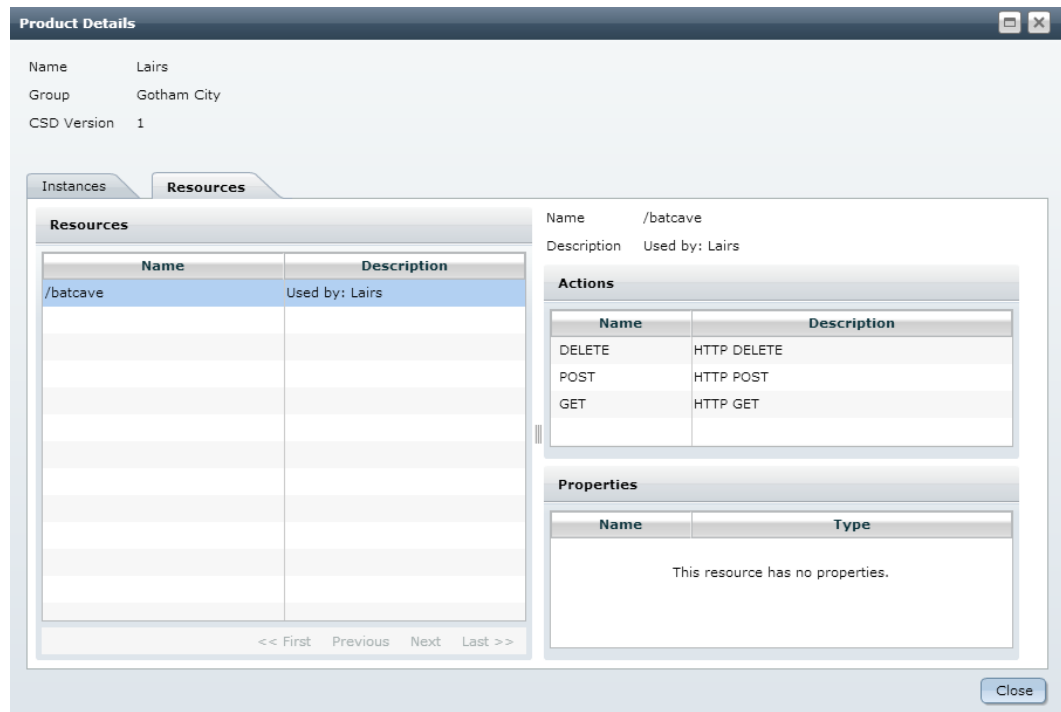
**Note** In PassPort version 4.4 these are called components.

Follow these steps:

1. In the PassPort Administration UI, select the **Administration** tab.



2. Click **Import CSD**, and select the CSD you created. This creates a Product called `Lairs` (this is the `ComponentName` in the CSD file).
3. To see the available resources and actions, click **Lairs** and click the **Resources** tab.

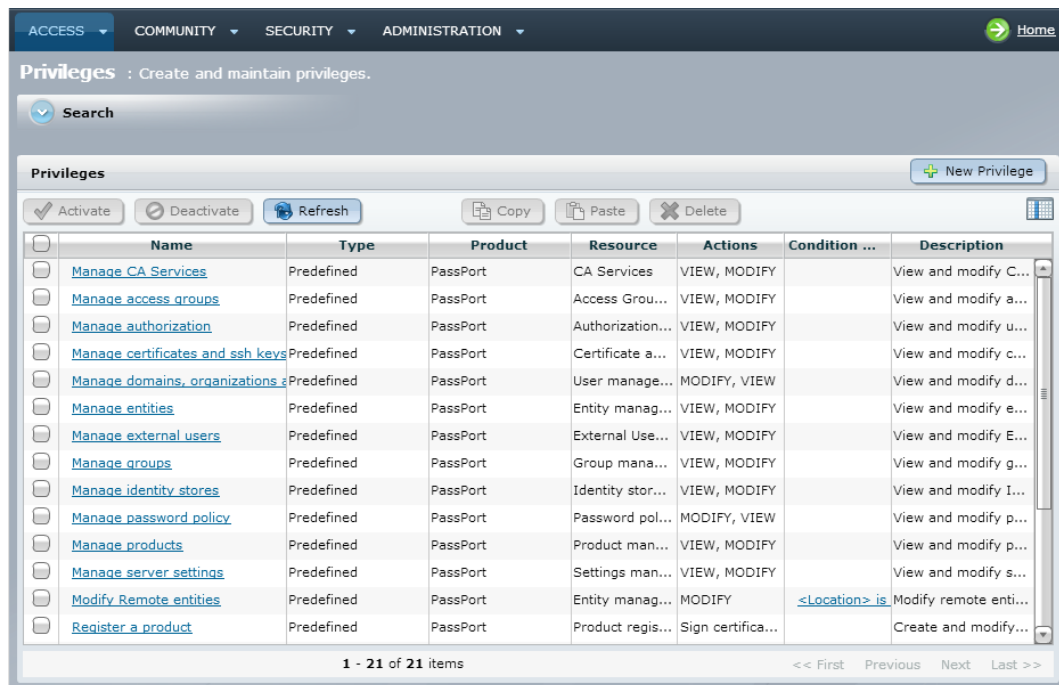


## Step 3 - Create the privileges in PassPort

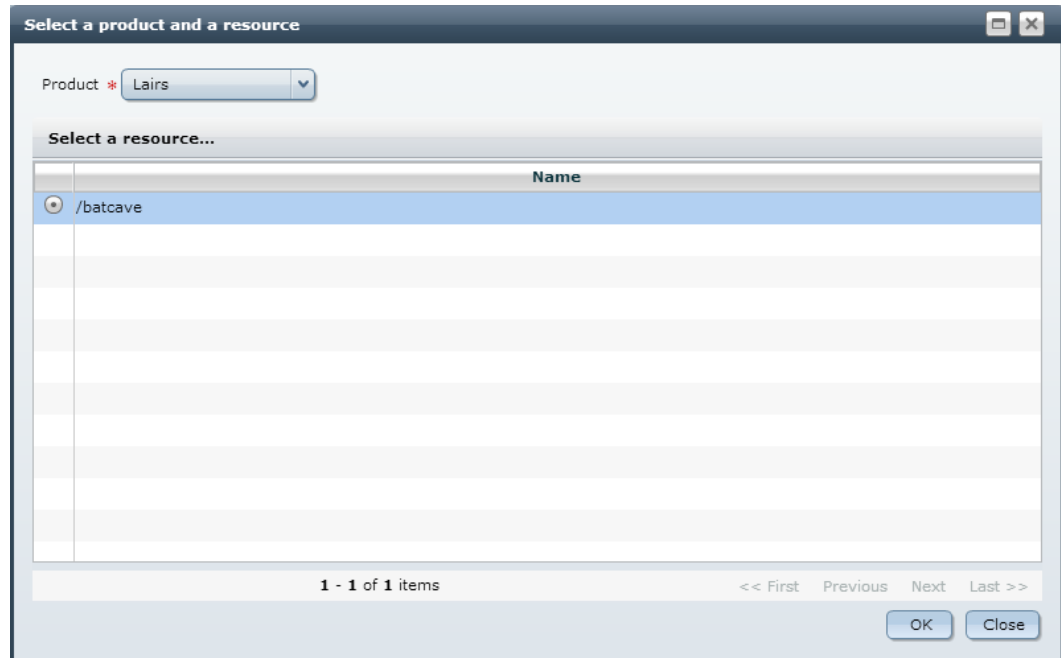
In PassPort a privilege is a grouping of actions on a specific resource. In this scenario two privileges are created; `Modify Lair` and `View Lair`. The modify privilege allows you to change the lair (POST/DELETE) while the view privilege only allows read access (GET).

Follow these steps:

1. In the PassPort Administration UI, select **Access > Privileges**.
2. Click **New Privilege**.



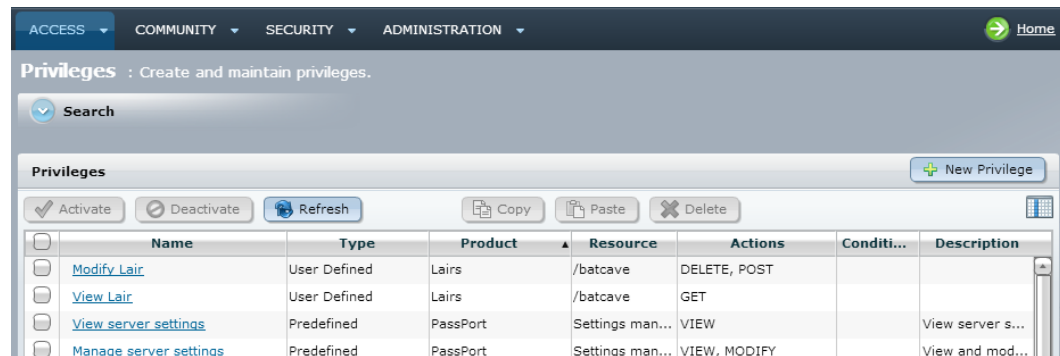
3. In the New Privilege dialog enter the name `Modify Lair` and click **Next**.
4. Click the search button next to the **Resource** field. This opens the product and resource selection dialog.
5. Select `Lairs` from the **Product** list.



6. Select the `/batcave` resource and click **OK**.
7. Select the **DELETE** and **POST** actions.
8. Click **Finish**.

To create the view privilege, repeat the above steps but this time enter the name `View Lair` and select the **GET** action.

There are now two user-defined privileges for the `Lairs` product:

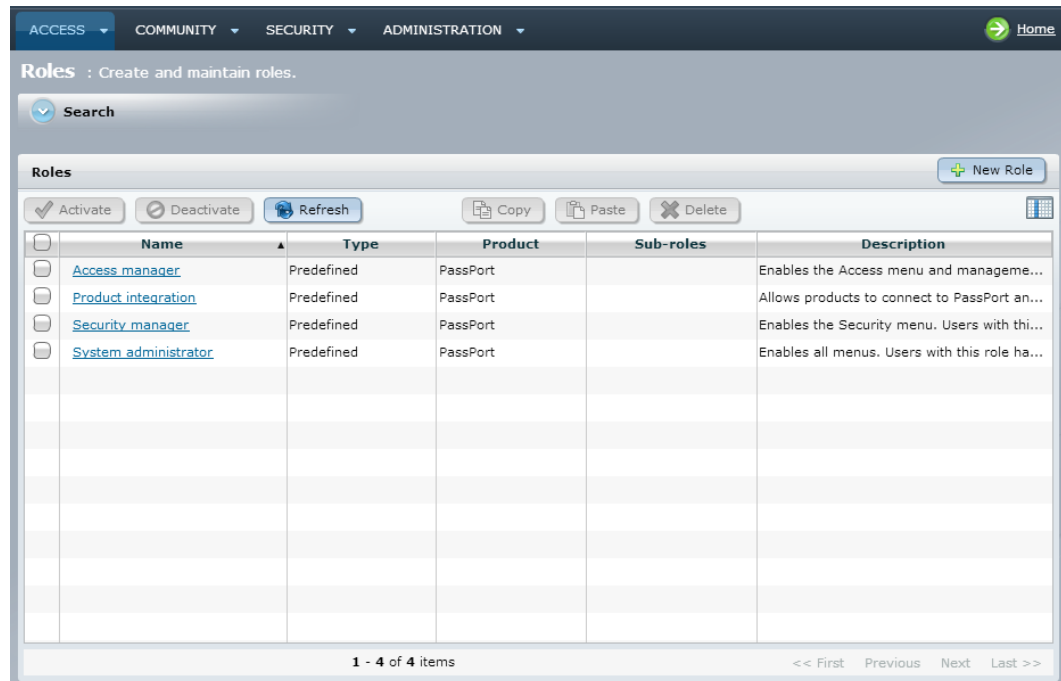


## Step 4 - Create the roles in PassPort

In PassPort a role is a grouping of privileges. In this scenario two roles are created; `Superhero` and `Sidekick`. The `Superhero` role has full access to the lair, the `Sidekick` role has read-only access, and everyone else has no access.

Follow these steps:

1. In the PassPort Administration UI, select **Access > Roles**.
2. Click **New Role**.



3. In the New Role dialog enter the name `Superhero`. Click **Next**.
4. Select the `Modify Lair` privilege and the `View Lair` privilege. Click **Finish**.

To create the `Sidekick` role, repeat the above steps but this time enter the name `Sidekick` and only select the `View Lair` privilege.

## Step 5 – Map the PassPort roles to LDAP roles

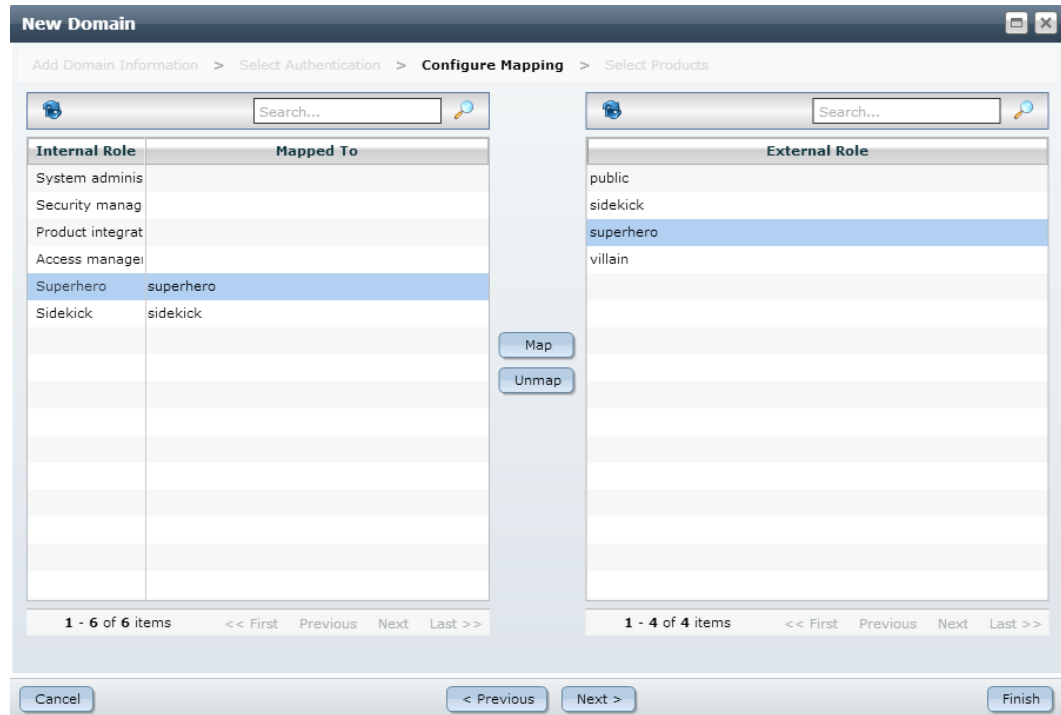
To create a mapping between roles defined in LDAP and roles defined in PassPort you must create a domain:

1. In the PassPort Administration UI, select **Access > Domains**.
2. Click **New Domain**.
3. In the New Domain dialog enter the name `Gotham`.

**Note** This name is referred to later in API Gateway when creating the Axway PassPort authentication repository. For more information, see [Step 2 – Configure the authentication repository profile on page 18](#).

4. Select **External Identity Store with External Authorization (External)**.
5. Deselect **Only assign the default PassPort product**.
6. Click **Next**.

7. Select the identity store for the LDAP server. The identity store must already have been created. For more information on configuring LDAP, see the *Axway PassPort Administrator Guide*.
8. Click **Next**.
9. In the Configure Mappings section the roles in LDAP are mapped to corresponding PassPort roles. Map `sidekick` to `Sidekick` and `superhero` to `Superhero`.



10. Click **Next**.
11. Select `Lairs`.
12. Click **Finish**.

## Step 6 – Export Axway PassPort certificates

To connect to Axway PassPort, API Gateway needs to trust the PassPort certificates. Secure communication with PassPort uses two different ports:

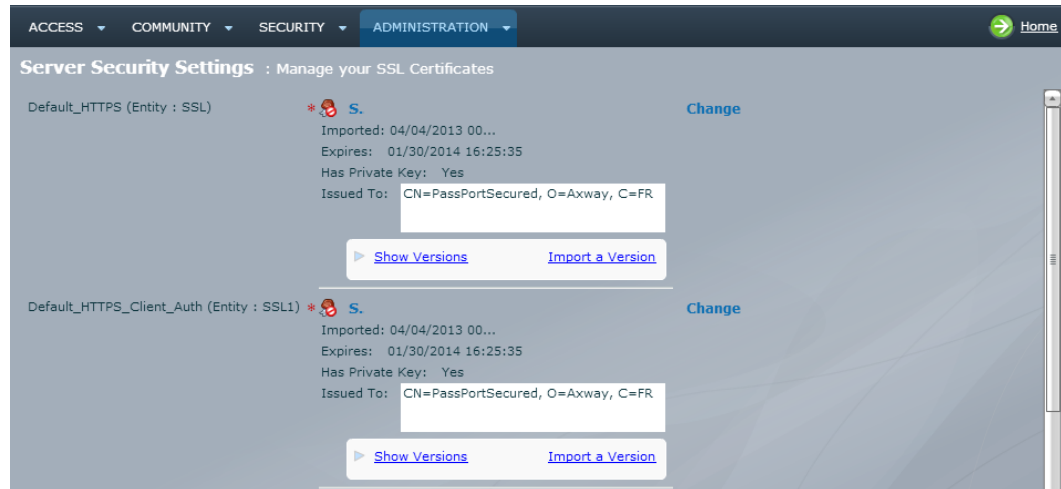
- HTTPS port
- HTTPS client authentication port

These ports might be using different certificates, in which case you must import both certificates into API Gateway's certificate store.

Follow these steps:

1. In the PassPort Administration UI, select **Administration > Server Security Settings**.
2. Make a note of the certificates being used for `Default_HTTPS` and `Default_HTTPS_Client_Auth`.





3. Click **Security > Certificates**.
4. Select the certificate that you noted down previously, and click the export button.
5. In the Export Certificate dialog select **.cer** as the **File Extension**.
6. Click **OK**. Click **Yes** on the confirmation dialog.
7. In the Save As dialog select a location to save the certificate to.

The certificates can be imported into API Gateway when you configure the authentication repository profile for PassPort. For more information, see [Step 2 – Configure the authentication repository profile on page 18](#).

---

# Configure API Gateway

# 3

This section describes how to configure API Gateway to delegate authentication and authorization to Axway PassPort.

The steps involved in configuring API Gateway are summarized as follows:

- [Step 1 – Deploy the CSD file to API Gateway on page 18](#)
- [Step 2 – Configure the authentication repository profile on page 18](#)
- [Step 3 – Create a policy that delegates authentication to PassPort on page 20](#)
- [Step 4 – Add HTTP services and paths on page 24](#)
- [Step 5 – Deploy the configuration on page 25](#)

## Step 1 – Deploy the CSD file to API Gateway

You must manually deploy the CSD file to API Gateway. This can be done at any stage, however we recommend doing it before deploying the Axway PassPort repository to avoid deployment errors.

Copy the CSD file that you used to create the product in PassPort to the `INSTALL_DIR/groups/GROUP-N/conf` folder.

**Note** If API Gateway is connecting to multiple PassPort servers, or to multiple products on a single PassPort server, there will be multiple CSD files.

In this scenario there is a single CSD file called `gotham.xml`.

## Step 2 – Configure the authentication repository profile

To configure the Axway PassPort authentication repository, follow these steps:

1. In Policy Studio, select **Environment Configuration > External Connections > Authentication Repositories**.
2. Select **Axway PassPort Repositories**.
3. Click **Add a new Repository**.
4. Complete the following fields:
  - **Repository Name:**  
Enter `Gotham PassPort`.
  - **Hostname:**

Enter the host name or IP address of the PassPort server

- **Shared Secret:**

Enter the shared secret. The PassPort shared secret is set during PassPort installation, and should be available from the PassPort administrator.

- **CSD Name:**

Enter the name of the CSD file to use. This file must exist in API Gateway's `INSTALL_DIR/groups/GROUP-N/conf` folder.

- **PassPort Certificates:**

Select the PassPort certificates that you exported from Axway PassPort in [Step 6 – Export Axway PassPort certificates on page 16](#). The **HTTPS Client Authentication** certificate is optional. It is only required if PassPort is configured to use different certificates for the different ports. Click **Select** to select a certificate in the certificates list and click **OK**. Alternatively, if the PassPort certificate has not yet been imported into API Gateway, you can click **Create/Import** to import a new certificate.

- **Authentication Domain:**

The Authentication Domain is the PassPort domain that is used by this authentication repository. This is the domain that was created in [Step 5 – Map the PassPort roles to LDAP roles on page 15](#).

5. Click **OK**.

The following figure shows an example of the completed configuration dialog.

The screenshot shows a configuration dialog box for Axway PassPort. It contains several sections with input fields and buttons:

- Repository Name:** Gotham PassPort
- Hostname:** gothamServer
- Shared Secret:** \*\*\*\*\*
- CSD Name:** gotham.xml
- PassPort Certificates:**
  - HTTPS:** Select button, gotham
  - HTTPS Client Authentication:** Select button, (unset)
- Ports:**
  - HTTPS:** 6453
  - HTTPS Client Authentication:** 6666
- Authentication:**
  - Domain:** Gotham

At the bottom are three buttons: OK (highlighted with a blue dashed border), Cancel, and Help.

## Step 3 - Create a policy that delegates authentication to PassPort

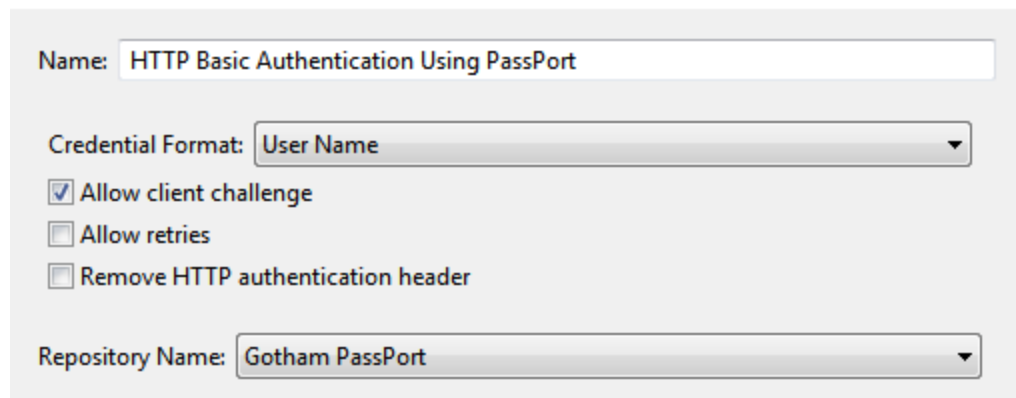
To delegate authentication to Axway PassPort, create a new policy in Policy Studio:

1. In the Policy Studio tree, right-click **Policies** and select **Add Policy**.
2. Enter the name `GothamCityAuthorization` and click **OK**.
3. Configure the policy as follows:
  - [Add a HTTP basic authentication filter on page 21](#)
  - [Add a PassPort authorization filter on page 21](#)
  - [Add a success message filter on page 22](#)
  - [Add a failure message filter on page 23](#)

## Add a HTTP basic authentication filter

To configure a HTTP Basic filter, perform the following steps:

1. Drag a **HTTP Basic** filter from the Authentication category in the palette and drop it onto the policy canvas.
2. Configure it as follows:
  - **Name:**  
Enter HTTP Basic Authentication Using PassPort.
  - **Credential Format:**  
Select User Name from the list.
  - **Allow Client Challenge:**  
Select the **Allow client challenge** check box.
  - **Repository Name:**  
Select Gotham PassPort from the list.



The screenshot shows the configuration window for the HTTP Basic Authentication filter. It includes the following fields and options:

- Name:** HTTP Basic Authentication Using PassPort
- Credential Format:** User Name (selected from a dropdown menu)
- ☒ Allow client challenge
- ☐ Allow retries
- ☐ Remove HTTP authentication header
- Repository Name:** Gotham PassPort (selected from a dropdown menu)

3. Click **OK**.
4. To set this authentication filter to be the starting filter of the policy, right-click the filter on the canvas and select **Set as Start**.

## Add a PassPort authorization filter

To configure the Axway PassPort Authorization filter, perform the following steps:

1. From the Authorization category on the right of Policy Studio, drag the **Axway PassPort Authorization** filter onto the policy canvas.
2. Select Gotham PassPort from the **PassPort Repository** list.

Name:	Axway PassPort Authorization
User ID:	<code>\${authentication.subject.id}</code>
Resource:	<code>\${http.request.uri}</code>
Action:	<code>\${http.request.verb}</code>
PassPort Repository:	Gotham PassPort ▼

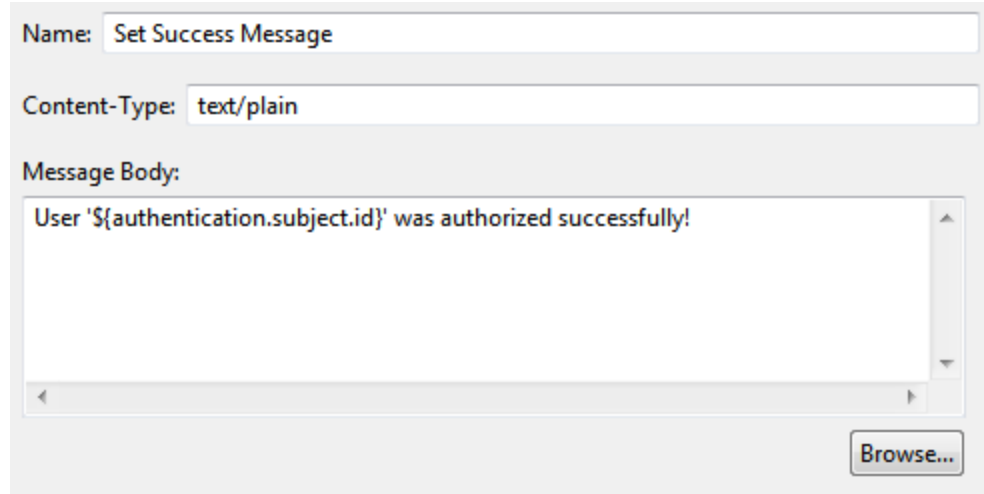
For this scenario use the default settings for the other fields.

- The default **User ID** is `${authentication.subject.id}`. This attribute gets set by the authentication filter.
  - The default **Resource** is `${http.request.uri}`. This is the URL of the request. In this scenario the URL is `/batcave` as defined in the CSD.
  - The default **Action** is `${http.request.verb}`. This is the HTTP action that the request is performing (for example, GET, POST, PUT, and so on).
3. Set the success path from the HTTP Basic filter to the Axway PassPort Authorization filter.

## Add a success message filter

To display a success message after successfully authorizing the user you can add a Set Message filter as follows:

1. Drag a **Set Message** filter from the Conversion category in the palette and drop it onto the policy canvas.
2. Configure the following fields on this filter:
  - **Name:**  
Enter `Set Success Message`.
  - **Content-Type:**  
Enter `text/plain` as the content-type of the message to return to the client.
  - **Message Body:**  
Enter the following message to return to the client: `User '${authentication.subject.id}' was authorized successfully!`



Name: Set Success Message

Content-Type: text/plain

Message Body:

User '\${authentication.subject.id}' was authorized successfully!

Browse...

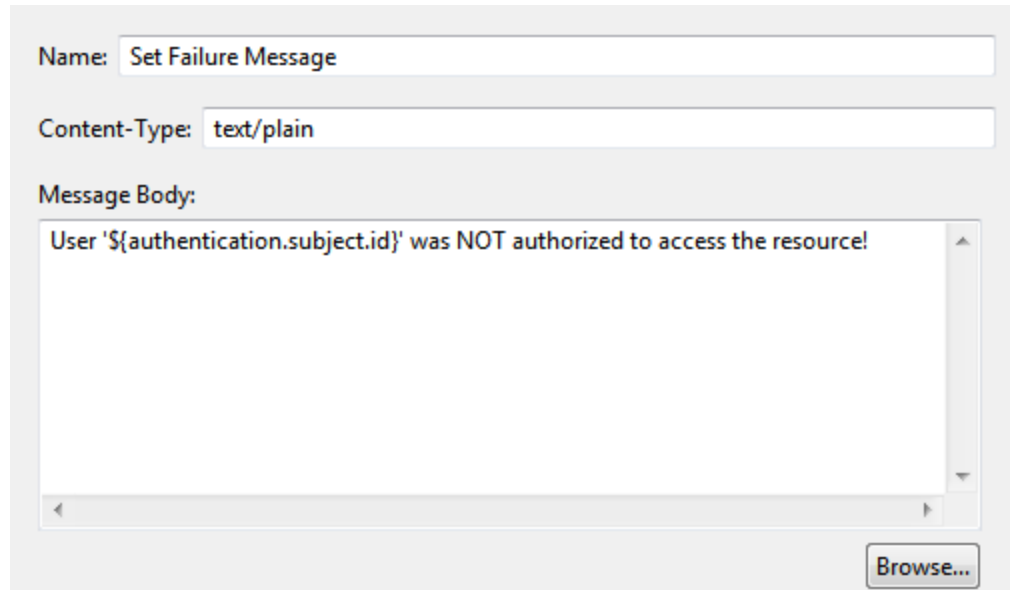
3. Click **OK**.
4. Set the success path from the Axway PassPort Authorization filter to the Set Success Message filter.

## Add a failure message filter

If Axway PassPort returns false for the authorization request an appropriate error message should be returned to the client.

To display a failure message after an unsuccessful authorization event you can add another Set Message filter as follows:

1. Drag a **Set Message** filter from the Conversion category in the palette and drop it onto the canvas.
2. Configure the following fields on this filter:
  - **Name:**  
Enter Set Failure Message.
  - **Content-Type:**  
Enter text/plain as the content-type of the message to return to the client.
  - **Message Body:**  
Enter the following message to return to the client: User  
'\${authentication.subject.id}' was NOT authorized to  
access the resource!



Name: Set Failure Message

Content-Type: text/plain

Message Body:

User '\${authentication.subject.id}' was NOT authorized to access the resource!

Browse...

3. Click **OK**.
4. Set the failure path of the Axway PassPort Authorization filter to the Set Failure Message filter.

## Step 4 - Add HTTP services and paths

Follow these steps to add HTTP services and paths:

1. In the Policy Studio tree, select **Environment Configuration > Listeners > API Gateway** and click **Add HTTP Services**.
2. In the **Name** field enter `Lairs` and click **OK**.
3. Select the **Ports** node and click **Add > HTTP Interface** to configure a port for the HTTP service.
4. Select the **Paths** node and click **Add > Relative Path**.
5. In the Resolve path to Policies dialog enter the path `/batcave`. For the Path Specific Policy select the `GothamCityAuthorization` policy. For example:




☒ Enable this path resolver

**Policies** | Audit Settings | HTTP Method | Advanced

When a request arrives that matches the path:

Call the following Policies:

☒ Global Request Policy

☒ Path Specific Policy:  

☒ Global Response Policy


6. On the **HTTP Method** tab, select `GET`.





7. Click **OK**.

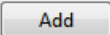

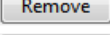
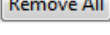
Create a path mapping for the `POST` and `DELETE` methods by repeating the steps to add a relative path, and selecting the appropriate HTTP method each time.

When you are finished, the following path resolvers should be configured:

**Resolvers**

 Filter

Resolver	Type	HTTP Method	Policy
 Paths			
 /batcave	Path	POST	GothamCityAuthorization
 /batcave	Path	DELETE	GothamCityAuthorization
 /batcave	Path	GET	GothamCityAuthorization

## Step 5 - Deploy the configuration

To push the configuration changes to the API Gateway instance, you must deploy the configuration. In Policy Studio click the **Deploy** button on the right of the toolbar. Alternatively, press the **F6** key.

After the configuration is successfully deployed you can test the configuration. For more information, see [Test API Gateway and Axway PassPort configuration on page 26](#).

---

# Test API Gateway and Axway PassPort configuration 4

To test the configuration of API Gateway and Axway PassPort you can use the Curl testing utility.

If API Gateway is running on a machine called `apigateway` on the default port of `8080`, to send a POST request to the authorization policy on the API Gateway using HTTP basic authentication, enter the following command:

```
> curl -X POST --user batman:password --data "test=data"
http://apigateway:8080/batcave

User 'batman' was authorized successfully!
```

This is batman attempting to modify the batcave. Batman has the `Modify Lair` privilege and is successfully authorized.

If robin attempts the same request he is rejected:

```
> curl -X POST --user robin:password --data "test=data"
http://apigateway:8080/batcave

User 'robin' was NOT authorized to access the resource successfully!
```

However, if robin attempts a GET request then he is successful as he has the `View Lair` privilege:

```
> curl -X GET --user robin:password http://apigateway:8080/batcave

User 'robin' was authorized successfully!
```