



CA SiteMinder Integration Guide

Copyright © 2013 Axway Software S.A.

Published: Nov 2013

Applies To

Axway API Server v7.x

Feedback

Send suggestions and comments about this document to support@axway.com.

Contents

[Introduction](#)

[SiteMinder Configuration](#)

[Starting the SiteMinder Server](#)

[Starting SiteMinder Administrator](#)

[Configuring SiteMinder](#)

[Step 1: Creating the Host Configuration Object:](#)

[Step 2: Creating an Agent Configuration Object:](#)

[Step 3: Creating a SiteMinder Realm:](#)

[Step 4: Creating a Policy for the Domain:](#)

[Setting up the Protected Resource](#)

[Configuring the API Server](#)

[Register the SiteMinder Agent](#)

[Registering the agent using Policy Studio:](#)

[Create the Routing Policy to the Protected Resource](#)

[Create the SiteMinder Authentication and Authorization Policy](#)

[Retrieving a SiteMinder Session Token from the Message Header](#)

[Configure a Routing Policy that will Add a SiteMinder Token to the HTTP Headers of the Response](#)

[Configuring the Policy that will Validate SiteMinder Session Tokens located in the HTTP Headers of a Request](#)

[Conclusion](#)

Introduction

This document describes how to configure the API Server to authenticate and perform authorization via CA SiteMinder R12 Web Access Manager. CA SiteMinder is a centralized Web access management system that provides user authentication and single sign-on, policy-based authorization, identity federation, and auditing of access to Web applications and portals.

This will be demonstrated by the following:

- The Server will be configured to authenticate a client via CA SiteMinder via user name and password from either HTTP Basic or a WS-Security username token.
- Upon successful authentication the Server will authorize the user for a particular resource via CA SiteMinder.
- The Server will be also be configured to add a SiteMinder session token into the HTTP headers or body of the message for authorization without having to authenticate repeatedly via CA SiteMinder.
- A policy will be created that will look for a SiteMinder session token and make a decision depending on whether a SiteMinder session token is present or not. When no token is found in the message or HTTP headers, the user will be authenticated using Basic HTTP via CA SiteMinder. If the session token is found in either the message header or body, the user need not be authenticated again and upon successful validation, the user will be granted access to the web resource.

Sections in this guide:

- Section 1 is an introduction and overview of what this document contains.
- Section 2 explains the setup and configuration of CA SiteMinder.
- Section 3 explains the setup of the resource protected by SiteMinder which is a sample service that ships with the Server.
- Section 4 explains the agent configuration required for by the Server to successfully connect to CA SiteMinder and the basic policy for authentication and authorization. This section also contains a test of the policy once configured with API Tester.
- Section 5 explains the configuration of a policy that will check for an existing SiteMinder session token in the message header. This section also contains a test of the policy once configured with API Tester.
- Section 6 explains the configuration of a policy that will check for an existing SiteMinder session token in the message body. This section also contains a test of the policy once configured with API Tester.

Setup used for this Guide:

- API Server 7
- CA SiteMinder Web Access Manager R12 connected to an iPlanet LDAP directory

SiteMinder Configuration

Starting the SiteMinder Server

The basics of setting up a SiteMinder Policy will be explained below. However it is strongly recommended that the SiteMinder user guide be consulted.

A copy of SiteMinder can be obtained from CA (www.ca.com).

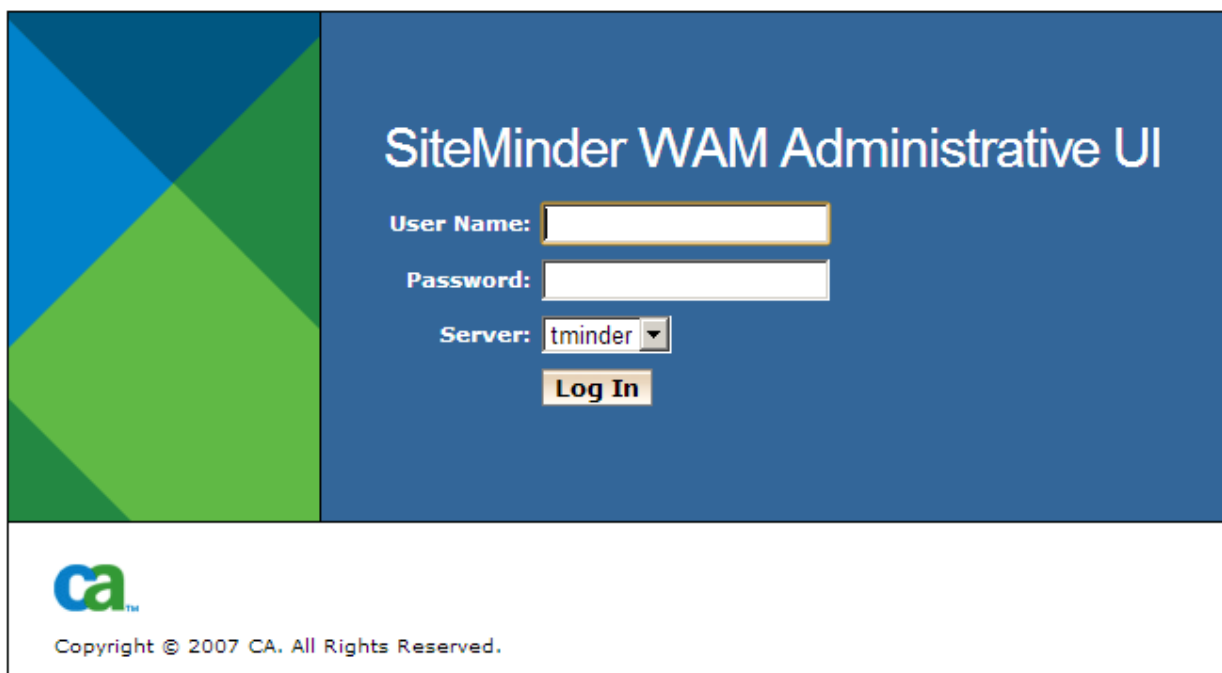
- Starting CA Policy Server on Windows:
- Open **"CA"** program group.
- Select **"IAM Suite"** and then select **"siteminderWAM"**.
- Click on **"Start Task Engine"**.
- Again from the **"CA"** program group, click on **"SiteMinder Policy Server Management Console"**.

Starting SiteMinder Administrator

The SiteMinder WAM Administrator User Interface is used to configure SiteMinder. Log into the Administrator console by using the administrator account credentials.

To log into the SiteMinder WAM Administrative User Interface:

- Open **"CA"** program group.
- Select **"IAM Suite"** and select **"siteminderWAM"**.
- Click on **"SiteMinder Administrative User Interface"**.




SiteMinder WAM Administrative UI

User Name:

Password:

Server:



Copyright © 2007 CA. All Rights Reserved.

SiteMinder Portal Login Screen

Configuring SiteMinder

Step 1: Creating the Host Configuration Object:

- In the SiteMinder Administrative User Interface, click on the **"Infrastructure"** tab.
- Select **"Hosts"** and click on **"Host configuration"**.
- Then click on **"Create Host Configuration"**.
- In the next screen select the **"Create a new object of type Host Configuration"** radio button option.
- Supply a **"Name"** and **"Description"** in the next screen for the Host. For this guide **"V6HostConfObject"** is used for the agent name.
 - Also click on **"Add"** under the Configuration Values section.
- Under the **"Host"** option enter the IP Address of the SiteMinder Policy server.
- The rest of the options can be left as default.
- Click on **"Submit"** to create the agent.

Step 2: Creating an Agent Configuration Object:

- In the SiteMinder Administrative User Interface, click on the **"Infrastructure"** tab.
- Select **"Agents"** and click on **"Agent configuration"**. For this guide **"V6AgentConfObject"** is used for the agent name.
- Then click on **"Create Agent Configuration"**.
- In the next screen select the **"Create a new object of type Agent Configuration"** radio button option.
- Supply a **"Name"** and **"Description"** in the next screen for the agent.
- Also click on **"Add"** under the Parameters section.
- For the **"Name"** field type in **"AgentName"**.
- For the **"Value"** field type in the name of the agent. In this case **"Serveragent"** is used for this guide.
- Click on **"Submit"** to create the agent.

Step 3: Creating a SiteMinder Realm:

- In the SiteMinder Administrative User Interface, click on the **"Policies"** tab.
- Select **"Domains"** and click on **"Realm"**.
- Then click on **"Create Realm"**.
- Select a domain from the Domains list before clicking **"Next"** and defining the Realm.
- Supply a **"Name"** and **"Description"** in the next screen for the Realm.
- The Agent that was created previously will now be selected for the agent. Click on the browse button next to the field to browse to the previously created Agent.
- For the **"Value"** field type in the name of the agent.
- The **"Resource Filter"** option specifies the path of the protected resource (web service). For purpose of this guide **/axis/** will be used. This means that all resources after **/axis/** is protected. This is the URI of the sample service that ships with the Server.

- Default Resource Protection needs to be set to **“Protected”**.
- **“Authentication Scheme”** needs to be set to **“Basic”**.
- Under the **“Rules”** section click on the **“Create”** tab.
- Supply a **“Name”** and **“Description”** in the next screen for the Rule.
- Under the **“Attributes/Realm and Resource”** section leave the resource as * (will apply to all resources after **/axis/**). (Setting the **“Resource”** to **/axis** implies that SiteMinder will explicitly enforce for that particular resource).
- Under the **“Action”** section select the **“Web Agent actions”** option and select all actions on the Action box.
- Click on **“OK”**.
- Also add **“Authentication events”** and **“Authorization events”** leaving the default actions.
- There will now be a total of three rules under the **“Realm”**.

NOTE: For step 4 if no domains have been created then a domain needs to be configured before creating the Realm. Settings like the LDAP directories for users, for example, are specified when creating the domain. Please consult the SiteMinder documentation on how to set up domains in SiteMinder.

Step 4: Creating a Policy for the Domain:

- In the SiteMinder Administrative User Interface, click on the **“Policies”** tab.
- Select **“Domains”** and click on **“Policy”**.
- Then click on **“Create Policy”**.
- Select the Domain specified in the list the same as the one associated with the Realm created before.
- Supply a **“Name”** and **“Description”** in the next screen for the Policy.
- Click on **“Next”**.
- Under the **“Local LDAP Directory”** section click on **“Add Member”**.
- When the LDAP connection has been setup correctly under the Domain configuration, there will be a LDAP directory reference that can be selected.

NOTE: For more information on SiteMinder configurations and implementations please refer to the SiteMinder documentation and/or your information technology administrator. The configuration described in the document is a basic SiteMinder Policy configuration and is only used for demonstrative purposes.

Setting up the Protected Resource

The Server ships with an axis sample service. For the purpose of this demonstration it is this service that is protected by SiteMinder.

Details of Sample Service:

- The axis service can be started by running the `axissimpleserver.bat` file located in the `/Server_Install_Dir/win32/bin` for Windows or `axissimpleserver.sh` located in the `/Server_Install_Dir/posix/bin` for Unix/Solaris

- For the purpose of this guide the axis service will be run on the local machine on **port 7070** which is the default port for this service.
- The URL for the resource is: **/axis/services/urn:xmletoday-delayed-quotes**
- The full URL for the service will be: http://host_ip:7070/axis/services/urn:xmletoday-delayed-quotes
- The WSDL for the available services can be retrieved from: http://host_ip:7070/axis/services/urn:xmletoday-delayed-quotes?wsdl
- The Service home page is: http://host_ip:7070/

Configuring the API Server

Configuring the Server to interact with CA SiteMinder consists of the following:

- Register the SiteMinder Agent.
- Create the routing policy to the protected resource
- Create the SiteMinder Authentication and Authorization policy.
- Create the relative path for the protected resource connected to the SiteMinder Authentication and Authorization policy.

Install the Siteminder jars on the API Server

1. Once downloaded, copy the siteminder jar file(s) smjavaagentapi.jar from the sdk/java/ or sdk/java64/ directory to the following location:
`apiserver/system/lib`
2. Once the file(s) have been copied ensure to restart the API Server for the driver to be picked up.

Smregghost Utility

The Server doesn't ship with the smregghost binaries so you need to obtain the binaries from your Siteminder installation and place them into the appropriate directory on the Server installation.

Windows

Copy all the files from the Siteminder\bin directory smregghost.exe, smcommonutil.dll & smerrlog.dll to
`$apiserver_intall_dir/win32/lib`

Also copy all the dll files from \Program Files\CA\sdketpki\ETPKI\lib\ to
`$apiserver_intall_dir/win32/lib`

They required etpki files may also be in the following location if not in the above
`\Program Files\CA\SOA Security Manager\siteminder\etpki-install\lib`

Linux 64 bit /32 bit & Solaris

Obtain a copy of the siteminder SDK for your platform. All the necessary files are contained within the bin directory for the specific platform, all the files can be copied to the `apiserver/platform/lib` directory. For 64 bit systems copy the contents of the `sdk/bin64` to the `apiserver/platform/lib` directory, for 32bit systems copy the contents of `sdk/bin` `apiserver/platform/lib`.

Registering the Agent can be done on two ways:

1. Using the “smreghost” utility via the command line
2. Via Policy Studio

Register the SiteMinder Agent

In order to act as a PEP (Policy Enforcement Point) for CA SiteMinder, the Server must have been set up as a *SOA Agent* with the SiteMinder Policy Server. Therefore the agent needs to be registered on the machine running the Server.

IMPORTANT: The agent needs to be registered on the machine running the Server, so in cases where the Policy Studio is remotely connecting to the Server for example an Appliance; the “smreghost” utility will be used to register the agent configuration via the command line.

Using command line:

For Unix/Solaris

- `cd /opt/Axway/apiserver/platform/lib`
- `export LD_LIBRARY_PATH=/opt/Axway/apiserver_install_dir/platform/lib`
- Run the smreghost command as in the following example:
- *Example:*
`smreghost -i 192.168.0.99 -u SiteMinder -p XXXXXX -hc V6HostConfObject -hn Serveragent`

For Windows:

- `cd $apiserver_intall_dir/win32/lib`
- Run the smreghost command as in the following example:
- *Example:*
`smreghost -i 192.168.0.99 -u SiteMinder -p XXXXXX -hc V6HostConfObject -hn Serveragent`

SMRegHost Usage:

```
./smreghost -i ipAddress[:port] -u username -p password -hn hostname -hc hostconfigobject
```

```
-i <ipAddress[:port]>  
-hn <Name for host to be registered>  
-hc <Name of host configuration object>  
-sh <Shared secret for the host>  
-rs (enable shared secret rollover for host)  
-u <Administrator username>  
-p <Administrator password>  
-f <File to store registration data in (defaults to ./SmHost.conf)>  
-cp <Name of crypto provider (BSAFE or PKCS11)>  
-cd <Path to crypto provider DLL or config file>  
-ct <Crypto provider token label>  
-ck <Crypto provider token PIN>
```

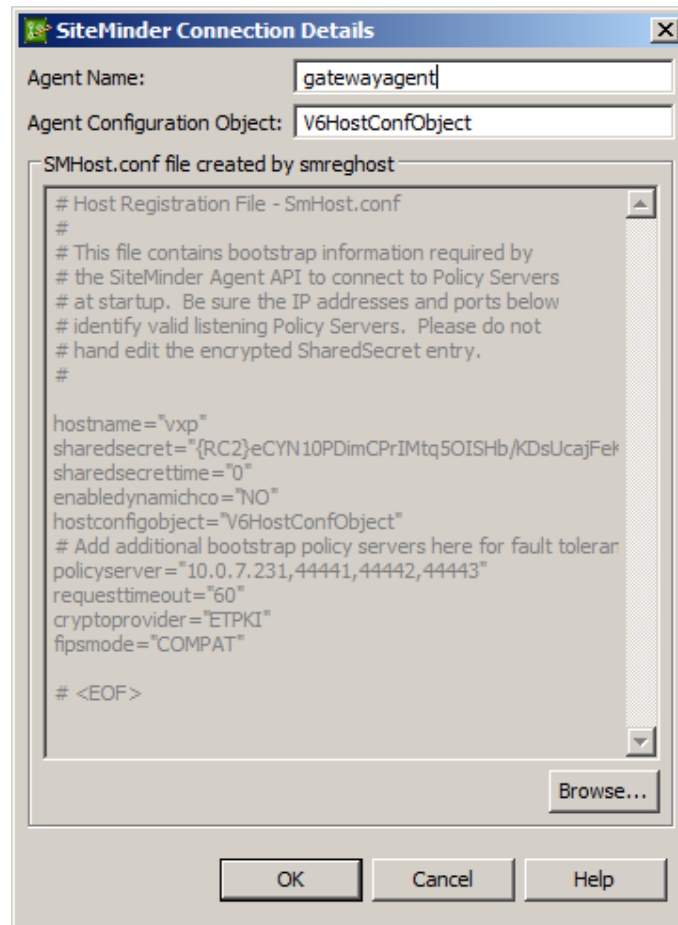

-o <Overwrite existing Trusted Host >

- Once the command has run successfully, a file called **"smhost.conf"** will be created in the /opt/Axway/apiserver_install_dir/Linux.x86_64/bin folder.
- Browse to the VXP Web Administration Interface on https://appliance_IP:10000/.
- Click on the **"Upload and Download"** option and download the **"smhost.conf"** file to the machine running the Policy Studio.
- Finally, use the **"Browse"** when configuring the agent in Policy Studio as described below instead of the **"Register"** option to import the configuration.

Registering the agent using Policy Studio:

- Start Policy Studio by running **"policystudio.exe"** (Windows) or **"policystudio.sh"** (Unix/Solaris) from the Policy Studio root directory.
- Click on the URL for the Server or Policy Director if the Server/s is managed via Policy Director.
- Click on the Server process listed to open the configuration window in a new tab.
- Click on the **"External Connections"** module.
- Right click on **"SiteMinder/SOA Security Manager Connections"** and select **"Add SiteMinder Connection"**.
- Enter the **"Name of the host configuration object"** as is configured in section 2.3 Configuring SiteMinder step1. For this guide it is **"V6HostConfObject"**.
- Click browse and select the **"smhost.conf"** already created.

The Register Host window for SiteMinder in Policy Studio



SiteMinder agent configuration

- Once all the details have been provided and click on **"OK"** and a dialogue will appear showing that the agent has been registered successfully.
- Expand the **"Authentication Repository Profiles"** object under the **"External Connections"** tree.
- Right click on **"CA SiteMinder Repositories"** and select **"Add a new Repository"**.
- Enter a name for the repository. For this guide **"SiteMinder"** is used.
- Select the configured agent in the drop down menu for the **"Agent Name"** value.
- The rest of the options can be left default. Click on the help button if more details are required for the configuration options here.

The SiteMinder Authentication Repository for screen in Policy Studio:

Siteminder Repository Configuration

Create the Routing Policy to the Protected Resource

- In the Policy Studio configuration window for the Server process click on the **“Policies”** module.
- Right click on **“Policies”** at the top node and select **“Add Container”**.
- Name the container **“Routing”**.
- Right click on **“Routing”** container and select **“Add Policy”**.
- Name the policy to **“Route to Axis Service”**.
- Drag a **“Connect to URL”** filter from the **“Routing”** category.
- Rename the Name of the filter to: **Route to Axis Service**
- Enter URL: **http://host_ip:7070/axis/services/urn:xmltoday-delayed-quotes**
- Click on **“Finish”**.
- This policy will be called to via a **“Policy Shortcut”** filter that will be configured in the next section as part of the SiteMinder Authentication and Authorization policy.

Create the SiteMinder Authentication and Authorization Policy

The policy is going to authenticate and authorize a user that exists in an LDAP directory server via SiteMinder.

The flow of the policy:

- User will present a username and password combination to the Server using HTTP Basic authentication. The user's credentials will be passed to CA SiteMinder, which will authenticate the user.
- The user will be authorized via CA SiteMinder for a particular resource
- The request will be routed to the resource (axis web service) via the policy shortcut that will call the **“Route to Axis Service”** routing policy.

Step 1: Configure a HTTP Basic Authentication Filter

- In the Policy Studio configuration window for the Server process click on the **"Policies"** module.
- Right click on **"Policies"** and select **"Add Policy"**. Name the policy **"SiteMinder"**.
- Click on the policy and add a **"HTTP Basic"** filter located in the **"Authentication"** filter category located on the right hand side of Policy Studio.
- **HTTP Basic Filter Configuration:**
- **Name:** of the filter can be left default or changed to any descriptive name.
- **Realm:** Populated automatically by the value specified in System Settings in the Server.
- **Credential Format:** select User Name from the drop down list.
- **Repository Name:** Select **"SiteMinder"** repository from the drop down field.
- Click on **"OK"**

Step 2: Configure a SiteMinder Authorization Filter

- Drag an **"Authorization"** filter from the **"CA SiteMinder"** filter category.
- Name the filter **"Authorize User via SiteMinder"**.
- Settings can be left as default.
- Click on **"Finish"**.
- Connect the **"HTTP Basic"** filter configured in step 1 to the **"CA Authorization"** filter to the with a success path.

Step 3: Configure a Policy Shortcut to call point to Route to Web

- Drag a **"Policy Shortcut"** filter from the **"Utility"** filter category.
- Choose the **"Route to Axis Service"** policy that will route the message to the Axis web service.
- Click on **"Finish"**.
- Connect the **"CA Authorization"** to the **"Policy Shortcut"** filter with a success path.

Step 4: Create a New Relative Path

- Click on the **"Services"** module in Policy Studio.
- Expand **"Listeners"**, **"API Server"** and **"Default Services"**.
- right click on the **Paths**
- Select **"Add Relative Path"** and enter: **/axis/services**
- Deploy the new configuration to the server by pressing **"F6"** on the keyboard or by clicking on **"Settings"** at the top and select **"Deploy"**.

Step 5: Test Policy with APITester

API Tester is a free stress and security tool for web services. It will be used as the client for testing.

API Tester can generate a request via the available web service:

- - Start **API Tester** by running **"apitester.exe"** (win32) or **"apitester.sh"** (UNIX) located in the APITester root directory.
- Click on the **"Import WSDL"** button on the top tool bar.
- Select the **"WSDL URL"** option and enter the URL of the Axis Service WSDL.

- If running the Axis Service locally as described in section 3 the URL is: <http://localhost:7070/axis/services/urn:xmltoday-delayed-quotes?wsdl>
- Click on “OK”.
- A request will be automatically generated.

Alternatively the following request can be used:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soap:Body>
    <ns:getQuote
      soap:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
      xmlns:ns="http://stock.samples" />
    </soap:Body>
  </soap:Envelope>
```

- Configure the URL that the request needs to be sent to by clicking on the configuration block underneath the top menu.
- Enter the URL for the XML Server and resource path. In this case it is:
- http://Server_host_ip:8080/axis/services
- Copy the test message above into the Soap Request window if it has not been autogenerated already.
- Click on “Security” tab followed by the “HTTP Authentication” tab.
- Choose ‘HTTP Basic’ and enter the username and password of the user will be authenticated via SiteMinder. For purpose of this guide it is a user called Hubert Farnsworth that is located in the LDAP directory that SiteMinder is connected to.
 - Username: cn=hubert farnsworth,o=planet express,l=new york,st=ny,c=us
 - Password: goodNews
- **NOTE:** The username in this case is actually the user’s distinguished name (DN) in the LDAP directory.
- Click on “Run” to send the message.
- The request will be sent via the Server that in turn will contact CA SiteMinder to authenticate and authorize the user.
- Once successfully completed the Server will send the request on to the protected resource.

Retrieving a SiteMinder Session Token from the Message Header

In the previous section a policy has been configured to authenticate and authorize the request.

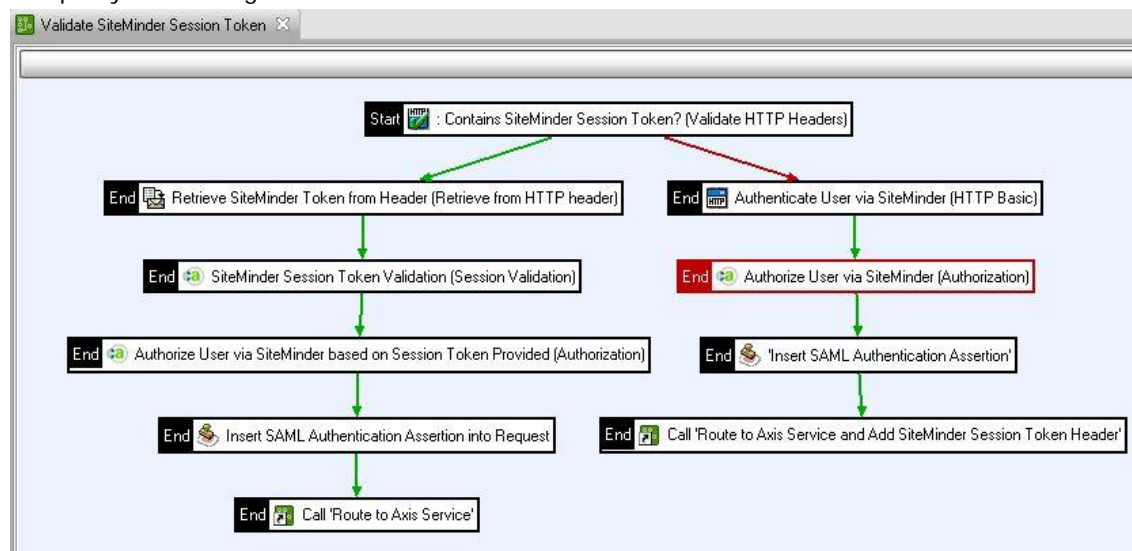
In this section a policy will be created that will look for an existing SiteMinder session token in the message headers. When a SiteMinder session token is found in the message (header or body); the user does not have to be authenticated again.

In a real world deployment scenario, it is not practical to have to authenticate and authorize each user for every request they send. By inserting a SiteMinder session token into a message after successfully authenticating a user the token in the message is validated instead of re-authenticating the user for every following request. A SAML Authorization Assertion will also be inserted for consumption by a downstream web service.

The flow of the policy will be as follows:

- Check the request's HTTP headers for the presence of a SiteMinder session token.
- When the token is present, the "Retrieve from HTTP Header" filter is run to retrieve the value of the token from the "siteminder.session" attribute.
- When the attribute containing the token has been retrieved successfully the session will be validated using a Ca SiteMinder "Session Validation" filter to check if the user token is still valid.
- When the token has been successfully validated then the user will be authorized using the CA SiteMinder "Authorization" filter.
- A SAML Authentication Assertion will be inserted into the message for consumption by downstream web services.
- The message will then be routed to the Axis service.
- If the Message did not have the "siteminder.session" token in the header and passed through the "HTTP Basic" filter the user will be authorized based on user credentials.
- The user will be authorized using the CA SiteMinder "Authorization" filter.
- The message will pass through a "SAML Authentication Assertion" filter and inserted into the message for consumption by downstream web services.
- The message will then be passed to the "Route to Axis Service" policy shortcut that will route the message to the protected resource. The response message will pass through an "Add HTTP Header" filter that will add a HTTP Header to the message response containing a SiteMinder session token.

The policy once configured:



SiteMinder policies

Configure a Routing Policy that will Add a SiteMinder Token to the HTTP Headers of the Response

The first step will be to create a new routing policy that will add the SiteMinder session token into the HTTP headers of the response request.

Creating the “Route to Axis Service and Add SiteMinder Token to Header” Policy:

- Navigate to the “Route to Axis Service” policy as configured in section 4.2.
- Right click on the “Route to Axis Service” on the left and click on “Copy”.
- Right click on the “Routing” container and select “Paste”
- Rename the policy to: **Route to Web Service and Add SiteMinder Token to Header**.
- Click on “OK”.
- Add an “Add HTTP Header” filter from the “Conversion” category.
- Name the filter “Add SiteMinder Session Token into Response Header”.
- For the “HTTP Header Name” type “SiteMinderToken”.
- For the “HTTP Header Value” type `${siteminder.session}`. This is the attribute value that is specified by the SiteMinder Authentication repository configuration as configured in section 4.1. When a user is successfully authenticated to SiteMinder, this message attribute will contain the user’s SiteMinder session token.
- Click on “Finish”.
- Connect the “Route to Axis Service” filter to the “Add SiteMinder Session Token into Response Header” filter via a success path.

The routing policy will now look like below:



Routing Policy

Configuring the Policy that will Validate SiteMinder Session Tokens located in the HTTP Headers of a Request

Step 1: Configure a “Validate HTTP Headers” filter to check if a SiteMinder Session Token is present in the HTTP Headers of the Message.

- Click on the “Policies” module in the tree on the left hand side of Policy Studio.
- Right click and select “Add Policy”.
- Name the Policy “Validate SiteMinder Session Token in HTTP Headers”.
- Drag a “Validate HTTP Headers” filter from the “Content Filtering” category.

- Name the filter **"Contains SiteMinder Session Token?"**
- Under the **"Enter a Regular Expression"** section click on the **"Add"** button.
- A Window titled **"Configure Regular Expression"** will open. Enter **"SiteMinderToken"** in the name field and select the **"Required"** radio button.
- Click on **"Ok"**.
- Right click on this filter and select **"Set as Start"**.

Step 2: Configure a "Retrieve from Attribute" filter that will retrieve the existing token from the HTTP Header of the message.

- The **"Retrieve from HTTP Header"** filter is located in the **"Attributes"** group.
- Name the filter **"Retrieve SiteMinder Token from Header"**.
- In the **'HTTP Header Name'** field enter **"SiteMinderToken"**. In the **'Attribute ID'** field enter **"siteminder.session"**.
- Click on **"Finish"**.
- Connect the **'Validate HTTP Header'** filter to the **"Retrieve from HTTP Header"** filter with a success path.

Step 3: Configure a SiteMinder "Session Validation" filter for validation of the SiteMinder Session Token.

- Add a **"Session Validation"** filter located in the **"CA SiteMinder"** category.
- For the **"Agent Name"** select the SiteMinder agent that was created in section 4.1 from the drop down field. For this guide it is **"Serveragent"**.
- Ensure that the **"Message Attribute Containing Session"** is set to **"siteminder.session"**.
- The rest can be left default.
- Click on **"Finish"**.
- Connect the **"Retrieve from HTTP Header"** filter to the **"Session Validation"** filter with a success path.

Step 4: Configure a "Authorization" filter that will Authorize the Client for the Protected Resource.

- Drag an **"Authorization"** filter from the **"CA SiteMinder"** category.
- Name the filter **"Authorize User via SiteMinder based on Session Token"**.
- Settings can be left as default.
- Click on **"Finish"**.
- Now connect the **"Session Validation"** filter to the **"Authorization"** filter with a success path.

Step 5: Configure an "Insert SAML Authentication Assertion" filter that will add a SAML Authentication Assertion to the Message for Consumption by Downstream Service. (This is done for demonstrative purposes only and is not necessary for purpose of this guide)

- The **"Insert SAML Authentication Assertion"** filter is located in the **"Authorization"** category.
- **Expiry Date:** Set to any desired value.
- **SOAP Actor/Role:** Choose **"Current Actor/Role Only"** from the drop down list.
- On the **"Sign Assertion"** Tab select **"No Signature with Assertion"** and select any value from the drop down field for **"Issuer Name"**.
- Under **"Advanced Options"** tick **"Insert SAML Attribute Statement"** and **"Indent"**.

- The rest of the options could be left default.
- Click on **“Finish”**.

Step 6: Configure a “Policy Shortcut” filter and link it to the “Route to Axis Service” policy.

- Drag a **“Policy Shortcut”** filter from the **“Utility”** category.
- Choose the **“Route to Web Service and Add SiteMinder Token to Header”** policy that will route the message to the sample web service and add the session token to the header of the response.
- Click on **“Finish”**.
- Connect the **“Policy Shortcut”** filter to the **“Insert SAML Authentication Assertion” filter** with a success path.

Conclusion

This document outlines a simple demonstration to configure API Server to authenticate a user and authorize their request for a particular resource against an CA Siteminder.

This configuration can be part of a larger policy, including features such as XML threat detection and conditional routing, features which are out of the scope of this document but are covered in other documents which can be obtained from the Axway Support Portal at <http://support.axway.com>.