IBM Security Access Manager
Version 9.0.6
November 2018

Federation Administration topics

IBM Security Access Manager
Version 9.0.6
November 2018

Federation Administration topics

IBM

Contents

Figures

Tables

Chapter 1. Managing federations

Use the local management interface to manage federations.

Ensure that you activate the Federation Module to use the federation features.

For information about establishing a federation, including creating a federation and adding a partner to an existing federation, see the instructions for your federation type:

- SAML Federations Overview
- WS-Federation federations
- OpenID Connect federations

Creating and modifying a federation

Use the Federations management page to create a new federation, or to view and modify the details about an existing federation.

Before you begin

Depending on the protocol you want to use, review the following topics:

- SAML 1.1
- SAML 2.0
- WS-Federation federations
- OpenID Connect federations

  Note: Do not use the Federation management page for new OpenID Connect Providers. New OIDC Provider federations are now managed through the API Protection page in the local management interface. See OpenID Connect Provider federations.

  Security Access Manager Version 9.0.4 supports enhanced features for OpenID Connect. The configuration and management tasks for new Providers and Relying Parties are enhanced. These tasks replace the management tasks for prior (Version 9.0.3 and older) OpenID Connect Provider Federations.

  The Version 9.0.3 and older federations remain fully supported as legacy federations. The legacy management tasks remain fully supported through the Federation management page.

  Review the information for your type of OpenID Connect federation:

  – For new OpenID Connect Providers, see OpenID Connect Provider federations.

  – For new OpenID Connect Relying Party federations, see OpenID Connect Relying Party federations

  – For existing legacy OpenID Connect Providers and Relying Party federations, see Legacy support for OpenID Connect federations.

Procedure

1. Log in to the local management interface.
2. Select **Secure Federation** > **Manage** > **Federations**. All configured federations are displayed.

3. You can create a federation or modify any existing federations.
   - To create a federation, click **Add** and then follow the wizard. The wizard pages differ depending on the federation protocol you select.

     Note: If you encounter a session timeout while you are creating a federation and then log back in, you might not be able to see the federation that you created. You must click **Refresh** to get the current data from the appliance.
   - To modify a federation, select the federation and then click **Edit**. Follow the wizard and modify the settings on each page as needed.

Exporting a federation

When you want to join a federation hosted by another business partner, you must supply your federation configuration properties. You can export your SAML 2.0 federation properties to a file to share them with your partner.

Procedure

1. Log in to the local management interface.
2. Click **Secure Federation** > **Manage** > **Federations**.
3. Select a federation from the table.
4. Click **Export**. The browser shows a message window that prompts you to save the file containing the exported data.

   Note: Exporting a federation is applicable to SAML 2.0 federation only.
5. Click **OK**. The browser download window prompts for a location to save the file.
6. Select a directory and metadata file. Metadata file names have the following syntax:

   federationname`_metadata.xml`

   For example, for a federation named `federation1`, and a company named `ABC`, the metadata file would be named:

   `federation1_ABC_metadata.xml`

   Note: Place the metadata file in an easily accessible location. You must provide the file to your partner, when your partner wants to import configuration information for the federation.
7. Click **Save**.

Deleting a federation

Use the Federations management page to delete the federation that you no longer need.

About this task

Attention: When you delete a federation, all of its partners are also deleted.

Procedure

1. Log in to the local management interface.
2. Click **Secure Federation** > **Manage** > **Federations**. The Federation Management panel shows a list of configured federations.

3. Select a federation.
4. Click **Delete** to delete the federation. A message box prompts you to confirm the deletion of the federation.
5. Click **YES** on the message box. The federation is deleted.

Managing federation partners

Use the Federations management page to create, modify, delete, enable, or disable your federation partners.

Procedure

1. Log in to the local management interface.
2. Select **Secure Federation** > **Manage** > **Federations**. All existing federations are displayed in the list.
3. Select the federation to manage partners for.
4. Click **Partners**. All existing partners for this federation are displayed.
5. You can create, modify, delete, enable, or disable your federation partners.
   - To create a partner, click **Add** and follow the wizard. The wizard pages differ depending on the federation protocols. When there are multiple partner templates to choose from, they are displayed in a table. You can use the filter to locate a particular partner template and then select it from the table.

     Remember: For a SAML 2.0 federation, the partner create wizard is a two-stage process. The first stage is to upload the metadata file for the partner. After this stage is complete, a new partner that matches the details in the metadata file is created. The second stage is to complete the wizard to update some properties in the new partner. Even if the wizard is canceled after the metadata upload, the new partner is still created. The upload operation creates the partner. For information about adding a partner to an existing federation, see Creating a SAML 2.0 partner.

     Note: If you encounter a session timeout while creating a partner and then log back in, you might not be able to see the partner you created. You must click **Refresh** to get the latest data from the appliance.
   - To modify a partner, select the partner and click **Edit**. Follow the wizard and modify the settings on each page as needed.
   - To delete a partner, select the partner and click **Delete**. Confirm the deletion by clicking **Yes**.
   - To enable or disable a partner, select the partner and click **Enable** or **Disable**.
   - To refresh the partners page, click **Refresh**.

Chapter 2. Managing federation partner templates

Use the Federation Partner Templates management page to see the current version of the templates on the appliance and update the templates to a new version.

About this task

Federation partner templates are designed to simplify the configuration of federations within the appliance. These templates contain partner definitions that can assist with the establishment of federations to well-known partners.

When you activate the Federation Module, the federation partner templates package that was included in the firmware is automatically applied. You can also update the existing templates to a new version by importing a new templates package into your appliance.

In a cluster environment, after you import a new templates package into the primary node appliance, the update is automatically applied to non-primary nodes.

Note: Templates package update is possible from the primary node only.

Procedure

1. Log in to the local management interface.
2. Select **Secure Federation** > **Global Settings** > **Partner Templates**. The version number of the current federation partner templates and the installation date are displayed.
3. To update the templates, click **Import**.
4. Click **Browse** and then select the new templates package file.
5. Click **Import** to upload the selected templates package file to the appliance.
6. Deploy the changes.

Chapter 3. Managing attribute sources

Use the Attribute Source management page to add, edit, or delete your identity attribute sources.

About this task

You can manage the following types of attribute sources with this UI:

Fixed
This type contains the **Attribute Name** and **Value** fields. Both fields are in free text format. You can specify any text in these fields to suit your needs.

Credential
This type contains the **Attribute Name** and **Credential Attribute** fields. For the **Credential Attribute** field, you can select from a list of commonly used credential attribute values or add a value that is not already in the list.

Note: The Credential attribute source does not work for the Relying Party in an OpenID Connect federation, because when the mapping occurs the user does not have the credential from which to retrieve the attribute.

LDAP
This type contains the attribute name and the details of the LDAP server to look up the attribute in. The following fields are available:

Attribute Name
Name of the attribute on the appliance. This field is required.

LDAP Attribute
Name of the attribute on the LDAP server. This field is required.

Server Connection
The ID of the existing LDAP server connection that contains information about the location and the credential that is required to connect to the LDAP server. This field is required.

Note: To add an LDAP attribute source, there must be at least one LDAP server connection present. For details about how to create an LDAP server connection, see Chapter 6, "Managing server connections," on page 25.

Scope
The scope of the search. Valid values are `Subtree`, `One level`, and `Base`. This field is optional.

Selector
A comma-separated list of the attributes to be retrieved from the search result. When multiple attributes are required from the same search result, you can use the selector to include all the required attributes. For example, `"cn,sn,mobile,email"`. This field is optional.

Search Filter
The search filter to use for the search. You can use a variable macro that will be replaced during the run time before the search. The macro will be replaced with a value from the STSUU attributes. If the value is not found, it will not be replaced. The macro is indicated by curly brackets. For example, `"(cn={AZN_CRED_PRINCIPAL_NAME})"`. This field is required.

BaseDN

The base DN to run the search on. You can use a variable macro that will be replaced during the run time before the search. The macro will be replaced with a value from the STSUU attributes. If the value is not found, it will not be replaced. The macro is indicated by curly brackets. For example, `"dc=iswga"` `or"dc={myBaseVariable}"`. This field is required.

Procedure

1. Log in to the local management interface.
2. Click **Secure Federation** > **Manage** > **Attribute Source**.
3. You can create, modify, or delete attribute sources.

   Creating an attribute resource

   a. Click **Add** and select the type of attribute source to create.
   b. Provide details for the attribute source.
   c. Click **Add**.
   d. Deploy the changes.

   Modifying an attribute source

   a. Select the attribute source to modify.
   b. Click **Edit**.
   c. Edit the details of the attribute source as needed.
   d. Click **Modify**.
   e. Deploy the changes.

   Deleting an attribute source

   Note: Before deleting an attribute source, ensure that the attribute source is not used by any federations or partners. Deleting an attribute source that is used by a federation or partner could cause failure of single sign-on flows.

   a. Select the attribute source to delete.
   b. Click **Delete**.
   c. Click **Delete** to confirm the deletion.
   d. Deploy the changes.

Chapter 4. Point of contact profiles for Federation

Use the local management interface to work with your point of contact profiles.

You can perform the following point of contact profile tasks:

- "Creating a point of contact profile"
- "Updating or viewing a point of contact profile" on page 10
- "Deleting a point of contact profile" on page 10
- "Setting a current point of contact profile" on page 11

Creating a point of contact profile

Create a point of contact server profile to capture the information needed for the runtime to communicate with the point of contact server.

About this task

You can create point of contact profiles with the Federation module or the Advanced Access Control module.

Three point of contact profiles provided by Security Access Manager are ready for use.

When you want to create your own profile that is similar to an existing one, use **Create Like** to save time. If you do not want to reuse any of the existing specifications, create a brand new one with **Create**. The details are in the following procedure.

Procedure

1. From the local management interface, select **Secure Federation** or **Secure Access Control**. Then, **Global Settings** > **Point of Contact**. A list of point of contact server profiles displays. The list includes three preconfigured profiles and any other custom profiles that you created.
2. Take one of the following actions:
   - Click **Create** to create a custom point of contact profile.
   - Select a profile from the list and click **Create Like** to start with values similar to an existing profile.
3. On the Profile Name page, enter the name of the profile. The first character of the profile name must be alphanumeric. The maximum number of characters is 200.
4. Optional: Enter a description.
5. Specify the parameter information:
   - Enter the information on each tabbed page, and click **Next**.
   - In the Callback Parameters section on each page, click **Create** to open a window to add a set of parameter name and value pairs. Click **Save** when complete.
   - Add as many parameters as you need. The **Value** field might be empty for some parameters.

- To delete a parameter name from the list, select the parameter and click **Delete**.

6. At the Summary page, if everything is correct, click **Finish**.
7. Deploy the pending changes.

What to do next

- See "Callback parameters and values" on page 11 for more information.
- You might want to change the current point of contact profile. See "Setting a current point of contact profile" on page 11.

Updating or viewing a point of contact profile

Update or view a point of contact server profile.

About this task

You cannot update the preconfigured point of contact profiles.

Procedure

1. From the local management interface, select **Secure Federation** or **Secure Access Control**. Then, **Global Settings** > **Point of Contact**. A list of point of contact server profiles displays.
2. Perform one of the following actions:
   - Update
     a. Select a profile from the list that is not a preconfigured profile and click **Update** to change the configuration details.
     b. Click **Next** to see each page and make updates if necessary.
     c. On the Summary page, click **Finish** to save your changes.
     d. Deploy the changes
   - View
     a. Select a profile from the list and click **Properties** to look at the configuration details without making updates.
     b. Click on each tab to see the information.
     c. Click **OK** when finished.

What to do next

See "Callback parameters and values" on page 11 for more information about the properties.

Deleting a point of contact profile

Use the local management interface to remove a point of contact profile.

About this task

You cannot delete the following profiles:

- A preconfigured point of contact profile.
- A profile that is set as the current profile. Select another profile as the current one, if necessary.

See "Setting a current point of contact profile" on page 11.

Procedure

1. From the local management interface, select **Secure Federation** > **Global Settings** > **Point of Contact** or **Secure Access Control** > **Global Settings** > **Point of Contact**. A list of point of contact server profiles displays.
2. Select a profile from the list, that is not a preconfigured profile, and click **Delete**. The details of the selected profile display.
3. Review the profile to ensure that it is the one you want to delete.
4. Click **Finish**.
5. Click **OK** to confirm.
6. Deploy the change.

Setting a current point of contact profile

Set a point of contact profile as the current one so that the federation runtime communicates with the point of contact server using the correct set of specifications.

Procedure

1. From the local management interface, select **Secure Federation** > **Global Settings** > **Point of Contact** or select **Secure Access Control** > **Global Settings** > **Point of Contact**. A list of point of contact server profiles displays. The list includes three preconfigured profiles and any other custom profiles that you created. The green dot indicates the current profile.
2. To change the current profile, select the profile you want to use as the current one and click **Set As Current**. The current profile indicator displays next to the profile you selected.
3. Deploy the changes.

Callback parameters and values

Specify the callback parameters and values when you define a point of contact profile.

Sign In callbacks

`fim.user.request.header.name`
The name of the header that contains the user name of the user.

Data type: String

Example: `iv-user`

`fim.attributes.response.header.name`
The name of the header that contains the attributes of the user.

Data type: String

Example: `am-fim-eai-xattrs`

`fim.groups.response.header.name`
The name of the header that contains the groups of the user.

Data type: String

Example: `fim.groups`

`fim.server.response.header.name`
The name of the header that contains the hostname that authenticates the user.

Data type: String

Example: `fim.server`

`fim.target.response.header.name`
The name of the header that contains the redirect URL.

Data type: String

Example: `am-fim-eai-redir-url`

`fim.user.response.header.name`
The name of the header that contains the user name of the user.

Data type: String

Example: `am-fim-eai-user-id`

`fim.user.session.id.response.header.name`
The name of the header that contains the reverse proxy session ID of the user.

Data type: String

Example: `user_session_id`

`fim.cred.response.header.name`
The name of the header that contains the IVCred of the user.

Data type: String

Example: `am-fim-eai-pac`

`url.encoding.enabled`
Indicates whether the EAI header names and values are URL encoded. The default setting for this property is `false`. The EAI header names and values are not URL encoded.

Data type: Boolean

Example: `false`

Sign Out callbacks

`fim.user.session.id.request.header.name`
The name of the header that contains the reverse proxy session ID of the user.

Data type: String

Example: `user_session_id`

`fim.user.request.header.name`
The name of the header that contains the user name.

Data type: String

Example: `iv-user`

Local ID

`fim.attributes.request.header.name`
The name of the header that contains the attributes of the user.

Data type: String

Example: `fim.attributes`

`fim.cred.request.header.name`
The header that contains the IVCred of the user.

Data type: String

Example: `iv-creds`

`fim.groups.request.header.name`

The name of the header that contains the groups of the user.

Data type: String

Example: `iv-groups`

`fim.user.request.header.name`

The name of the header that contains the user name.

Data type: String

Example: `iv-user`

Authenticate

`fim.user.request.header.name`

The name of the header that contains the user name.

Data type: String

Example: `iv-user`

`authentication.macros`

A list of macros that defines contextual information to pass to the web reverse proxy login page. The macros you specify can customize an authentication login page for a specific service provider. For more information, see Customizing the SAML 2.0 login form.

Data type: String

Example: If an identity provider wants to display the provider ID and target URL of a partner, specify the following macros:

```
%PARTNERID%,%TARGET%
```

Chapter 5. Managing trust chains

Use the Security Token Service page to achieve token conversions through the WS-Trust interface.

Managing module chains

A module chain defines a WS-Trust endpoint. It contains a reference to a template and the properties for all modules within that template.

About this task

You can define the following fields for a module chain:

The following fields are available on the **Overview** tab:

Name Name of the module chain. This field is required.

Description
Description of the module chain. This field is required.

Template
The template that is referenced by this module chain.

The following fields are available on the **Lookup** tab:

Request Type
The type of request to associate with this chain. The request is one of the types that are supported by the WS-Trust specification.

Issue Issue a new token, based on information that is obtained from the request.

Renew
Renew an expired token.

Validate
Validate the specified security token and return the requested result.

Cancel
Cancels a previously issued token so that it is no longer used.

Key Exchange
Transfer of a new key for use by the receiver of the request.

Other A custom request type.

URI A Uniform Resource Indicator for each request type. This field is read-only except when you select **Other** as the request type. For the **Other** type, enter the URI for your custom request type.

Lookup Type

XPath Select this option to enable a text field that can be used to define a custom lookup rule. Use XML Path Language to define the rule.

Traditional WS-Trust Elements

Specifies that the trust service uses the values in the request for **Applies To**, **Issuer**, and **Token Type** as input to determine which trust service module chain to call.

Select this option to show the data entry fields for **Applies To**, **Issuer**, and **Token Type**.

You must provide a value for at least one of the following fields for the Traditional WS-Trust Elements:

- Issuer address
- Applies To address
- Token Type selected, where value is not **None**

Or, if you do not specify one of the above options, then you must select the **XPath** option, plus a specified XPath.

Note: If you supply values for all of the parameters, including the XPath, then the Issuer, Applies To, and Token Type values take priority over XPath in the runtime.

Applies To

Defines the scope of the token.

Address

The address of the service that is being requested. For example:

```
http://sp.example.com:9080/TrustServer/SecurityTokenService
```

You can use the asterisk (*) wildcard character at the end of a string. For example, `token*` matches all strings that start with `token`.

You can also use regular expressions in the format of `REGEXP:(`*`regular_expression_here`*`)`. For example, `REGEXP:(.*/application.*)` matches any string that contains `/application`.

When you specify the scope for single sign-on protocols, this URI is sufficient information for identifying the issuer.

Service Name

A qualified name (QName) that includes a namespace URI and a local part for the Web service.

Note: A QName is a term that is defined in XML specifications. A QName consists of a namespace URI plus a local part.

For example:

```
http://sp.example.com:SecurityTokenService
```

Note: The **Service Name** field provides a colon (:) to separate the namespace URI from the local part.

This field is typically not used for single sign-on protocols, but can be used for Web services security management.

Port Type

Web services operations are grouped by port type. Use this field when there are more than one Web services port types to specify.

For example, a stock market data service might provide both a stock quoting service and a stock price history service. Use this field to specify the needed Web service.

```
http://sp.example.com:RequestSecurityTokenPort
```

The port type is also a QName.

Note: The **Port Type** field provides a colon (:) to separate the namespace URI from the local part.

This field is typically not used for single sign-on protocols, but can be used for Web services security management.

Issuer

Address

The URL for the company or enterprise that issued the token. For example:

```
example.com
```

You can use the asterisk (*) wildcard character at the end of a string. For example, `token*` matches all strings that start with `token`.

You can also use regular expressions in the format of `REGEXP:(`*`regular_expression_here`*`)`. For example, `REGEXP:(.*/application.*)` matches any string that contains `/application`.

When you specify the scope for single sign-on protocols, this provider ID is sufficient information for identifying the issuer.

Service Name

A qualified name (QName) that includes a namespace URI and a local part for the Web service.

Note: A QName is a term that is defined in XML specifications. A QName consists of a namespace URI plus a local part.

For example:

```
http://idp.example.com:myWebService
```

Note: The **Service Name** field provides a colon (:) to separate the namespace URI from the local part.

This field is typically not used for single sign-on protocols, but can be used for Web services security management.

Port Type

Web services operations are grouped by port type. Use this field when there is more than one Web service port type to specify.

For example, a stock market data service might provide both a stock quoting service and a stock price history service. Use this field to specify the needed Web service.

```
http://idp.example.com:getQuotePortType
```

The port type is also a QName.

Note: The **Port Type** field provides a colon (:) to separate the namespace URI from the local part.

This field is typically not used for single sign-on protocols, but can be used for Web services security management.

Token Type

Defines the STS module type. Select a type to map a request message to an STS chain. Select a type other than **None** for it to be part of the runtime flow.

URI A Uniform Resource Indicator for each token type. This field is read-only except when you select **Other** as the token type. For the **Other** type, enter the URI for your custom token type.

The following fields are available on the **Security** tab.

Validate Requests

Select to require a signature on the SOAP Body element that contains the RequestSecurityToken messages. When you select this option, your Trust Service client must use their private key to sign the body of the SOAP messages in accordance with the Web Services Security SOAP Message Security 1.1 specification.

Select the public key that was provided by the client to validate the signature on the message.

Certificate Database

Specify the database where the certificate is stored.

Certificate Label

Specify the name of the certificate to use for validation.

Send Validation Confirmation

Send signature validation confirmation. When you select this option, the Trust Server includes a SignatureValidationConfirmation element in the Security header in accordance with the Web Services Security SOAP Message Security 1.1 specification.

Sign Responses

Select to sign the Trust Server SOAP response messages. When you select this option, the Trust server uses the private key you select to sign the body of the SOAP response messages in accordance with the Web Services Security SOAP Message Security 1.1 specification.

Provide the corresponding public key to your client so that they can validate the signature on the message.

Certificate Database
Specify the database where the certificate is stored.

Certificate Label
Specify the name of the certificate to use for validation.

Include
Determine which elements to include in the digital signature for a Trust Server SOAP response message.

Public Key
Specify whether to include the public key with your signature.

Subject Key Identifier
Specify whether to include the X.509 subject key identifier with your signature.

Subject Name
Specify whether to include the subject name with your signature.

Certificate Data
Specify whether to include the BASE64 encoded certificate data with your signature.

Issuer Details
Specify whether to include the issuer name and the certificate serial number with your signature.

The **Properties** tab contains the properties for all modules within the template. Properties for each module can be viewed and updated by selecting the module on the **Template Contents** list.

Procedure

1. Log in to the local management interface.
2. Click **Secure Federation** > **Manage** > **Security Token Service**. By default, the Security Token Service **Module Chains** tab is open. All existing module chains are listed.
3. You can create, modify, or delete module chains.

   Creating a module chain

   a. Click **Add**.
   b. Provide details for the module chain on all tabs.
   c. Click **Save**.
   d. Deploy the changes.

   Modifying a module chain

   a. Select the module chain to modify.
   b. Click **Edit**.
   c. Edit the details on various tabs as needed.
   d. Click **Save**.

   e. Deploy the changes.

Deleting a module chain

   a. Select the module chain to delete.
   b. Click **Delete**.
   c. Click **Delete** to confirm the deletion.
   d. Deploy the changes.

Managing templates

A template is an ordered list of modules and their modes.

Procedure

1. Log in to the local management interface.
2. Click **Secure Federation** > **Manage** > **Security Token Service**.
3. Click **Templates**. All existing templates are listed on the left pane.
4. You can add, modify, or delete templates.

   Adding a template

   a. Click **Add** on the left pane.
   b. Enter a name and description for the new template.
   c. Click **OK**. The new template is added to the template list on the left pane.
   d. Select the new template from the left pane.
   e. Click **Add** on the right pane.
   f. Select the module instance and mode to add to the template.
   g. Click **OK**.
   h. Repeat the previous three steps for each module instance to add to the template.
   i. Click **Move Up** or **Move Down** to adjust the order of the module instances if needed.
   j. Deploy the changes.

   Modifying a template

   a. Select the template to modify from the left pane.
   b. Use the controls in the right pane to make changes.
   c. Deploy the changes.

   Deleting a template

   a. Select the template to delete from the left pane.
   b. Click **Delete** on the left pane.
   c. Click **Delete** in the dialog box to confirm the operation.
   d. Deploy the changes.

Template page scripting

You can use JavaScript to add server-side scripting for Advanced Access Control and Federation template pages. You can use JavaScript functions, closures, objects, and delegations.

Usage

You can customize template files or pages on the server. For example, you can customize an error message that is displayed by the runtime server.

The template files menu is located under both the Secure Federation and Secure Access Control menus.

To edit a Federation template file, go **Secure Federation > Template Files**, select the specific template file, and click **Edit**.

To edit a Secure Access Control template file, go to **Secure Access Control > Template Files**, select the specific template file, and click **Edit**.

The JavaScript engine supports the following syntax:

- Insert JavaScript code between <% and %>.
- Embed JavaScript expressions between <%= and %>.

Example tasks

- Access whitelisted Java classes. For example,
  ```
  var javaStr = new java.lang.String("Hello")
  ```
- Access all the macro variables through templateContext. The standard object to access a Java object is templateContext. For example,
  ```
  templateContext.macros["@TIMESTAMP@"]
  ```
- Use the document.write function to write content to the output stream. For example,
  ```
  templateContext.response.body.write("Hello")
  ```

Examples

Table 1. Example JavaScript

Template HTML	Output

`<%`
`var contents = {product:"ISAM",department:"Lab",country:"SG",region:"Asia"};`
`templateContext.response.body.write(contents.product);`
`%>`
	ISAM

`<%`
`var date = templateContext.macros["@TIMESTAMP@"].substring(0, 10);`
`templateContext.response.body.write(date);`
`%>`
	2017-01-25

The following code example shows how to use repeatable macros. The following example shows an OAuth consent page.

```
<%
var test = templateContext.macros["oauthTokenScopeNewApprovalRepeatable"];
n = test.length;
for (i=0; i<n; i++){
  var scope = test[i]["@OAUTH_TOKEN_SCOPE_REPEAT@"];
  if (scope == "contacts"){
   label ="Do you grant permission to the client to access your phone book";
  }
  else if (scope == "photos"){
   label ="Do you grant permission to the client to access your photos";
  }
  else if (scope == "messages"){
   label ="Do you grant permission to the client to access your WhatsApp messages";
```

```
  }
  else{
   label ="Do you grant permission to the client to access your "+scope;
  }
%>
```

Setting an HTTP response header

You can use `templateContext.response.setHeader(HeaderName, HeaderValue)` to set an HTTP response header.

For example, you can set the `Content-Type` to support both a mobile-based browser and a traditional browser. A mobile-based browser might expect JSON format while a traditional browser expects forms-based HTML.

```
 <%
templateContext.response.setHeader("Content-Type","application/json");
var myObj = { "name":"John", "age":31, "city":"New York" };
templateContext.response.body.write(JSON.stringify(myObj));
%>
```

To set an HTTP header that uses forms-based HTML:

```
templateContext.response.setHeader("Content-Type","text/html");
```

Setting an HTTP status code

You can use `templateContext.response.setStatus(Code)` to set an HTTP response status code.

For example, if you want to set the status to `400` (standard code for a bad request):

```
templateContext.response.setStatus(400);
```

Setting a Redirect URL

You can use `templateContext.response.sendRedirect(URL)` to redirect the HTTP response to a different URL.

For example, when you configure single logout, you can redirect the response to a specific target page, based on the federation name. An example scenario is a deployment that has one SAML 2.0 federation with two partner federations. The partner federations are named `saml20app2` and `saml20sp`. The `saml20app2` federation uses an application that is named `jkebank`. The `saml20sp` federation uses an application that is named `jkeschool`. The page to display on logout is determined by the federation name.

```
var fedName = templateContext.macros[@FEDERATION_NAME@"];
if (fedName == "saml20app2")
{
    templateContext.response.sendRedirect("http://jkebank:1337");
}
else if
{
(fedName == "saml20sp")
{
    templateContext.response.sendRedirect("http://jkeschool:1400");
}
```

Obtaining a list of macros that are available for a template page

In some scenarios, you might want to write JavaScript based on configuration values in your deployment. For example, you might implement one action based on the authentication type, such as if the OTP type is TOTP. Another example is you might implement an action if the Federation name of the single sign-on partner matches a certain value.

Information such as the OTP type and partner name can be retrieved only through the template page macros. To use such information, you need to know which macros are used by the page. The JavaScript engine support provides a utility that can print the available macros for a page.

Use the following syntax to obtain a list of the available macros.

```
<% templateContext.response.body.write(JSON.stringify(templateContext.macros)); %>
```

For example, the following sample code prints the macros from a template page that ran a single sign-on flow with a partner that does not exist.

```
{

    "@PAGE_IDENTIFIER@": "/saml20/invalid_init_msg.html",
    "@TARGET@": "https://www.mysp.ibm.com/isam/mobile-demo/diag",
    "@PARTNER_ENTITY_ID@": "",
    "@ERROR_MESSAGE@": "FBTSML002E The value https://saml.partner.com for attribute PartnerId is not valid.",
    "@FEDERATION_NAME@": "saml20idp",
    "@FEDERATION_ENTITY_ID@": "https://www.myidp.ibm.com/isam/sps/saml20idp/saml20",
    "@REQ_ADDR@": "/sps/saml20idp/saml20/logininitial",
    "@ERROR_CODE@": "FBTSML002E",
    "@EXCEPTION_STACK@": "",
    "@PARTNER_NAME@": "",
    "@TIMESTAMP@": "2017-06-22T03:34:39Z",
    "@SAMLSTATUS@": "<fim:FIMStatusCollection xmlns:fim=\"urn:ibm:names:ITFIM:saml\"
     xmlns:samlp=\"urn:oasis:names:tc:SAML:2.0:protocol\"><fim:FIMStatusCollectionEntry>
     <samlp:Status><samlp:StatusCode Value=\"urn:oasis:names:tc:SAML:2.0:status:Responder\"></samlp:StatusCode>
     <samlp:StatusDetail><fim:FIMStatusDetail MessageID=\"invalid_attribute_value\">
     <fim:SubstitutionString>https://saml.salesforce.com</fim:SubstitutionString>
     <fim:SubstitutionString>PartnerId</fim:SubstitutionString></fim:FIMStatusDetail>
     </samlp:StatusDetail></samlp:Status></fim:FIMStatusCollectionEntry></fim:FIMStatusCollection>",
    "@EXCEPTION_MSG@": ""

}
```

The format is JSON `{ "name1":"value1","name2":"value2"}`

Limitations

- JavaScript validation is done only when a template file is edited (imported) or created. A template file that is imported as a part of an Import compressed file is not validated.
- You must restart the runtime manually to activate changes to OpenID Connect template files. In the administrative interface, click **Secure Federation -> Runtime Tuning -> Restart Runtime**.
- When you access a variable, do not end the variable name with a semicolon. For example, in the following JavaScript, do not end `<%=example%>` with a semicolon `<%=example;%>`.

  ```
  <%var example = "Hello World"; %>
  <%=example%>
  ```

  The correct syntax is `<%=example%>`. Do not use the incorrect syntax `<%=example;%>`.

Managing modules

A module is a module class and set of initial parameters.

Procedure

1. Log in to the local management interface.
2. Click **Secure Federation** > **Manage** > **Security Token Service**.
3. Click **Modules**. All existing modules are displayed.
4. You can view the modules.
   a. Select the module to view.
   b. Click **View**.
   c. Observe the module properties.
   d. Click **Close**.

Chapter 6. Managing server connections

To access data from outside of your appliance, you must define a server connection.

Before you begin

Obtain the connection information for an existing LDAP database server.

About this task

With a Federation module activated, you can create server connections to an LDAP data source. You can have multiple servers for an LDAP connection.

Note: Even though other server connection types are available to select in the local management interface, such as DB2, only the LDAP server connection is used by Federation module.

If you also have the Advanced Access Control module activated, you can create any of the server connection types. See Managing server connections.

Procedure

1. Log in to the local management interface.
2. Click **Secure Federation**.
3. Under **Global Settings**, click **Server Connections**.
4. Take one of the following actions:

   Filter server connections:

   a. In the Quick Filter field, type one or more characters. For example, enter `g` to search for all server connection names that contain g or G.
   b. Press Enter.

   Add a server connection:

   a. Click the ✚ drop-down button.
   b. Select **LDAP**.
   c. Complete the properties for the new server connection. See "Server connection properties" on page 26. Look specifically for the LDAP properties.

   Modify an existing server connection:

   a. Select a server connection.
   b. Click the edit icon ✎.
   c. Complete the properties for the server connection. See "Server connection properties" on page 26.

   Delete a server connection:

   Note: Be careful about removing a server connection that is in use.

   a. Select a server connection.

b. Click the delete icon .
c. Click **Delete** to confirm the deletion.

What to do next

After you define a server connection to an LDAP data source, you can create an attribute source that looks up information from the LDAP server.

Server connection properties

To access a data source outside of the appliance, define the properties of the server.

The Server Connection properties table describes the properties on the **Server Connections** panel for the Advanced Access Control and Federation module activation levels.

- **Advanced Access Control**: Configure LDAP, database, web service, or Cloud Identity server connections so that you can set up policy information points. You can configure any of the server connection types.
- **Federation**: Configure an LDAP server as an attribute source for attribute mapping. Federation does not configure any of the other database server connection types.

Table 2. Server Connection properties

Property	**Description**
Name	Specifies the name for the server connection. Ensure that the name is unique. Select this name when you define the policy information point. **Note:** The server connection name must begin with an alphabetic character. Do not use control characters, leading and trailing blanks, and the following special characters ~ ! @ # $ % ^ & * ( ) + | ` = \ ; " ' < > ? , [ ] { } / anywhere in the name.
Description	Describes the server connection. This property is optional.
Type	Shows the server connection type. (Read only)
JNDI ID (Oracle, DB2, solidDB, PostgreSQL only)	Specifies the JNDI ID that the server uses. Ensure that the ID is unique. Use only alphanumeric characters: a-b, A-B, 0-9
Server name (Oracle, DB2, solidDB, PostgreSQL, SMTP only)	Specifies the name or IP address for the server.
Port (Oracle, DB2, solidDB, PostgreSQL, LDAP, SMTP only)	Specifies the port number where the connection to the server can be made.
URL (Web Service only)	Specifies the URL where the connection to the server can be made.
User name (Oracle, DB2, solidDB, PostgreSQL, SMTP, and Web Service only)	Specifies the user name that has the correct permissions to access the resources.
Password (Oracle, DB2, solidDB, PostgreSQL, SMTP, and Web Service only)	Specifies the password to access the server.

Table 2. Server Connection properties (continued)

Property	Description
SSL	Specifies whether SSL is used for connecting to the server. Select **True** or **False**. The default value is **True**.
Driver type (Oracle only)	Specifies the driver type. Select **Thin** or **OCI**. The default value is **Thin**.
Service name (Oracle only)	Specifies the name of the service.
Database name (DB2, PostgreSQL only)	Specifies the name of the database.
Host name (LDAP only)	Specifies the host name or IP address of the LDAP server.
Bind DN (LDAP only)	Specifies the LDAP distinguished name (DN) that is used when binding, or signing on, to the LDAP server. **Note:** If this value is set to `"anonymous"`, the appliance uses an anonymous bind to the LDAP directory server. Typically the **`bind-dn`** has significant privileges so that it can be used to modify LDAP registry entries, such as creating users and resetting passwords via pdadmin or the Registry Direct Java API. Using an anonymous connection to LDAP typically comes with very limited access, perhaps at most search and view of entries, at the least no access at all. If anonymous access has sufficient privileges, then it might be usable for the WebSEAL level of access on users and groups. This access includes the permission for a user to change password if `"bind-auth-and-pwdchg = yes"` is set (`"ldap.bind-auth-and-pwdchg = true"` for Registry Direct Java API).
Bind Password (LDAP only)	Specifies the password for the LDAP bind DN. **Note:** If bind DN (`bind-dn`) is set to `anonymous`, you can use any non-empty string as the value of bind password (`bind-pwd`).
Administration hostname (Cloud Identity only)	Specifies the administration hostname of the Cloud Identity subscription.
Client ID (Cloud Identity only)	Specifies the client ID of an API Client on Cloud Identity.
Client Secret (Cloud Identity only)	Specifies the client secret of an API Client on Cloud Identity.
SSL Truststore (LDAP, Web Service, and Cloud Identity only)	Specifies the truststore that verifies the credentials.
SSL Mutual Authentication Key (LDAP, Web Service, and Cloud Identity only)	Label of the client certificate to be presented when connecting to the LDAP. This property is sourced from SSL Truststore. **Note:** This field is required only if mutual SSL authentication is required by the server.

Note: For information on SSL configuration, see Configuring SSL connections.

The properties in the following table are connection manager properties. The defaults that are listed are the current known defaults. All tuning properties are optional.

Table 3. Tuning properties

Property	Description
Aged timeout (seconds) (Oracle, DB2, solidDB, PostgreSQL only)	Specifies the amount of time, in seconds, before a physical connection is discarded by pool maintenance. Specify `-1` to disable this timeout. The default is `-1`.
Connection timeout (seconds)	Specifies the amount of time, in seconds, after which a connection times out. For Oracle, DB2, solidDB, PostgreSQL, and SMTP, specify `-1` to disable this timeout. The default is `30` seconds. For LDAP, specify only integers, 1 or greater. The default is `120` seconds.
Max Idle Time (seconds) (Oracle, DB2, solidDB, PostgreSQL only)	Specifies the maximum amount of time, in seconds, after which an unused or idle connection is discarded during pool maintenance. Specify `-1` to disable this timeout. The default is `1800` seconds.
Max Idle Time (seconds) (LDAP only)	Specifies the amount of time, in seconds, after which an established connection is discarded as idle. Set this to a value lower than the connection idle timeout on the LDAP server. **Note:** This is only applicable for performing Attribute Mapping from an LDAP server.
Reap time (seconds) (Oracle, DB2, solidDB, PostgreSQL only)	Specifies the amount of time, in seconds, between runs of the pool maintenance thread. Specify `-1` to disable pool maintenance. The default is `180` seconds.
Max pool size (Oracle, DB2, solidDB, PostgreSQL only)	Specifies the maximum number of physical connections for a pool. Specify `0` for unlimited. The default is `50`.
Max pool size (LDAP only)	Specifies the maximum number of connections that are pooled. **Note:** This is only applicable for performing Attribute Mapping from an LDAP server.
Min pool size (Oracle, DB2, solidDB, PostgreSQL only)	Specifies the minimum number of physical connections to maintain in a pool. The aged timeout can override the minimum.
Purge policy (Oracle, DB2, solidDB, PostgreSQL only)	Specifies which connections to delete when a stale connection is detected in the pool. Select from the following options: **Entire pool** When a stale connection is detected, all connections in the pool are marked stale, and when no longer in use, are closed. This is the default option. **Failing connection only** When a stale connection is detected, only the connection that was found to be bad is closed. **Validate all connections** When a stale connection is detected, connections are tested and the ones that are found to be bad are closed.
Max connections per thread (Oracle, DB2, solidDB, PostgreSQL only)	Specifies the limit of open connections on each thread.

Table 3. Tuning properties (continued)

Property	Description
Cache connections per thread (Oracle, DB2, solidDB, PostgreSQL only)	Specifies the number of cache connections for each thread.

Chapter 7. Managing LTPA keys

You can create, import, export, and delete LTPA key files that are used by the LTPA token conversion module.

Before you begin

Ensure that your browser allows pop-up windows to be displayed.

Procedure

1. Log in to the local management interface.
2. Click **Secure Federation**.
3. Under **Global Keys**, click **LTPA Keys**.
4. Perform any of the following actions:

   Importing an LTPA key:

   a. Click **Manage** > **Import**.
   b. Click **Browse**.
   c. Select the file that you want to import.
   d. Click **Import**.

   Exporting an LTPA key:

   a. Click **Browse**.
   b. Select the file that you want to export.
   c. Click **Manage** > **Export**.
   d. Confirm that you want to save the file to your local workstation.

   Deleting an LTPA key:

   a. Select the file that you want to delete.
   b. Click **Delete**.
   c. Click **Yes** when you are prompted to confirm the deletion.
5. Deploy the changes as described in Configuration changes commit process.

Chapter 8. Managing JavaScript mapping rules

Create, edit, or delete JavaScript mapping rules.

About this task

When you activate the Federation offering, the following mapping rule types are available:

OIDC OpenID Connect mapping rule.

SAML2_0
SAML 2.0 mapping rule.

Procedure

1. Click **Secure Federation**.
2. Under **Global Settings**, click **Mapping Rules**. All existing mapping rules are displayed.
3. You can create, edit, or delete a mapping rule.
   - To create a mapping rule
     a. Click **Add**.
     b. In the **Content** field, enter the JavaScript mapping rule content.
     c. In the **Name** field, enter a name for the rule.
     d. In the **Category** field, select the type of the mapping rule from the list, or type a name to create your own mapping rule type.

        Note: Only the mapping rule types that apply to your current activated offering are displayed in the list.
     e. Click **Save**.
   - To modify a mapping rule
     a. Select the mapping rule to modify.
     b. Click **Edit**.
     c. Modify the mapping rule in the **Content** field as needed.

        Note: The **Name** and **Category** fields are not editable.
     d. Click **Save**.
   - To delete a mapping rule

     Note: Do not delete a mapping rule that is currently used by a SAML 2.0 or OpenID Connect federation.
     a. Select the mapping rule to delete.
     b. Click **Delete**.
     c. Confirm the delete operation.

Import a mapping rule from another mapping rule

You can reuse mapping rules by importing a mapping rule from another mapping rule.

When you want to create a new mapping rule, or customize an existing mapping rule, you can reuse JavaScript code from a previously defined mapping rule. With this feature, you can define a mapping rule once and then reuse it in other mapping rules.

Use the function `importMappingRule()` to specify a mapping rule to import. For example, you can define a mapping rule that is called `Utility.js` that contains functions for obtaining an HTTP header and an HTTP cookie.

```
function getHeader(name) {
          // function for getting HTTP header
}

function getCookie(name) {
         // function for getting HTTP cookie
}
```

If you have another mapping rule that is called `Credential.js`, which also needs to obtain HTTP headers, use the following code to include the functions from the `Utility.js` mapping rule:

```
importMappingRule("Utility");
var host = getHeader("Host");
// do something with the host header
var sessionID = getHeader("PD-SESSION-ID");
// do something with the session ID
```

The function `importMappingRule()` accepts a list of mapping rule names and imports each of the mapping rules. For example:

```
importMappingRule("Utility","Credential","UserIdentity");
```

Alternatively, you can also make multiple calls to `importMappingRule()` within one script. For example:

```
importMappingRule("Utility");
importMappingRule("Credential");
importMappingRule("UserIdentity");
```

The JavaScript engine throws an error if you do not specify a mapping rule name, or if you specify the name of a mapping rule that does not exist.

Use the Local Management Interface (LMI) to view existing mapping rules that are defined on your system. Select **Secure Federation > Global Settings > Mapping Rules**, or **Secure Advanced Access > Global Settings > Mapping Rules**.

Note:

On the LMI menu, the icon **Import** is for importing mapping rules into IBM Security Access Manager, not for importing a mapping rule into an existing mapping rule. Use the **Edit** icon to add the `importMappingRule()` function to an existing mapping rule.

Appendix. Accessibility features for Security Access Manager

Accessibility features assist users who have a disability, such as restricted mobility or limited vision, to use information technology content successfully.

Accessibility features

Security Access Manager includes the following major accessibility features:

Accessibility features
Supports interfaces commonly used by screen readers. This feature applies to applications on Windows operating systems only.
Can be operated by using only the keyboard.
Allows the user to request more time to complete timed responses.
Supports customization of display attributes such as color, contrast, and font size.
Communicates all information independently of color.
Supports interfaces commonly used by screen magnifiers. This feature applies to applications on Windows operating systems only.
Allows the user to access the interfaces without inducing seizures due to photosensitivity.

Security Access Manager uses the latest W3C Standard, WAI-ARIA 1.0 (http://www.w3.org/TR/wai-aria/), to ensure compliance to US Section 508 (http://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/section-508-standards), and Web Content Accessibility Guidelines (WCAG) 2.0 (http://www.w3.org/TR/WCAG20/). To take advantage of accessibility features, use the latest release of your screen reader in combination with the latest web browser that is supported by this product.

The Security Access Manager online product documentation in IBM® Knowledge Center is enabled for accessibility. The accessibility features of IBM Knowledge Center are described at https://www.ibm.com/support/knowledgecenter/help?view=kc#accessibility.

Keyboard navigation

This product uses standard navigation keys.

Interface information

The Security Access Manager user interfaces do not have content that flashes 2 - 55 times per second.

The Security Access Manager web user interfaces and the IBM Knowledge Center rely on cascading style sheets to render content properly and to provide a usable experience. The application provides an equivalent way for low-vision users to use a user’s system display settings, including high-contrast mode. You can control font size by using the device or web browser settings.

The Security Access Manager web user interface includes WAI-ARIA navigational landmarks that you can use to quickly navigate to functional areas in the application.

Related accessibility information

In addition to standard IBM help desk and support websites, IBM has established a TTY telephone service for use by deaf or hard of hearing customers to access sales and support services:

TTY service 800-IBM-3383 (800-426-3383) (within North America)

IBM and accessibility

For more information about the commitment that IBM has to accessibility, see IBM Accessibility (www.ibm.com/able).

Index

Printed in USA