Configuring WebSphere for SAML SSO

This document describes the configuration of SAML-based SSO authentication in WebSphere Application Server 7 and above. These are generic steps that should be applicable to any application but I will make specific application notes where appropriate.

> If you'd like more detailed information and background on WebSphere SAML support please see Understanding the WebSphere Application Server SAML Trust Association Interceptor article on DeveloperWorks

Step-by-step guide

These steps require access to the WebSphere Admin Console as well as the operating system where WebSphere is installed

1. Login to the operating system where WebSphere is installed
2. Install the default SAML ACS (Assertion Consumer Service) servlet supplied with WebSphere

> The WebSphereSamlSP.ear (i.e. the application installed by this process) can only be installed once using the method below. In a multi-tenant environment where SAML SSO is already configured you will need to follow the steps labeled Manually installing the WebSphere ACS Servlet

   a. If using Windows, open a command prompt
   b. Navigate to the WAS application bin directory (`/opt/IBM/WebSphere/AppServer/bin` on Linux/Unix or `C:\Program Files\IBM\WebSphere\AppServer\bin` on Windows)
   c. If you are installing to a cluster (for example: multi-tenant SCCD) then follow these steps:
      i. Run the following command:

Operating System	Command
Windows	`wsadmin.bat -lang jython -f installSamlACS.py install clusterName`
Linux/Unix	`./wsadmin.sh -lang jython -f installSamlACS.py install clusterName`

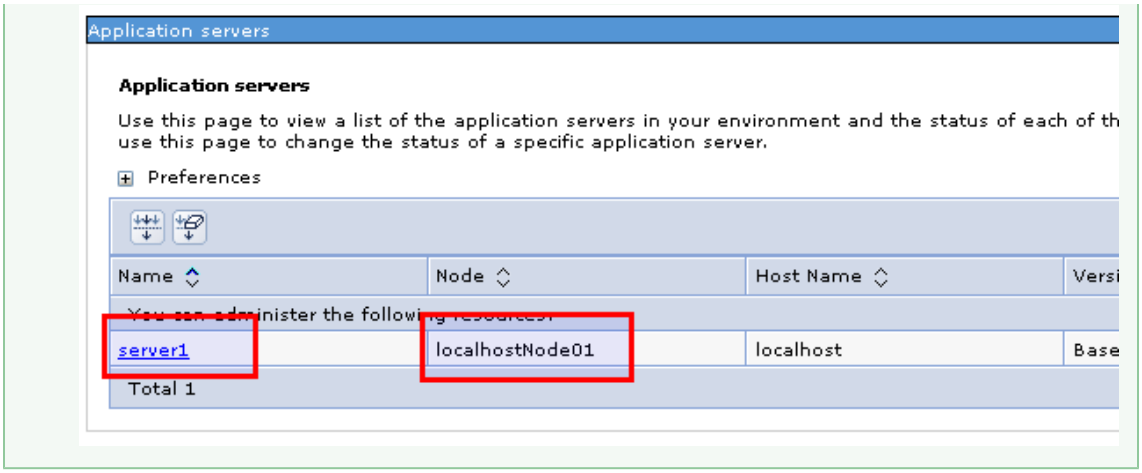
      Where *clusterName* is the name of your WebSphere cluster

   d. If you are installing to a non-clustered server (for example: single-server environments) then follow these steps:
      i. Run the following command:

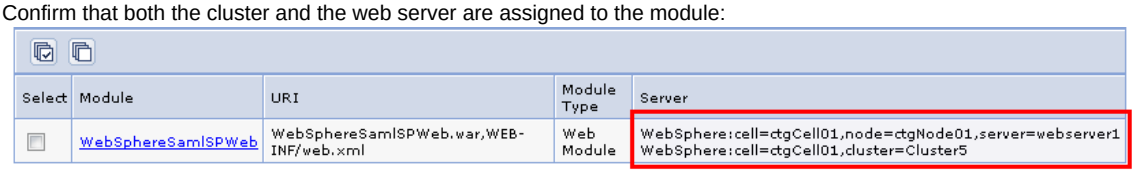
Operating System	Command
Windows	`wsadmin.bat -lang jython -f installSamlACS.py install nodeName serverName`
Linux/Unix	`./wsadmin.sh -lang jython -f installSamlACS.py install nodeName serverName`

      Where *nodeName* and *serverName* are your node and server values respectively

> The server and node names can be seen in the list on the **WebSphere application servers** screen within the Admin Console



e. If you are using a web server such as IBM HTTP Server in front of your application be sure that the newly installed EAR is targeted to the web server
   i. Login to the WebSphere Admin Console
   ii. Using the left-hand menu go to **Applications** and then **WebSphere enterprise applications**
   iii. Click the link for **WebSphereSamlSP**
   iv. Under **Modules** click **Manage Modules**
   v. Confirm that both the cluster and the web server are assigned to the module:



   vi. If changes were required, generate and propagate the plugin configuration
      1. Using the left-hand menu go to **Servers** and then **Server Types** and click **Web servers**
      2. Click the checkbox next to your web server and click **Generate Plug-in** from the toolbar menu
      3. Click the checkbox next to your web server and click **Propagate Plug-in** from the toolbar menu
      4. Restart the web server

3. Login to the WebSphere Admin Console (if necessary)
4. Create a new Security Domain
   a. Using the left-hand menu, select **Security** then **Security Domains**
   b. Click New
   c. Provide a name for your security domain. If this installation of WebSphere is dedicated to a single customer then you can use an application name for this such as MAXIMO or TRIRIGA. If this is a multi-tenant installation then you should probably use a name that indicates which customer and environment this security domain is used by (for example: CUSTADEV, CUSTBPROD, etc)
   d. Provide a description for clarity if necessary
   e. Click **OK**
5. Confirm that Application Security is enabled
   a. From the list of Security Domains, click the new domain you created
   b. Check the value next to **Application Security**. If the value is *Enabled* then you can continue on to step 6
   c. Expand the **Application Security** section and select **Customize for this domain**
   d. Enable the **Enable application security** checkbox and click **Apply**
6. Configure a new Trust Association Interceptor
   a. From the list of Security Domains, click the new domain you created
   b. Expand the **Trust Association** section and select the **Customize for this domain** option
   c. Click to enable the **Enable trust association** checkbox and click **Apply**
   d. Click the **Interceptors** link under **Trust Association**
   e. Click **New**
   f. For the *Interceptor class name* enter `com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor`
   g. Under Custom Properties enter the property name `sso_1.sp.acsUrl` with a value of your ACS URL (see note below)
   h. Click **New** to add an additional property
   i. Enter the name `sso_1.sp.EntityID` and provide a value for the SP entity ID (see second note below)
   Click **OK**

> **ACS URLs**
> The ACS URL is the web address of the servlet you installed in step 2. By default this will be the IP or hostname of your webserver with the context */samlsps* added to the end (ex. https://my.host.com/samlsps). In multi-tenant configurations the context will be different (ex. /samplspsa, /samlspsb, etc). The servlet will accept any arbitrary URL following the /samlsps context which can be used to create a slightly more meaningful URL. A simple example would be https://my.host.com/samlsps/sso

Entity IDs
In the exchange of SAML data between the SP (i.e. WebSphere) and the IdP (the identity provider) an entity ID is used to verify the originating party of each message. The IdP will be configured to accept a specific entity ID from the SP and the SP will in turn be configured to accept a specific entity ID from the IdP. An entity ID can be any arbitrary string but is often defined as a URL – it does not have to be a valid URL. The key point to take from this is that no matter what you use for the entity ID, it must be configured correctly on both sides of the exchange (IdP and SP) in order for SAML processing to work correctly. It is not necessary (and is actually undesirable) for the SP entity ID to match the IdP entity ID. The SP entity ID `sso_1.sp.EntityID` you provide in this step will be automatically added to the metadata you export in step 8. On step 11 you import the metadata from the IdP and its entity ID will be automatically added as `sso_1.idp_1.entityID`.

A simple Entity ID could be the site's external URL with domain name – https://my.host.com

7. Save settings and synchronize nodes if required
8. Export SAML SP metadata
    a. If using Windows, open a command prompt
    b. Navigate to the WAS application bin directory (`/opt/IBM/WebSphere/AppServer/bin` on Linux/Unix or `C:\Program Files\IBM\WebSphere\AppServer\bin` on Windows)
    c. Launch the wsadmin tool

Operating System	Command
Windows	`wsadmin.bat -lang jython`
Linux/Unix	`./wsadmin.sh -lang jython`

    d. Execute the following command

```
AdminTask.exportSAMLSpMetadata('-spMetadataFileName sp_metadata.xml -ssoId
1 -securityDomainName MYDOMAIN')
```

    Be sure to change the `MYDOMAIN` value to match the domain you created in step 4.

    If you do not specify a path for the metadata file it will default to the profile home directory. If you launched *wsadmin* fr om the main bin directory as described above it will default to the deployment manager profile (ex. `/opt/IBM/WebSp here/AppServer/profiles/ctgDmgr01`)

9. Send the SP metadata file to your IdP. Be sure to include the following information with your request:
    a. The target URL for your application. This is the URL you would like to be sent to after authenticating at the IdP
    b. Request that the IdP provide its signing certificate inside the metadata file. If this is not possible you will have to request it separately and import the signing certificate manually later
    c. Ensure the IdP includes the certificate in the signature `KeyInfo` element of the assertion, as some identity providers will need to specify this
10. Once you have received your IdP's metadata file you can proceed to the next step
11. Import the IdP metadata file
    a. Follow steps 8a, b and c to launch the *wsadmin* tool
    b. Execute the following commands

```
AdminTask.importSAMLIdpMetadata('-idpMetadataFileName idp_metadata.xml
-signingCertAlias MyCertAlias -securityDomainName MYDOMAIN')
AdminConfig.save()
```

    If the `idp_metadata.xml` file is not in the same path as the *wsadmin* tool then you will need to specify the full path to the file. The value for *signingCertAlias* can be any string; it will be used to identify the signing certificate in the WebSphere Trust Store so just choose a suitable name that is not already in the store (see **Security > SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore > Signer certificates** for a list of keys already in the store). Be sure to change `MYDO MAIN` to match the domain you created in step 4.
    c. Exit the *wsadmin* tool and return to the WebSphere Admin Console
12. Verify TAI custom properties
    a. Using the left-hand menu select **Security** and then **Security Domains**
    b. Click the link to your security domain
    c. Expand the **Trust Association** section and click the **Interceptors** link
    d. Click **com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor**
    e. Check that the following properties are defined:

Property	Value
sso_1.sp.acsUrl	The value you assigned the property in step 6g
sso_1.sp.EntityID	The value you assigned the property in step 6i
sso_1.idp_1.certAlias	The name of the certificate alias you provided in step 11b. See the note below if you do not see this value
sso_1.idp_1.entityID	The entity ID of the IdP which is provided in the IdP metadata file and will be automatically populated
sso_1.idp_1.singleSignOnUrl	The URL endpoint for IdP authentication (automatically populated from metadata)

If you do not see the `sso_1.idp_1.certAlias` property then a certificate was not provided with the metadata file. You will need to obtain the certificate from the IdP and add it to WebSphere manually by going to **Security > SSL certificate and key management > Key stores and certificates > CellDefaultTrustStore > Signer certificates** and clicking **Add**. Once imported, you will need to add a custom property to your TAI (see steps above on how to get to the custom properties screen) named `sso_1.idp_1.certAlias` and assign it the value of the new certificate alias you created.

Keep a note of the value for `sso_1.idp_1.entityID` as we will make use of it in a future step

f. If you are setting up SSO with an existing LDAP configuration in WebSphere (for example: Maximo SCCD with LDAP enabled) you need to set an additional custom property

If setting up SSO for TRIRGA you can set the `sso_1.sp.idMap` value to `idAssertion` but it isn't necessary because `idAssertion` is the default.

i. Click **New** to add a new property if there is not already space to enter a property on the screen
ii. Enter `sso_1.sp.idMap` as the property name
iii. Enter `localRealm` as the property value
iv. Click **OK**
v. Save changes

13. Finalize Security Domain setup
    a. Using the left-hand menu select **Security** and then **Security Domains**
    b. Expand **User Realm**, click the **Customize for this domain** radio button
    c. **IMPORTANT!** Click **Apply** at the bottom of the screen and save changes
    d. Go back to the Security Domain, expand User Realm (it should already be set to Customize...) and click **Configure...**
    e. If you are not taken to the **Trusted authentication realms - inbound** page automatically then click the associated link in the lower right part of the screen (under Related Items)
    f. Click the **Add External Realm...** button in the toolbar
    g. Enter the value of the `sso_1.idp_1.entityID` from step 12d and click **OK**
    h. Click **Apply** and save changes and return to the security domain configuration screen by following steps 13a and 13b
    i. Click the **Custom Properties** link at the bottom of the screen
    j. Add the following two properties:

Property	Value
com.ibm.websphere.security.DeferTAItoSSO	com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor
com.ibm.websphere.security.InvokeTAIbeforeSSO	com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor

    k. Click **OK** and save changes
14. Assign security domain to servers or clusters

You must assign the server/cluster where the WebSphereSamlSP.ear was deployed in step 2 to your security domain or the SSO configuration will not work

    a. Proceed to the security domain configuration screen as described in steps 13a and 13b
    b. Under the heading **Assigned Scopes** expand the tree starting at *Cell*
    c. Locate the server(s) or cluster(s) where you would like to enable SSO and click each one to enable it. If you are not using clusters then your servers will appear under the *Nodes* section. If your servers are in clusters then you must look under the *Clust ers* section.
    d. When you have enabled all appropriate servers click the **OK** button at the bottom of the page and save your changes
15. **TRIRGA ONLY**
    a. Modify the TRIRIGAWEB.properties file and change the following values:

```
SSO=Y
SSO_REMOTE_USER=N
SSO_USER_PRINCIPAL=Y
```

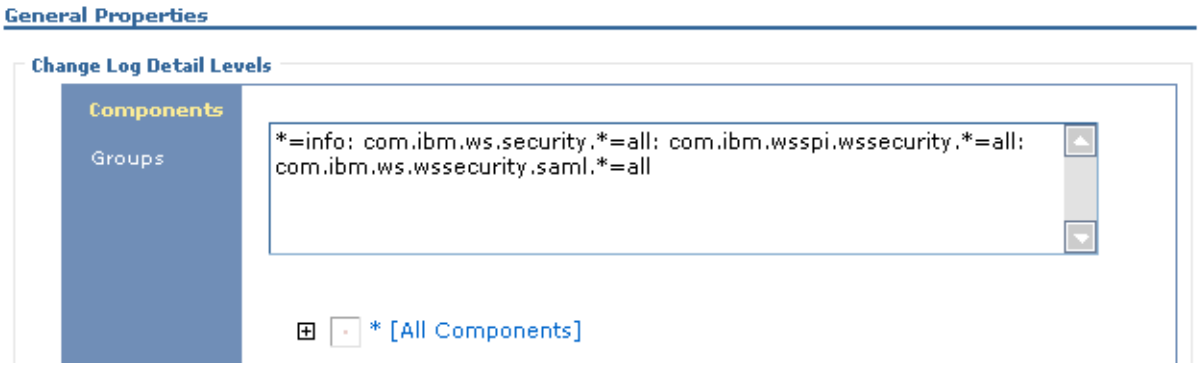
16. Restart application servers and web server to pick up configuration changes
17. Test SSO using the login URL provided by your IdP

Troubleshooting

To troubleshoot issues with SAML you will need to enable trace logging on the server where the SAML ACS servlet has been installed. By default, WebSphere provides no feedback in the standard logs for most SSO issues. If you are troubleshooting SAML in a cluster where multiple servers are running it is recommended you stop all but one server to simplify diagnosing your problem.

To enable trace logging for the server where the ACS servlet is installed, login to the WebSphere Admin Console and do the following:

1. Using the left-hand menu select **Servers** then **Server Types** then **WebSphere application servers**
2. Locate the server where you installed the ACS servlet in step 2 of the Step-by-step guide and click it
3. Under **Troubleshooting** on the right side click **Diagnostic trace service**
4. Under **Additional Properties** click **Change Log Detail Levels**
5. Add the following log levels, separating each with a colon:
    a. `com.ibm.ws.security.*=all`
    b. `com.ibm.wsspi.wssecurity.*=all`
    c. `com.ibm.ws.wssecurity.saml.*=all`



6. Click **OK** and save your changes
7. Restart the server where you have enabled trace logging

Now test your SSO flow again and view the `trace.log` file in the log folder of your server for errors. The messages will generally give some indication of where the problem lies but you may need to find a proper person to escalate to if you cannot determine the problem.

Problem: After logging in to the IdP you are presented with a login prompt by the browser for "Default Realm"

Solution:

Verify that you have enabled the TAI and saved the changes. Also make sure you have assigned your Security Domain correctly. See steps 6c and 14 in the Step-by-step guide.

Problem: After logging in to the IdP you are presented with a "403 Forbidden" page

Solution:

The 403 error indicates that the TAI is correctly intercepting the SAML Response but there is a problem somewhere in the SAML processing. You will need to enable the trace logging as described above and find the exact error in the log file.

Problem: I would like users to be able to login using the standard login page as well as SSO

Solution:

This should work by default for TRIRIGA because it is not designed to use container managed security as of March 2014. For Maximo/SCCD you will need to set the `sso_1.sp.filter` property to prevent the TAI from intercepting the login page (otherwise you will get a "403 Forbidden" message when trying to go directly to the login page) and disable enforcing of the TAI cookie. Add new properties to the SAML TAI as in step 6 but with the following names and values:

Name	Value
sso_1.sp.filter	request-url!=/login
sso_1.sp.enforceTaiCookie	false

Problem: How do I automatically redirect users to the IdP login URL?

Solution:

This feature requires an application that supports container-based authentication (JAAS) so it will not currently work with TRIRIGA. You can do this by setting the following value in the TAI properties:

Name	Value
sso_1.sp.login.error.page	*Your IdP Login URL*

Keep in mind that you should remove the values for `sso_1.sp.filter` and `sso_1.sp.enforceTaiCookie` from the previous solution or you will not be redirected when hitting the login page.

Problem: The /samlsps URL returns a 404 error

Solution:

Make sure to regenerate and propagate the Web Server plug-in configuration and restart IHS

Problem: ADFS does not support RelayState

Solution:

If the IdP does not support RelayState you can set the Target URL via SAML TAI properties

Name	Value
sso_1.sp.useRelayStateForTarget	false
sso_1.sp.targetUrl	*Application landing page* (ex. https://host/maximo/webclient/login/login.jsp)

Manually Installing the WebSphere ACS Servlet

In order for SSO processing to work correctly in a multi-tenant environment, the ACS servlet (WebSphereSamlSP.ear) must be installed to a server/cluster within the security domain you have configured (refer to steps 4 and 14 in the Step-by-step instructions). The default method of installing the servlet using *wsadmin* will not work if you have already installed it once for another tenant. This is for two reasons: 1) WebSphere only allows one application with the same name to be deployed on a cell and the script does not allow you to change the application name and 2) the application always installs using the same context which will cause issues with web server deployment.

To manually deploy the servlet, login to the WebSphere Admin Console and follow these steps:

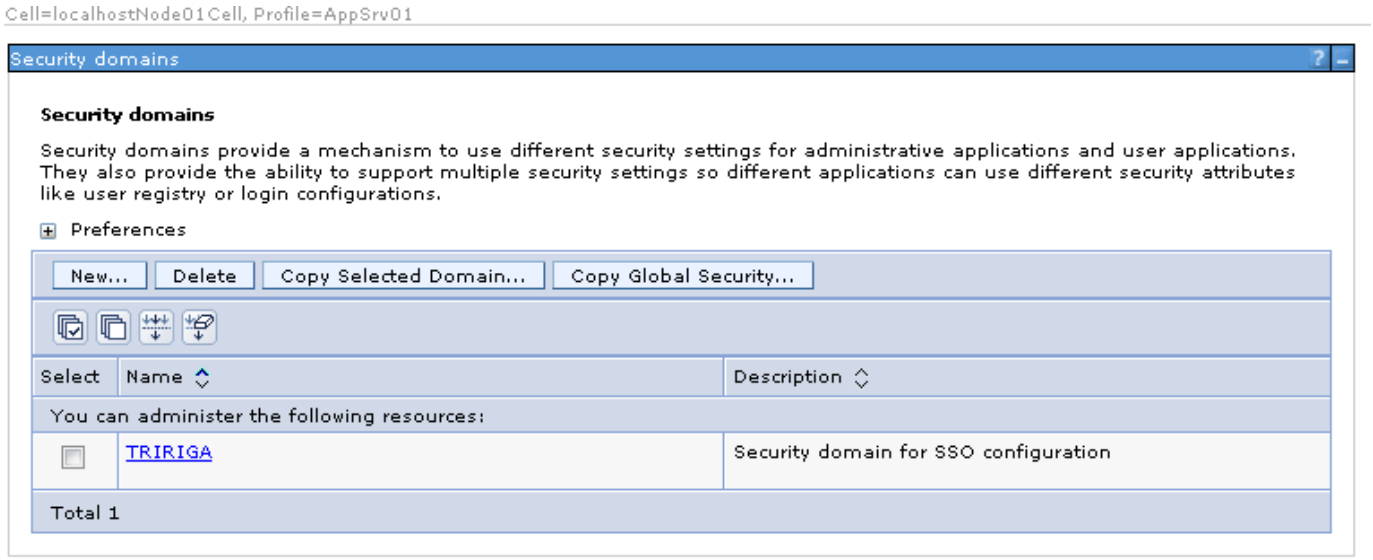
1. From the left-hand menu, expand **Applications** and then **Application Types**
2. Click **WebSphere enterprise applications**
3. Click **Install** on the toolbar
4. Select **Remote file system** and enter the path `/opt/IBM/WebSphere/AppServer/installableApps/WebSphereSamlSP.ear` (if WebSphere is not installed at `/opt/IBM` then correct the path as required)
5. Click **Next**
6. Click **Next** on the **Preparing for the application installation** screen

7. Change the value of *Application name* to a value that is not already used in your cell. For SCCD multi-tenant environments it's best practice to add the same letter that is used for the application cluster you will be targeting. For example, if this is for MAXIMOA then use WebSphereSamlSPA, etc
8. Click **Next**
9. On the **Map modules to servers** screen be sure that the WebSphereSamlSPWeb module is assigned to the desired cluster and the web server if you are using one
10. Click **Next**
11. Click **Finish** and **Save** when the deployment is complete
12. Follow steps 1 and 2 to return to the **WebSphere enterprise applications** screen
13. Click the link to the application you just installed (for example: WebSphereSamlSPA)
14. Under **Web Module Properties** on the right-side click **Context Root For Web Modules**
15. Change the context root to match the tenant designation (for example: *samlsps* becomes *samlspsa*)
16. Click **OK** and save your changes
17. Update the web server plugin as described at the bottom of step 2e in the Step-by-step guide

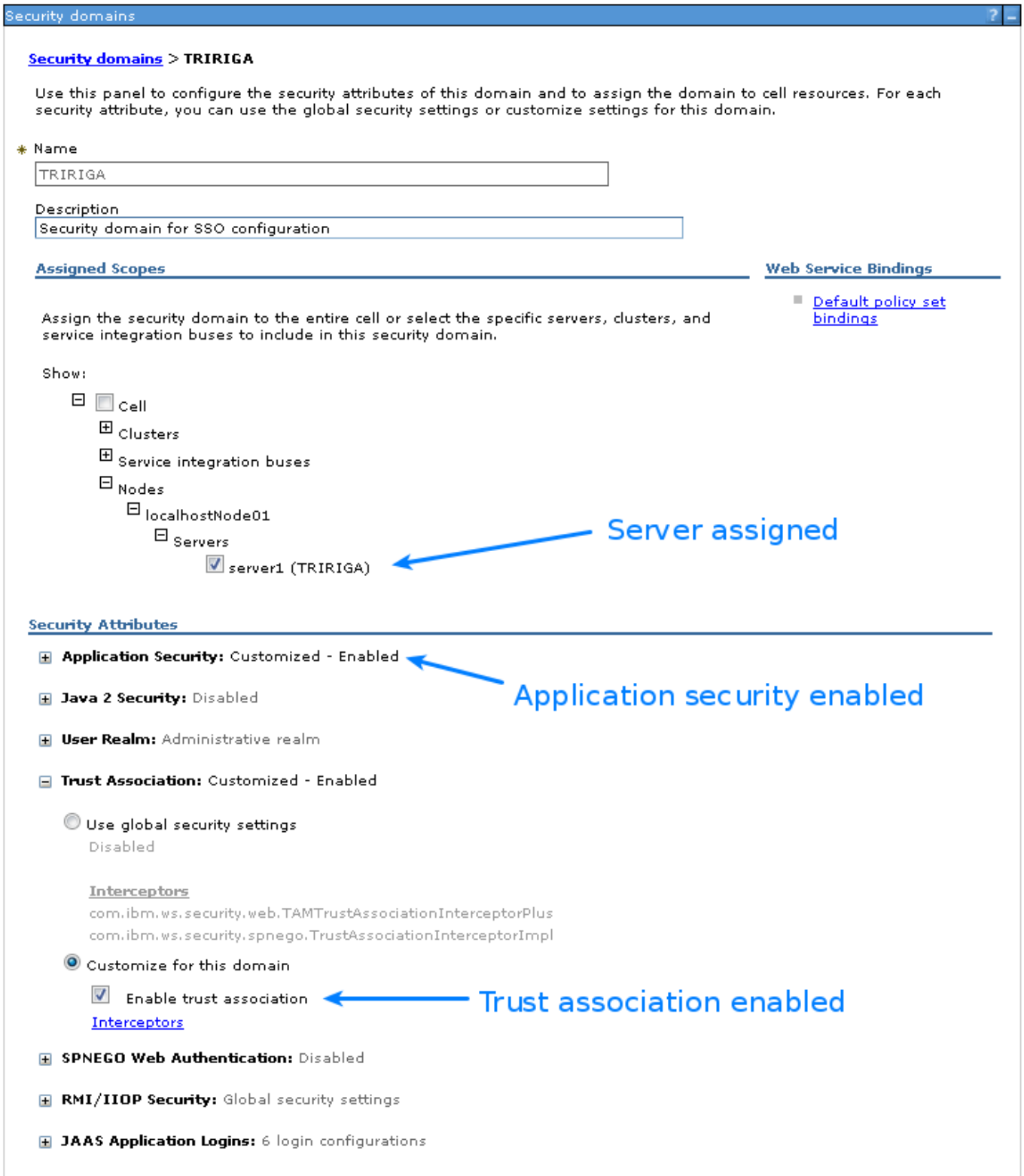
Working Examples

IBM CIO SSO (Bluepages) with IBM Tririga

New Security Domain (Step 4)



Security Domain configuration (Steps 5, 6 and 14)



Security Domain custom properties (Step 13)

Cell=localhostNode01Cell, Profile=AppSrv01

Security domains

Security domains > TRIRIGA > Custom properties

Specifies arbitrary name and value pairs of data. The name is a property key and the value is a string value that can be used to set internal system configuration properties. The name and value pairs below are local to this domain and override the global custom properties. Any name and value pairs not listed below will be obtained from the global custom properties.

Custom properties

New | Edit | Delete

Select	Name	Value
	com.ibm.websphere.security.InvokeTAIbeforeSSO	com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor
	com.ibm.websphere.security.DeferTAItoSSO	com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor

Apply | OK | Reset | Cancel

TAI properties (Steps 6, 8, 11 and 12)

Cell=localhostNode01Cell, Profile=AppSrv01

Security domains

Security domains > TRIRIGA > Interceptors > com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor

Specifies the trust information for reverse proxy servers.

General Properties

* Interceptor class name

com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor

Custom properties

New | Edit | Delete

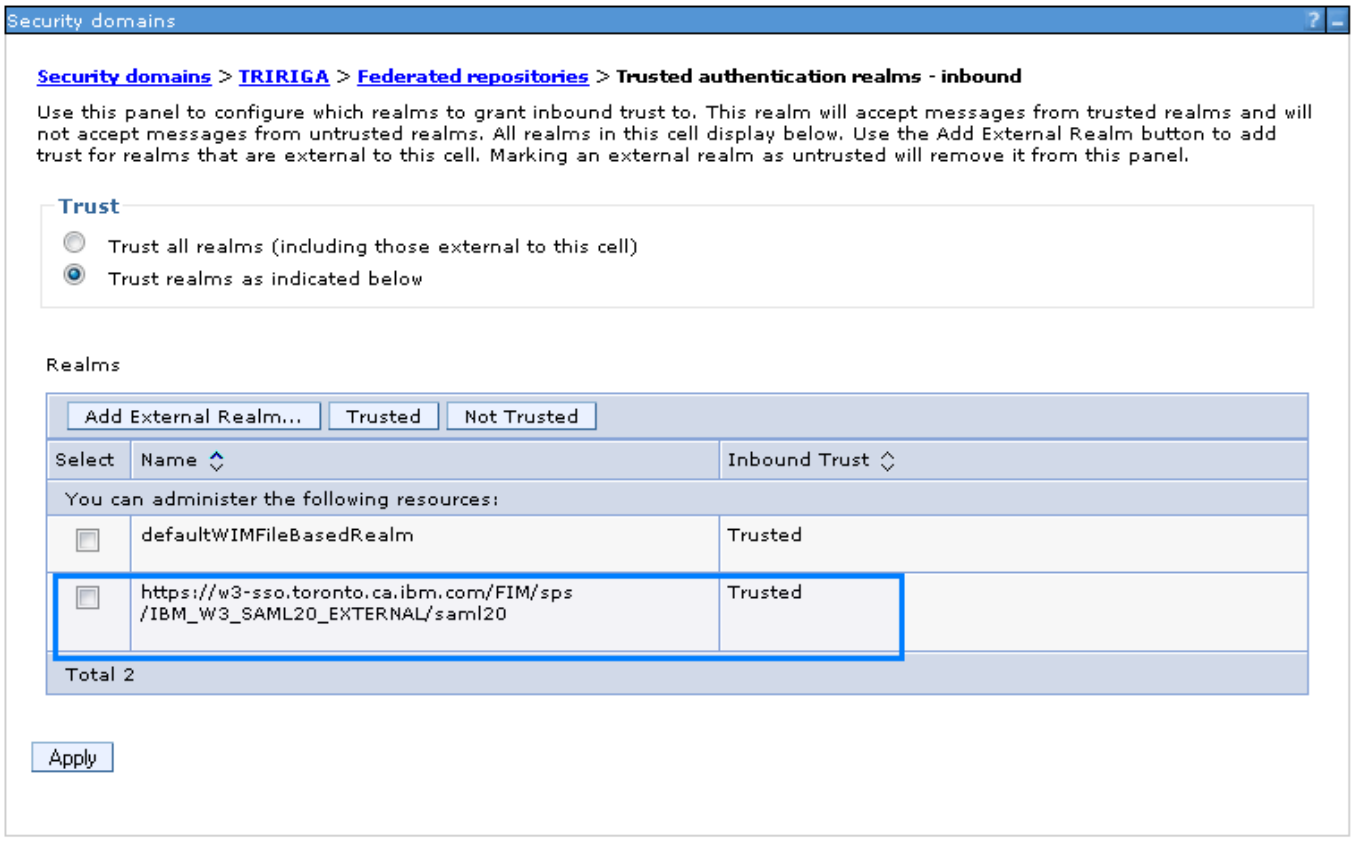
Select	Name	Value
	sso_1.sp.acsUrl	https://reso-test.mro.com/samlsps/gateway/ See step 6
	sso_1.sp.idMap	idAssertion See step 12f
	sso_1.sp.EntityID	https://reso-test.mro.com/tririga/ See step 6
	sso_1.sp.login.error.page	https://w3-sso.toronto.ca.ibm.com/FIM/sps/IBM_W3_SAML20_EXTERNAL/saml20 /login?providerId=https%3A%2F%2Freso-test.mro.com optional
	sso_1.sp.acsErrorPage	https://reso-test.mro.com/html/en/default/error /defaultError.htm?messageId=SSO_ERROR optional
	sso_1.sp.targetUrl	https://reso-test.mro.com/login optional
	sso_1.idp_1.certAlias	shibbolethcert
	sso_1.idp_1.entityID	https://w3-sso.toronto.ca.ibm.com/FIM/sps/IBM_W3_SAML20_EXTERNAL/saml20
	sso_1.idp_1.singleSignOnUrl	https://w3-sso.toronto.ca.ibm.com/FIM/sps/IBM_W3_SAML20_EXTERNAL/saml20/login

Step 11

Apply | OK | Reset | Cancel

Trusted authentication realms - inbound (Step 13b-e)

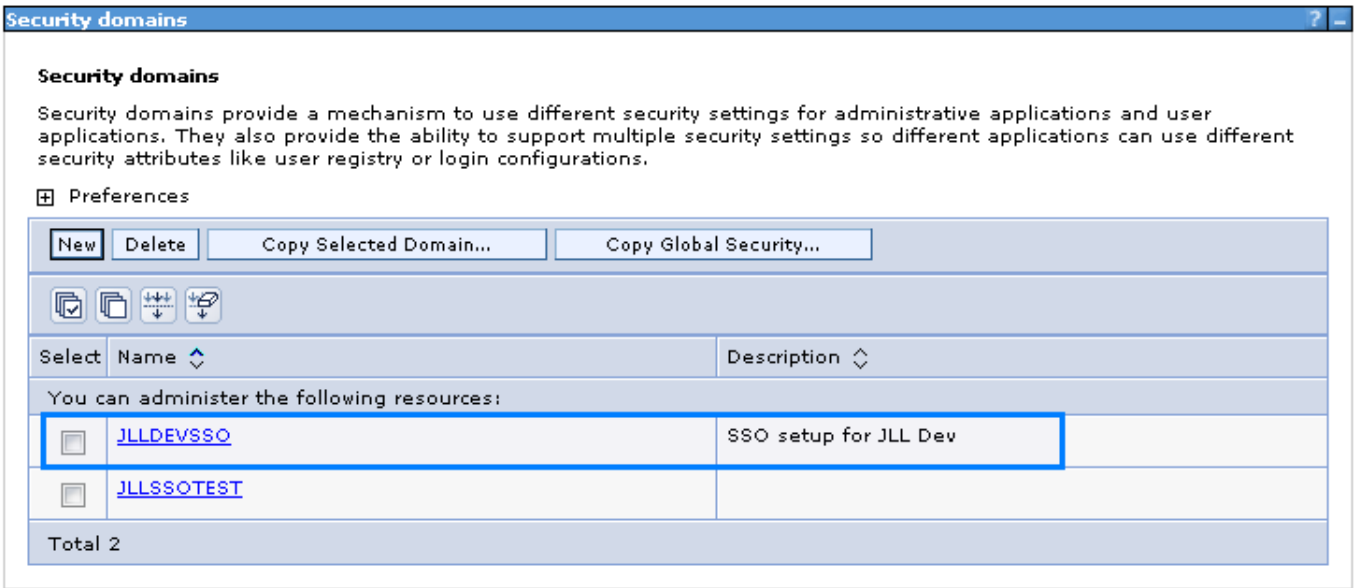
Cell=localhostNode01Cell, Profile=AppSrv01



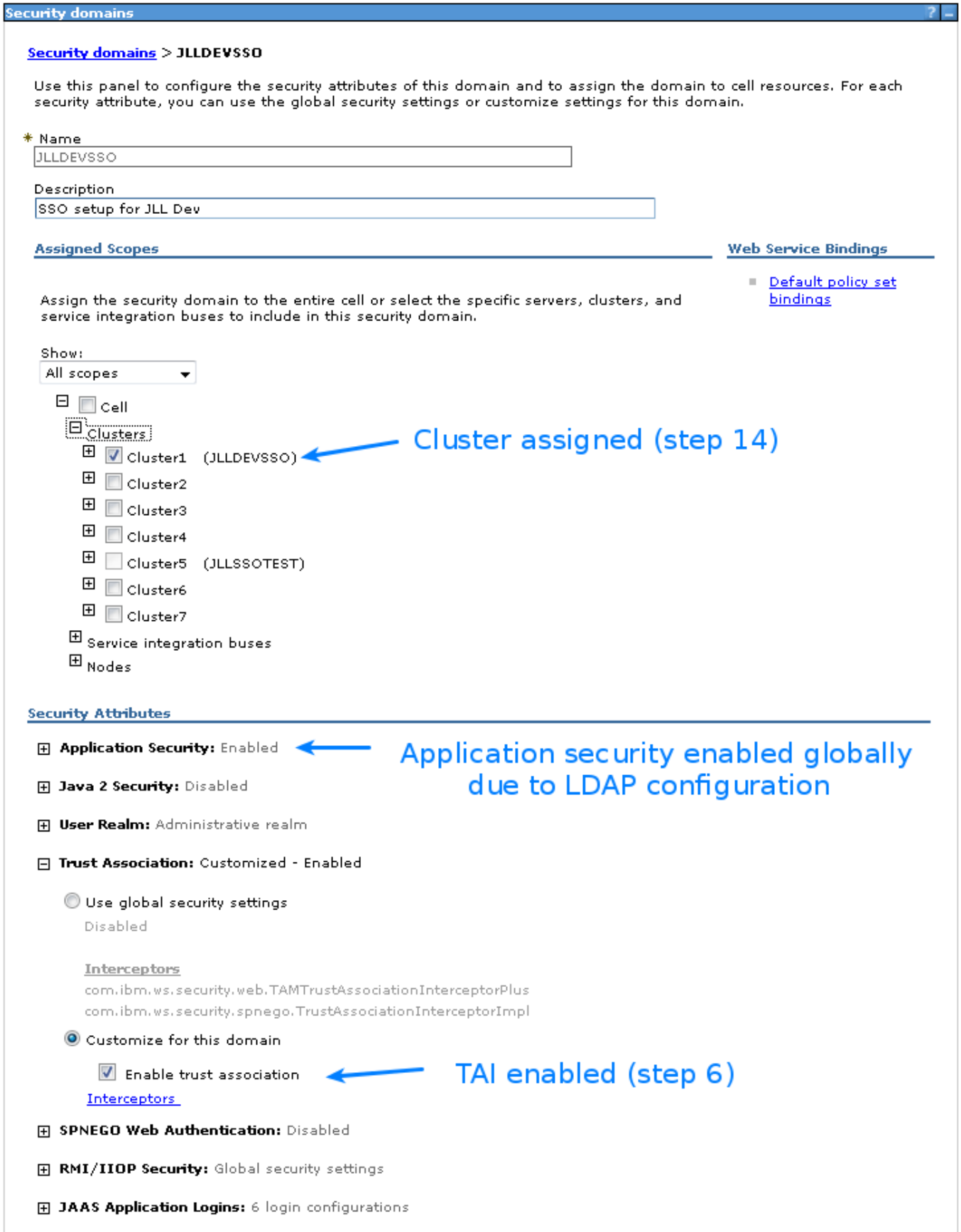
Ping Identity SSO with Maximo SCCD (multi-tenant)

New Security Domain (Step 4)

Cell=ctgCell01, Profile=ctgDmgr01



Security Domain configuration (Steps 6 and 14)



Security Domain custom properties (Step 13)

Security domains

Security domains > JLLDEVSSO > Custom properties

Specifies arbitrary name and value pairs of data. The name is a property key and the value is a string value that can be used to set internal system configuration properties. The name and value pairs below are local to this domain and override the global custom properties. Any name and value pairs not listed below will be obtained from the global custom properties.

Custom properties

Select	Name	Value
☐	com.ibm.websphere.security.DeferTAItoSSO	com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor
☐	com.ibm.websphere.security.InvokeTAIbeforeSSO	com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor
☐		

New Edit Delete

Apply OK Reset Cancel

TAI Properties (Steps 6, 8, 11 and 12)

Security domains

Security domains > JLLDEVSSO > Interceptors > com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor

Specifies the trust information for reverse proxy servers.

General Properties

* Interceptor class name

com.ibm.ws.security.web.saml.ACSTrustAssociationInterceptor

Custom properties

Select	Name	Value
☐	sso_1.sp.acsUrl	https://jll-dev.sccd.ibmserviceengage.com/samlspsc/jll
☐	sso_1.sp.EntityID	https://jll-dev.sccd.ibmserviceengage.com
☐	sso_1.idp_1.certAlias	jlldevsso
☐	sso_1.idp_1.entityID	https://smtest02.am.joneslanglasalle.com
☐	sso_1.idp_1.singleSignOnUrl	https://smtest02.am.joneslanglasalle.com/idp/SSO.saml2
☐	sso_1.sp.idMap	localRealm Step 12f
☐	sso_1.sp.filter	request-url!=/login optional
☐	sso_1.sp.enforceTaiCookie	false optional
☐		

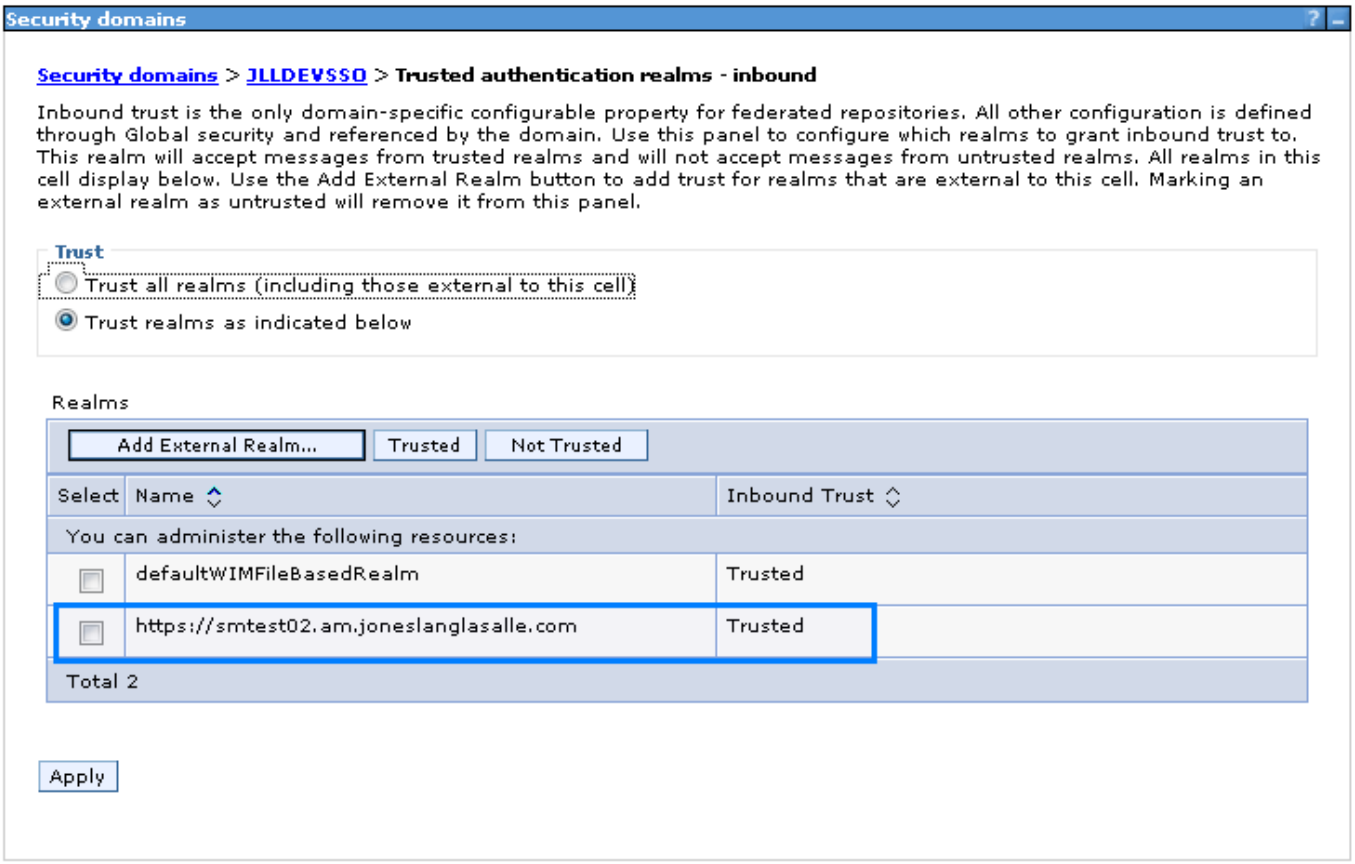
Step 6

Step 11

New Edit Delete

Apply OK Reset Cancel

Trusted authentication realms - inbound (Step 13b-e)



Related articles

Content by label

There is no content with the specified labels