An Oracle White Paper
March 2014

Integrating Oracle Sales Cloud with On-Premise Oracle WebCenter Portal

Table of Contents

EXECUTIVE OVERVIEW **1**

INTRODUCTION **1**

VISUALIZING SALES OPPORTUNITIES USING A SYSTEM ACCOUNT **3**

SETTING UP YOUR ENVIRONMENT 3

VISUALIZING THE SALES OPPORTUNITY DATA 7

VISUALIZING SALES OPPORTUNITIES USING THE ORACLE SECURITY TOKEN SERVICE **13**

REPLICATING ORACLE SALES CLOUD USERS 14

SETTING UP AND CONFIGURING ORACLE STS 14

CONFIGURING WEBCENTER PORTAL TO USE ORACLE STS 28

CONCLUSION **30**

Executive Overview

Cloud computing is becoming a reality for many companies and organizations who have started using Cloud-based service providers or are moving their IT infrastructure to the Cloud. While subscribing to a Cloud-based service or deploying applications on the Cloud may be relatively straight-forward, how do you integrate with other enterprise applications that are still running on-premise?

Enterprise portals are traditionally used to aggregate data and content from various enterprise applications on-premise. Can enterprise portals provide a platform for integrating applications running on-premise and on the Cloud? This white paper shows how you can meet this requirement with Oracle WebCenter Portal, which can itself run either on-premise or on an Oracle Public or Private Cloud service.

Introduction

Oracle WebCenter Portal customers deploy Intranet portals to aggregate data and content from various enterprise applications and provide self-services for employees, partners, and customers. Traditionally, these enterprise applications and the portal ran on-premise. As adoption of Cloud-based applications grows, our customers more and more often have a requirement to extend their Intranet portals by integrating data and content from Cloud service providers, along with traditional on-premise applications and content repositories.

Integrating a portal running on-premise with Cloud-based applications brings new challenges. For example, a portal server deployed on-premise normally shares the same LDAP repository with other enterprise applications running on-premise. This shared repository makes it relatively straight-forward to enable user identity propagation between the portal and integrated applications, without configuring a complete SSO solution.

Cloud-based services or applications deployed on the Cloud typically manage user accounts in their own user repositories. This presents challenges for configuring security that enables components running on the portal on-premise to access secured data in the Cloud-based applications. Identity Federation is one proven concept that can address these challenges and securely share user profile information across the portal and external services or applications.

The primary objective of this white paper is to recommend an integration pattern between WebCenter Portal and Cloud-based Oracle Applications that enables portal components to access data stored in the applications. This is done by providing a step-by-step guide for configuring security and integrating on-premise WebCenter Portal with Cloud based Oracle Applications using the following two approaches:

- On-premise WebCenter Portal is integrated with the Oracle Sales Cloud application using a fixed System Account
- On-premise WebCenter Portal is integrated with the Oracle Sales Cloud application using Identity Federation

To illustrate application data integration, this white paper describes how the SOAP web services that Oracle Sales Cloud exposes can be consumed in WebCenter Portal's Data Presenter to create, at runtime, a data visualization that shows a list of sales opportunities. This integration is created using Oracle WebCenter Portal 11*g* Release 1 (11.1.1.8).

The fixed System Account approach is simpler to configure. However, if web service security is configured to use the same fixed Sales Cloud account (System Account), every WebCenter Portal user will see the same list of sales opportunities. We recommend using this approach if the use case for Cloud service integration only requires providing portal users with shared access to data.

If the integration use case requires providing access to secured data using the credentials of each individual user, Identity Federation should be used. Once Identity Federation is configured, the identity of each WebCenter Portal user is propagated in the call to the Oracle Sales Cloud web service, and as a result each WebCenter Portal user sees the list of sales opportunities applicable to his or her role in the Sales Cloud application. A pre-requisite for using Identity Federation is to ensure that each WebCenter Portal user has a corresponding Cloud identity.

Visualizing Sales Opportunities Using a System Account

In this section, we look at how to display a list of sales opportunities in Oracle WebCenter Portal using SOAP web services to access data stored in the Oracle Sales Cloud application. To achieve this we describe how to:

- Configure security for a web service call using a fixed System Account managed in the Oracle Sales Cloud.
- Display data that this web service returns on the portal page using WebCenter Portal Data Presenter.

Setting Up Your Environment

Install WebCenter Portal 11.1.1.8.0

If you have not already done so, install and configure WebCenter Portal 11.1.1.8.0:

- Download Oracle WebCenter Portal 11.1.1.8.0 from Oracle Technology Network. WebCenter Portal software is available from:

http://www.oracle.com/technetwork/middleware/webcenter/portal/downloads/index.html

- Follow the instructions in *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal* to install and configure WebCenter Portal.

  Note, some configuration steps described in this white paper require access to Enterprise Manager Fusion Middleware Control. Fusion Middleware Control is typically configured for a domain and automatically starts up when you start the Oracle WebLogic Server Administration Server.

- To verify that Fusion Middleware Control is accessible, navigate to the following URL in your browser:

  http://*hostname.domain*:*port*/em

  For example:

  http://myhost.mycompany.com:7001/em

  The port number is the port number of the Administration Server. By default, this port is 7001. If you are not sure, the port number is listed in the domain *config.xml* file, which can be found at the following location:

  - On Windows: *DOMAIN_HOME**config**config.xml*
  - On UNIX: *DOMAIN_HOME/config/config.xml*

Create the WebCenter Portal Domain Keystore

Once WebCenter Portal is installed and configured you must create the keystore for the WebCenter Portal domain. This section describes how to create the keystore using Java Keystore (JKS). JKS is the proprietary keystore format defined by Sun Microsystems. To create and manage the keys and certificates in JKS, use the *keytool* utility that is distributed with Java JDK 6.

To create the keystore for WebCenter Portal domain:

- Go to ***JDK_HOME/jdk/bin*** and open a command prompt.
- Using ***keytool***, create a keystore for the WebCenter Portal domain called, for example *webcenter.jks*. Then generate a key pair for the *webcenter.jks* keystore:

  keytool -genkeypair -keyalg RSA -dname "*consumer_dname*" -alias webcenter
  -keypass *key_password* -keystore *keystore* -storepass *keystore_password* -validity *days_valid*

  Where:
  - *consumer_dname* is the name of the consumer (for example, cn=spaces, dc=example, dc=com)
  - *key_password* is the password for the new public key, (for example, MyPassword)
  - *keystore* is the keystore name, (for example, *webcenter.jks*)
  - *keystore_password* is the keystore password, (for example, MyPassword)
  - *days_valid* is the number of days for which the key password is valid (for example, 1064)

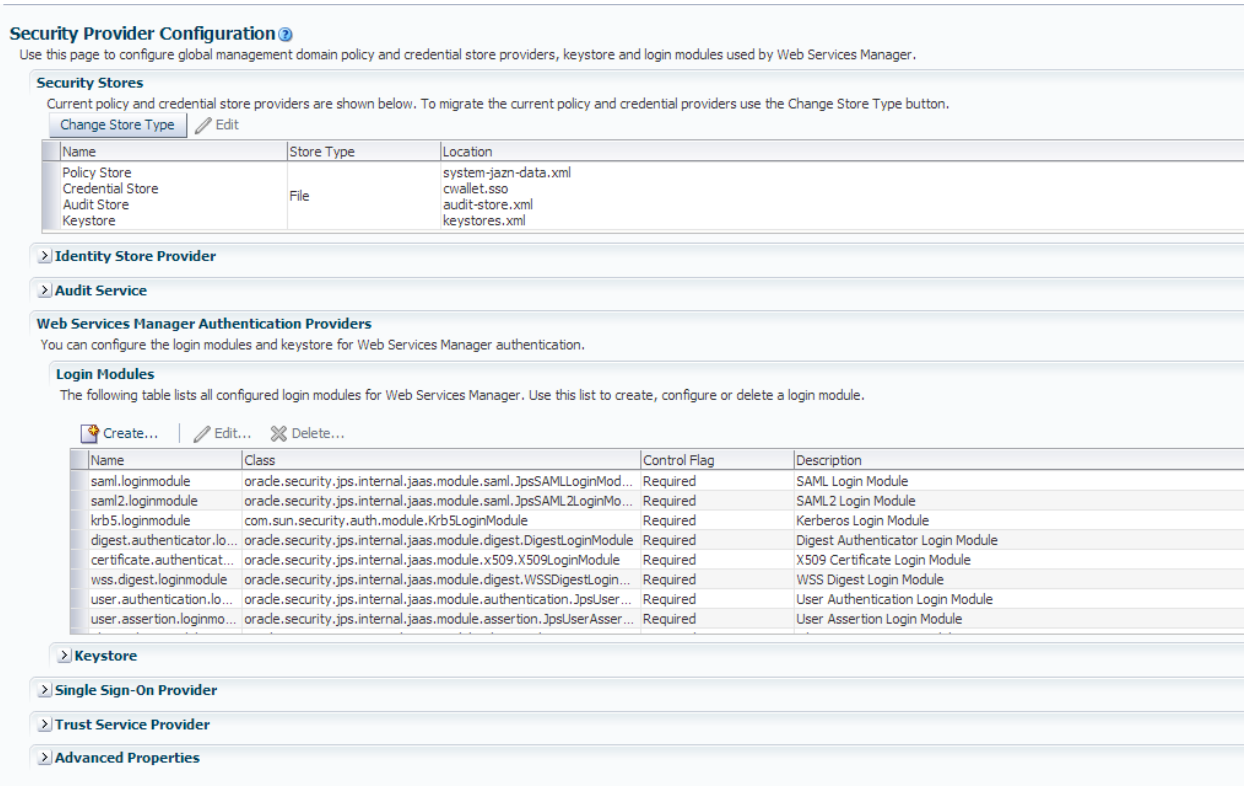
Configure the Keystore Service Using Fusion Middleware Control

Once you have created the WebCenter Portal domain keystore, configure the keystore service and update the credential store so that Oracle Web Services Manager (OWSM) can read the keys correctly.

To configure the keystore provider:

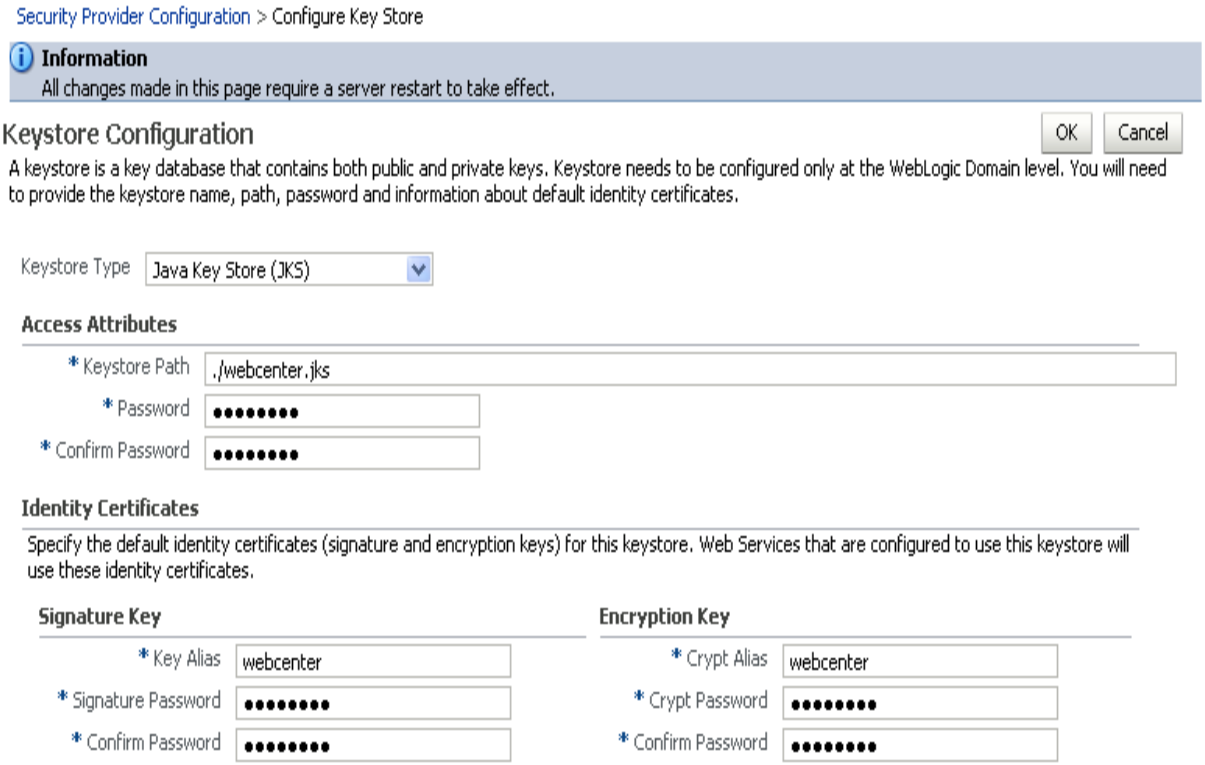
- Open Fusion Middleware Control and log into the WebCenter Portal domain (for information on logging in to Fusion Middleware Control, see Chapter 6, "Starting Enterprise Manager Fusion Middleware Control" in *Oracle Fusion Middleware Administering Oracle WebCenter Portal*.
- In the Navigation pane, expand the WebLogic Domain node and click the WebCenter Portal domain (by default, *wc_domain*).
- From the WebLogic Domain menu, select **Security -> Security Provider Configuration** to access the Security Provider Configuration page (for details see chapter 36 Configuring WS-Security in *Oracle Fusion Middleware Administering Oracle WebCenter Portal*).

Figure 1: Security Provider Configuration Page



- On the Security Provider Configuration page expand the Keystore section
- Click **Configure** to open The Keystore Configuration page (for details see Section 36.3.2 Setting Up the WebCenter Portal Domain Keystores in *Oracle Fusion Middleware Administering Oracle WebCenter Portal*).

Figure 2: Keystore Configuration Page



- Use the following settings to specify the location of the keystore that contains the certificate and private key, and the signature key and encryption key aliases:
    - **Keystore Path**: ./webcenter.jks
    - **Password**: Enter and confirm the password for the keystore.
    - **Key Alias**: webcenter
    - **Signature Password**: Enter and confirm the password for the signature key.
    - **Crypt Alias**: webcenter
    - **Crypt Password**: Enter and confirm the password for the encryption key.
- Click **OK** to save your settings.
- After you have configured the keystore, restart the Administration server for the domain.

Import the Certificate for the Sales Opportunity Web Service

To be able to access data using the sales opportunity web service, the X509 certificate for the service must be imported into the WebCenter Portal domain keystore:

- Obtain the sales opportunity web service WSDL URL for the Oracle Sales Cloud application. This URL should be similar to the following: http://host:port/opptyMgmtOpportunities/OpportunityService?WSDL

- Access the WSDL URL in the web browser to display the WSDL file. In the file, search for the X509 certificate section which should like the following:

  <dsig:X509Certificate>
  …Certificate…
  </dsig:X509Certificate>

- Copy the text between the *X509Certificate* tags and then save it into a file with the name *crm-cert.cer*. This is a public key for the sales opportunity web service. All out-going messages from WebCenter Portal should be encrypted with this public key to make secure the communication.
- Import the *crm-cert.cer* file into the WebCenter Portal domain's keystore.
  You can perform certificate import either via *keytool,* or from the Enterprise Manager.
  The following example shows the *keytool* command to import the certificate:

  ```
  keytool -importcert -file  crm-cert.cer -keystore webcenter.jks -
  alias crm –storepass keystore-password
  ```

- Restart WebCenter Portal to apply configuration changes.

Visualizing the Sales Opportunity Data

This section describes the steps to visualize the list of sales opportunities in a table format using WebCenter Portal's Data Presenter:

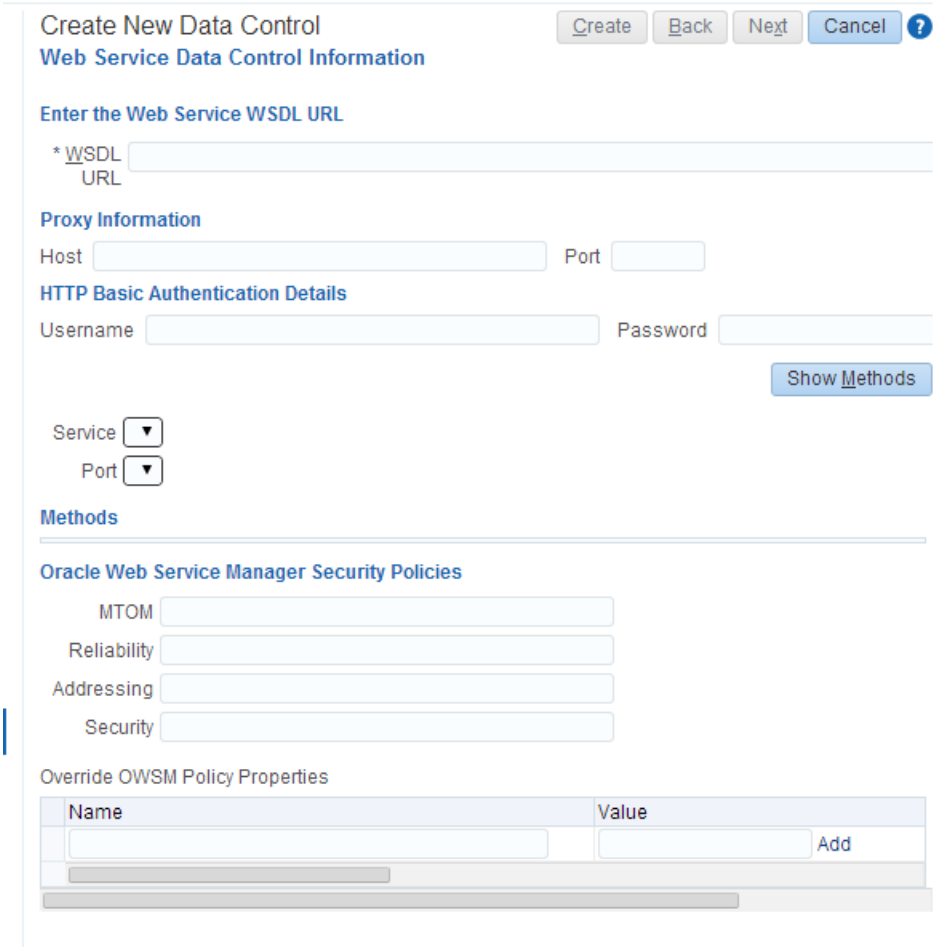
1. Create Web Service Data Control for the sales opportunity web service
2. Create the runtime task flow to visualize data in a table format

Create Sales Opportunity Web Service Data Control

Log in on WebCenter Portal and then create a new portal called **CRM Portal**. Once you have created the new portal, navigate to the **Administration > Assets** page for the portal. On the Assets page, select the **Data Controls** option and then click **Create** to start configuring the new data control.

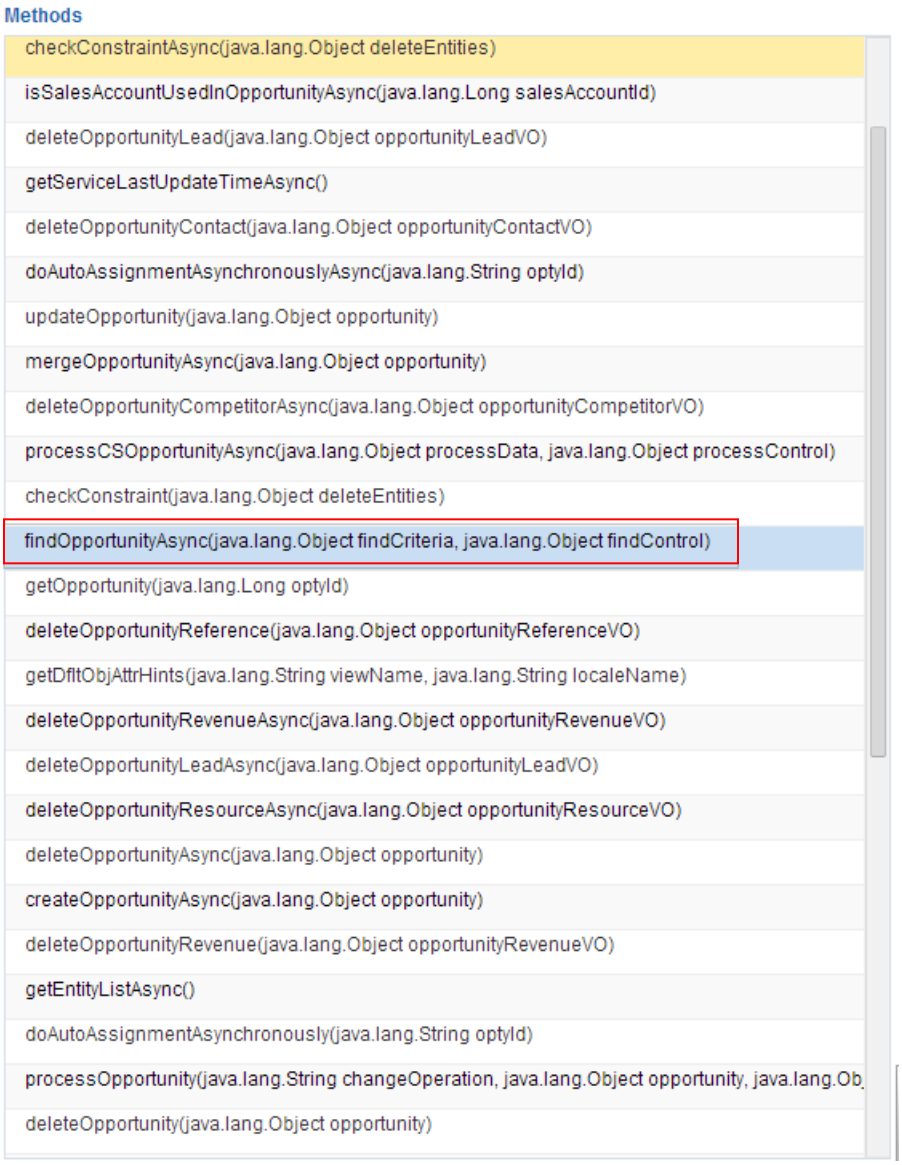
Name the data control **CRMOpportunity** and select **Web Service** as the data control type:

Click **Continue**.

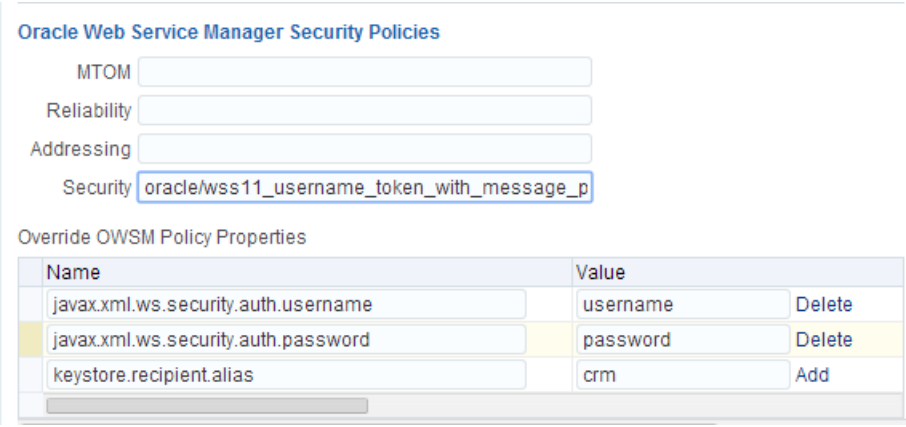


On the Web Service Data Control Information page provide details of the sales opportunity web service:

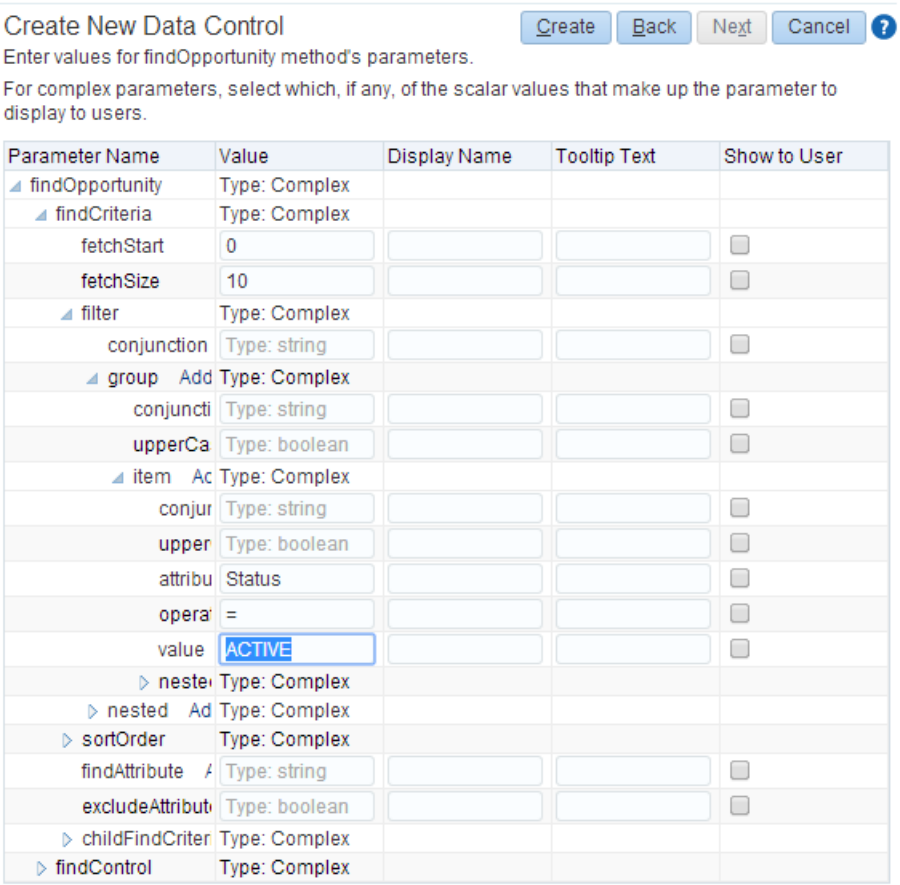
- Enter the **WSDL URL** for the sales opportunity web service
- Click **Show Methods** to get the list of methods exposed by this web service
- From the list select the **findOpportunity** method as shown below:



- Configure the web service security as follows:
  - In the **Security** field enter the name of security policy that is required by the Oracle Sales Cloud web service: *oracle/wss11_username_token_with_message_protection_client_policy policy*
  - In the Override OWSM Policy Properties section, enter the following properties:
    - *javax.xml.ws.security.auth.username* – the username of the System Account to access the Oracle Sales Cloud application
    - *javax.xml.ws.security.auth.password* – the password of the System Account to access the Oracle Sales Cloud application
    - *keystore.recipient.alias* - enter **crm**

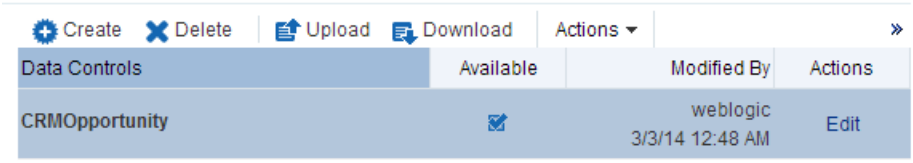


Click **Next** to proceed to the page that allows setting default values for the parameters that are required to invoke the web service method. For example, to get 10 sales opportunities with Status ACTIVE, set method parameters as shown below:



Click **Finish** to create new data control.

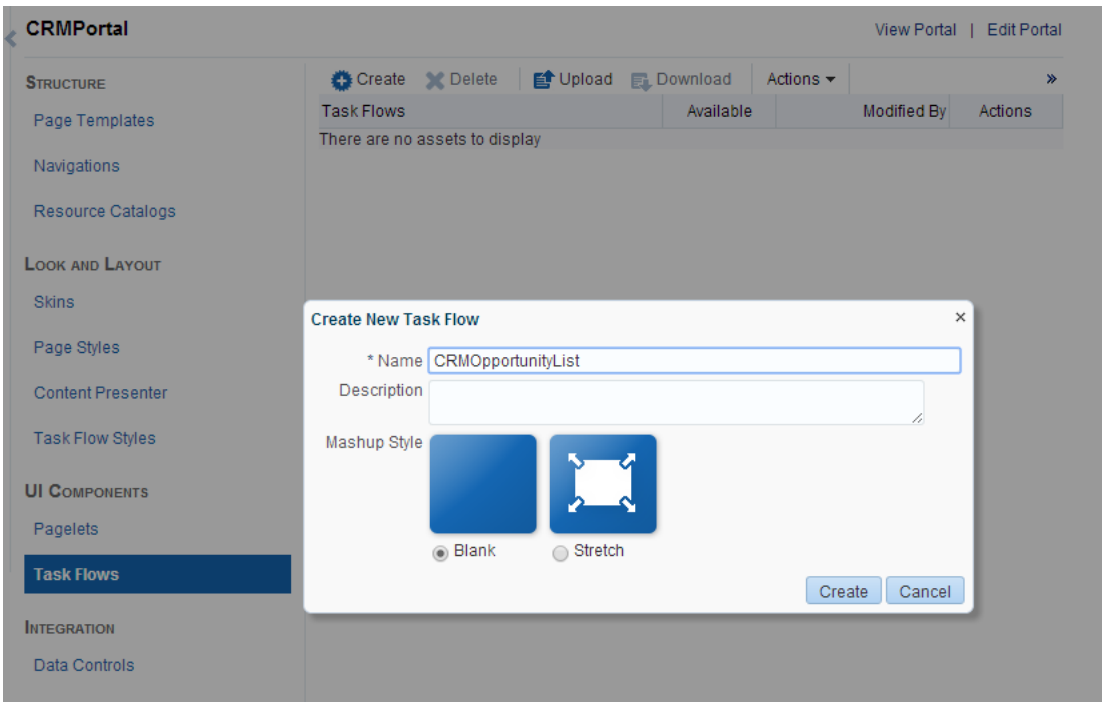
On the Data Controls page in the Assets area select the data control that you just created and select the **Available** check box to make the data control available for creating visualizations.



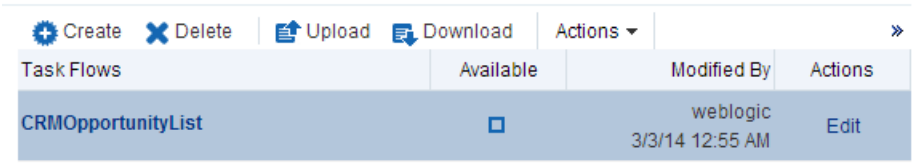
Create the Sales Opportunity Visualization

In this section, we use the CRM Opportunity data control that we created in the previous section to create runtime task flow to display the list of the sales opportunities.

To start, navigate to the Task Flows options in the portal's Assets page. Click **Create** to start configuring a new visualization. Enter the name for the new task flow. Keep the default **Blank** style and then click **Create**.

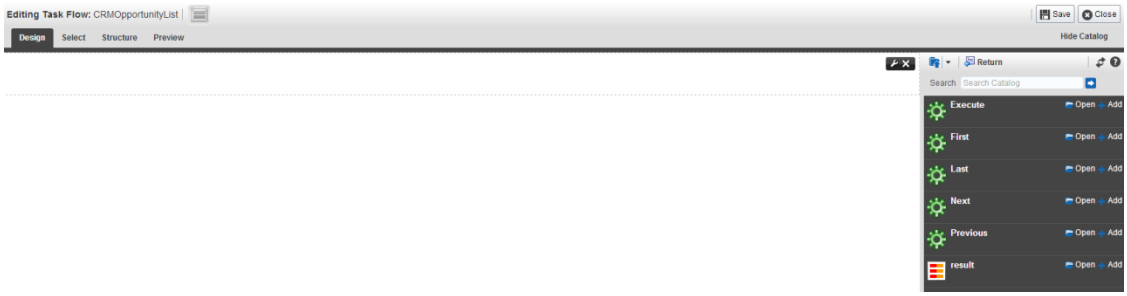


Once new task flow is created, select it and click **Edit** in the **Actions** column.

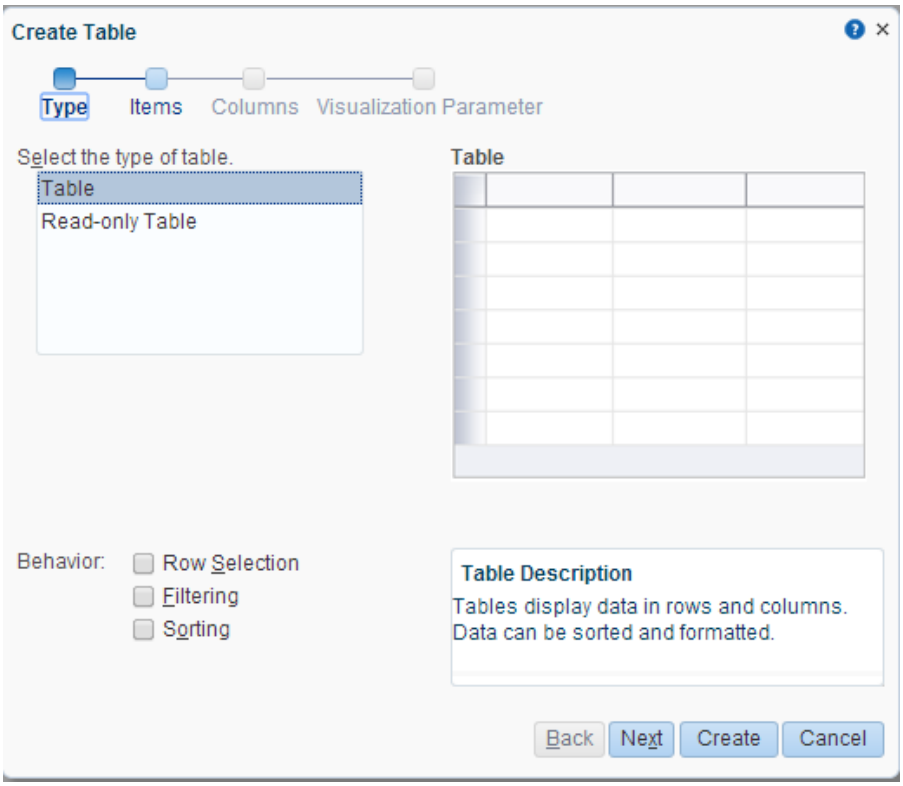


This opens the task flow in Composer Edit mode. The task flow content is shown on left and the resource catalogue on right. Since we picked the Blank task flow template, initially the task flow has no content. In the resource catalog open the **Integration** folder and then navigate to **Data**

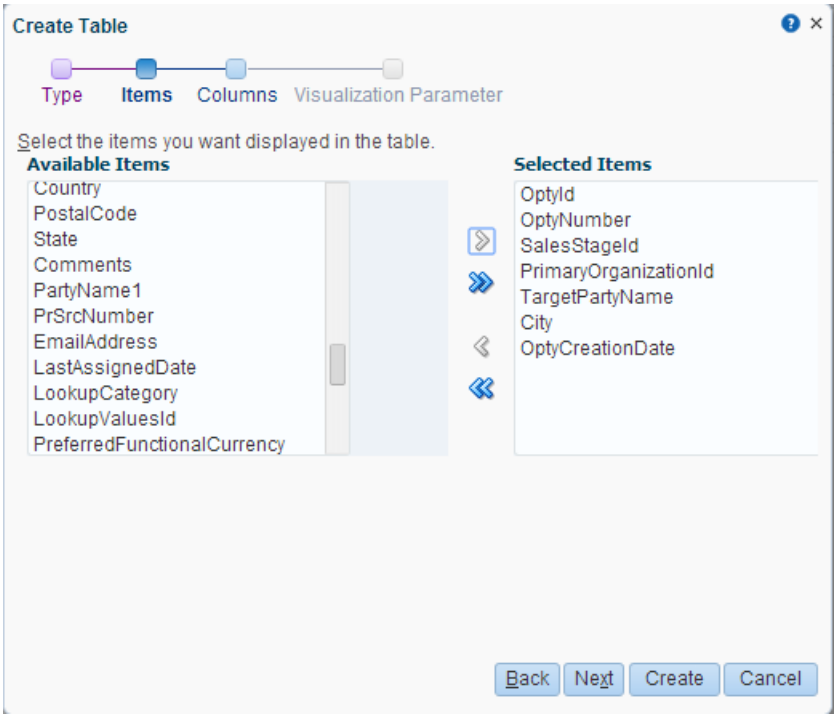
Controls -> CRMOpportunity -> findOpportunity -> Return. In the resource catalog you should see the following list:



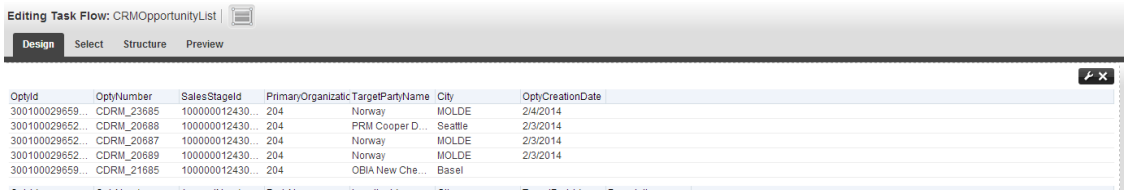
Click **Add** next to the Result. In the displayed contextual menu select **Table** as the type of visualization to use. This will open wizard to configure the table visualization as shown below:



Accept the default type and click **Next** to proceed to the wizard page that allows you to select the attributes in the sales opportunity web service response that you want to see in the table. For example, select the list of attributes as shown on the image below (for example, OptyId, OptyNumber, SalesStageId, PrimaryOrganizationId, TargetPartyName, City, and OptyCreationDate):



Click **Create** to generate the table visualization of the sales opportunities. Once the visualization is generated, you will see the following table in the task flow content area:



Click **Save** to save the task flow and close Composer. To make the new task flow available in the resource catalog select it on the list displayed on the Task Flows page in the Assets page and then select the **Available** check box. Once available, you can add this task flow to a portal page.

Visualizing Sales Opportunities Using the Oracle Security Token Service

So far we have described how to use a System Account that is shared by all portal users to integrate Oracle Sales Cloud data into WebCenter Portal. In this section we describe a different approach that you should use if the integration use case requires accessing sales data by giving individual user credentials. In this approach, the identity of each user is propagated separately via a SAML Bearer Token asserted by the Oracle Security Token Service (STS). The main steps to achieve this setup are as follows:

- Replicate all Sales Cloud users into the local LDAP repository (for example, OID) that is being used by WebCenter Portal.
- Install and configure Oracle STS.
- Configure WebCenter Portal to use Oracle STS to access data stored in the Oracle Sales Cloud.

Replicating Oracle Sales Cloud Users

There are two prerequisites to successfully configure user replication between the Oracle Sales Cloud and WebCenter Portal:

- The identity store configured for WebCenter Portal should contain the same set of users as that used in the Oracle Sales Cloud application, that is, each portal user should have the same user name as in the Sales Cloud. The user passwords for each application do not need to be identical.
- WebCenter Portal and Oracle STS should share same Oracle Identity Directory (OID) so that STS can create SAML assertion for the WebCenter Portal logged in user.

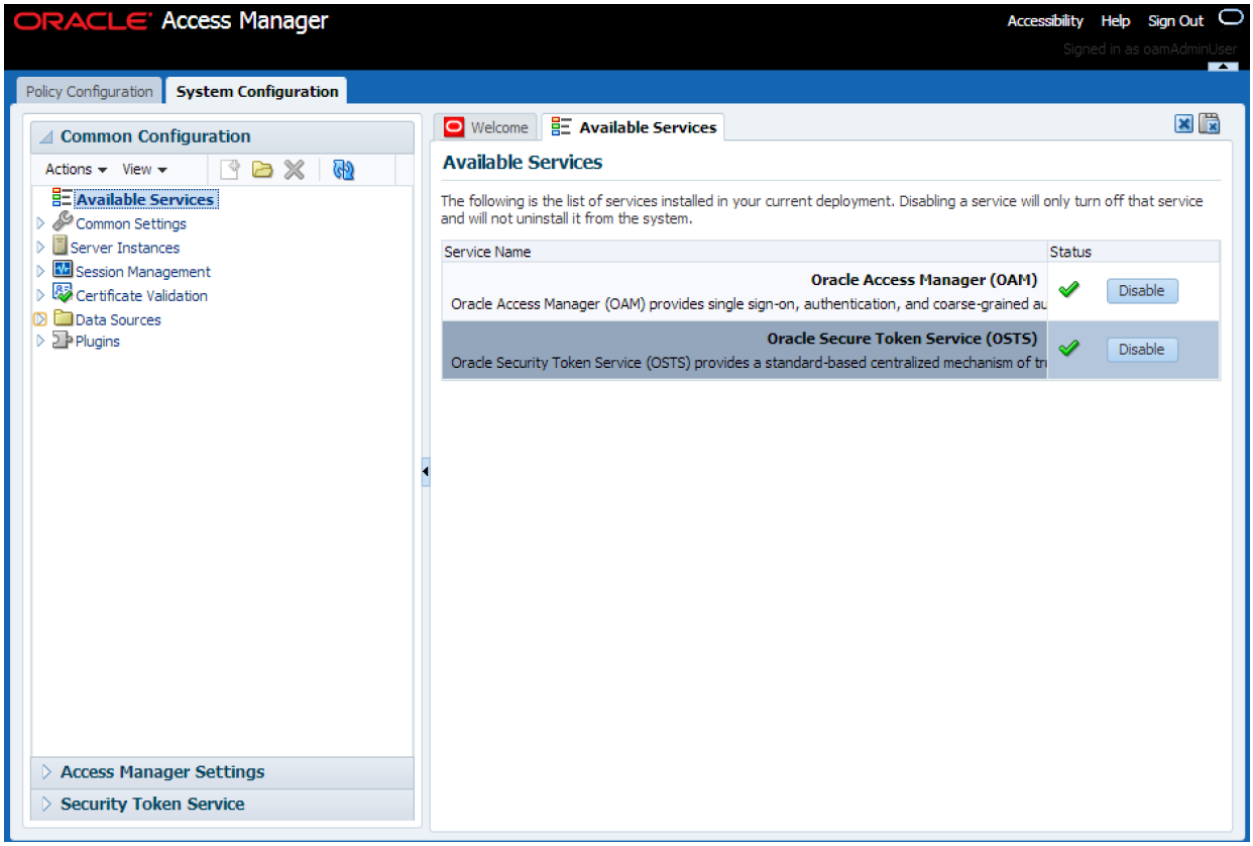
Setting Up and Configuring Oracle STS

Oracle STS is available as part of Oracle Access Manager (OAM) 11*g* R1 PS1 or later. For details of enabling and configuring Oracle STS in OAM, Part V Oracle Security Token Service in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager with Oracle Security Token Service*.

Enable Oracle STS

Out of the box, STS is disabled in OAM. To enable STS, perform the following steps:

- Go to the OAM Admin Console: http://wlsadminhost:wlsadminport/oamconsole
- Log in as the OAM Administrator (for example, orcladmin/welcome1)
- Navigate to **System Configuration >Common Configuration >Available Services:**
  - Click the **System Configuration** tab.
  - The Common Configuration accordion group should be expanded by default.
  - Double click the **Available Services** list item.
- Click the **Enable** checkbox on the Security Token Service row.



Generate New STS Signing and Encryption Keys

This section describes how to generate new RSA key pairs with their corresponding certificates that are used by STS during runtime operations to sign outgoing messages, and decrypt incoming messages.

In order to create new key entries in the OAM JCE Keystore, you must perform one of the following actions on the keystore password:

- If OAM is version R1PSx, retrieve the password.
- If OAM is version R2 or later, reset the password.

To retrieve the OAM JCE Keystore password:

- Open a terminal window.
- Enter the WSLT scripting environment ($IAM_ORACLE_HOME/common/bin/wlst.sh).
- Connect to the WebLogic Server AdminServer, using the connect() command.
- Execute the following commands:
    - domainRuntime()
    - listCred(map="OAM_STORE", key="jks")
    - exit()
- Note the password.

To reset the OAM JCE Keystore password:

- Open a terminal window.
- Enter the WSLT scripting environment ($IAM_ORACLE_HOME/common/bin/wlst.sh).
- Connect to the WebLogic Server AdminServer, using the connect() command.
- Execute the following commands:
    - domainRuntime()
    - resetKeystorePassword (password="NEW_PASSWORD")
    - exit()

NOTE: There may be more than one wlst.sh in your environment. It is crucial to use the correct instance, which should be found in the OAM deployment. For example:

OAM/Middleware/Oracle_IDM1/common/bin/wlst.sh

Perform the following operations to generate the new keys:

- Open a terminal window.
- Navigate to $MW_HOME/JDK_HOME/bin (should be jrockitjdk1.6.0_24).
- Execute the following command to create the signing key:
  keytool –genkeypair -alias stssign –keyalg rsa –keysize 1024 –sigalg sha1withrsa –dname "cn=STS Signature" –validity 1200 –storepass OAM_JCE_KEYSTORE_PASSWORD -keystore $DOMAIN_HOME/config/fmwconfig/.oamkeystore keypass STS_SIGN_KEY_PASSWORD storetype JCEKS

  Where:
    - **$DOMAIN_HOME** is the path to the WLS Domain directory (which may be identified by the Domain.home parameter to the running server process)
    - **OAM_JCE_KEYSTORE_PASSWORD** is the OAM JCE Keystore password
    - **STS_SIGN_KEY_PASSWORD** is the password that should be used to protect the new STS Signing Key Entry
    - **OAM_JCE_KEYSTORE** is the keystore where the new key entry is located.
    - **alias** is the alias of the new key entry, stssign
    - **dname** is the name of the certificate, cn= STS Signature

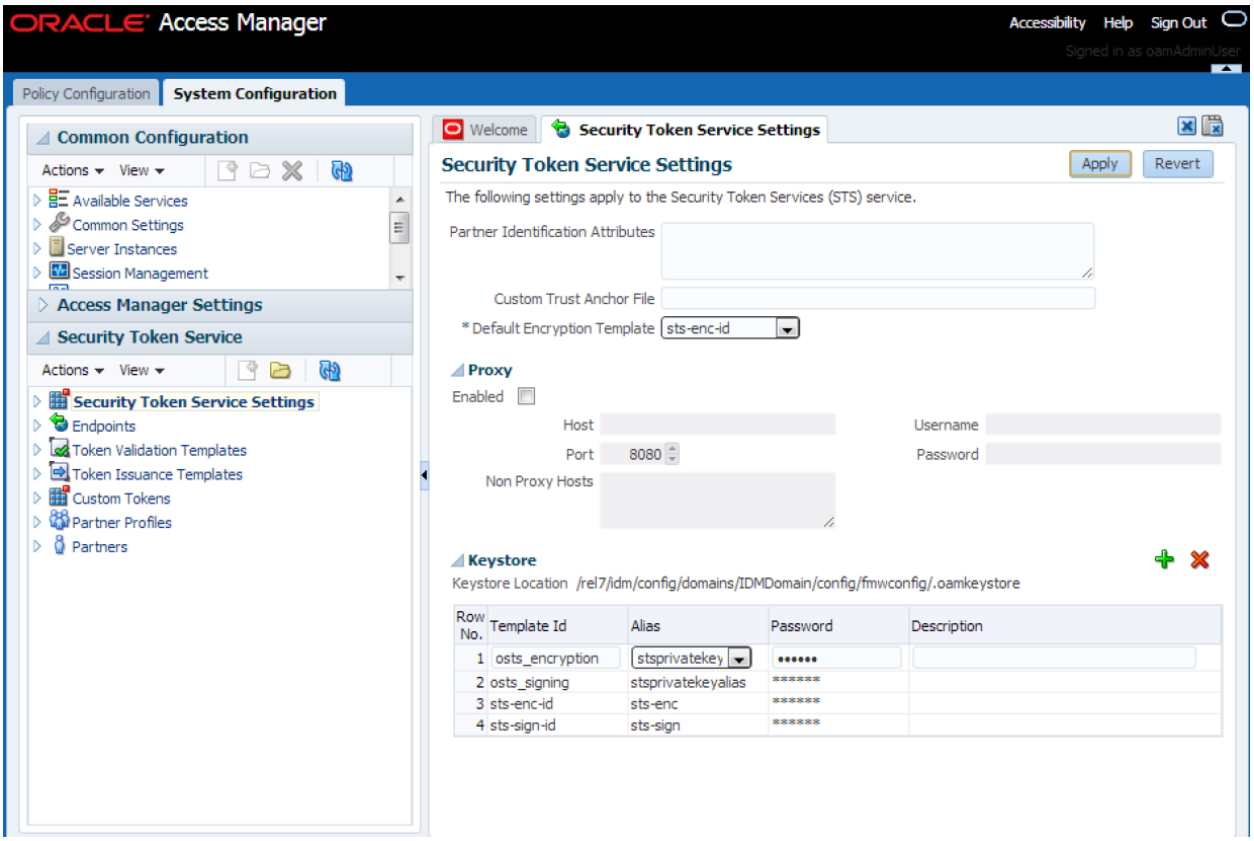
- Execute the following command to create the encryption key:
  keytool –genkeypair –alias stsenc keyalg rsa –keysize 1024 –sigalg sha1withrsa –dname "cn=STS Encryption" –validity 1200 –storepass OAM_JCE_KEYSTORE_PASSWORD –keystore $DOMAIN_HOME/config/fmwconfig/.oamkeystore –keypass STS_ENC_KEY_PASSWORD –storetype JCEKS

  Where:

- **$DOMAIN_HOME** is the path to the WLS Domain directory (which may be identified by the Domain.home parameter to the running server process)
- **OAM_JCE_KEYSTORE_PASSWORD** is the OAM JCE Keystore password
- **STS_ENC_KEY_PASSWORD** is the password that should be used to protect the new STS Encryption Key Entry
- **OAM_JCE_KYESTORE** is the keystore where the new key entry is located
- **alias** is the alias of the new key entry, stsenc
- **dname** is the name of the certificate, cn= STS Encryption

Finally, the keys must be defined in the STS configuration.

- Go to the OAM Admin Console: http://wlsadminhost:wlsadminport/oamconsole
- Log in as the OAM Administrator.
- Navigate to **System Configuration -> Security Token Service -> Security Token Service Settings**
- Click on the **Create** button in the Keystore table
- Enter **sts-sign-id** as the **Template ID**
- Select **sts-sign** as the alias from the drop down
- Enter the value for **OSTS_SIGN_KEY_PASSWORD** used during the generation of the **sts-sign** key entry
- Click on the **Create** button in the Keystore table
- Enter **sts-enc-id** as the **Template ID**
- Select **sts-enc** as the alias from the drop down
- Enter the value for **OSTS_ENC_KEY_PASSWORD** used during the generation of the **sts-enc** key entry
- Click **Apply**
- Select **sts-enc-id** as the Default Encryption Template
- Click **Apply**



Create an Issuance Template

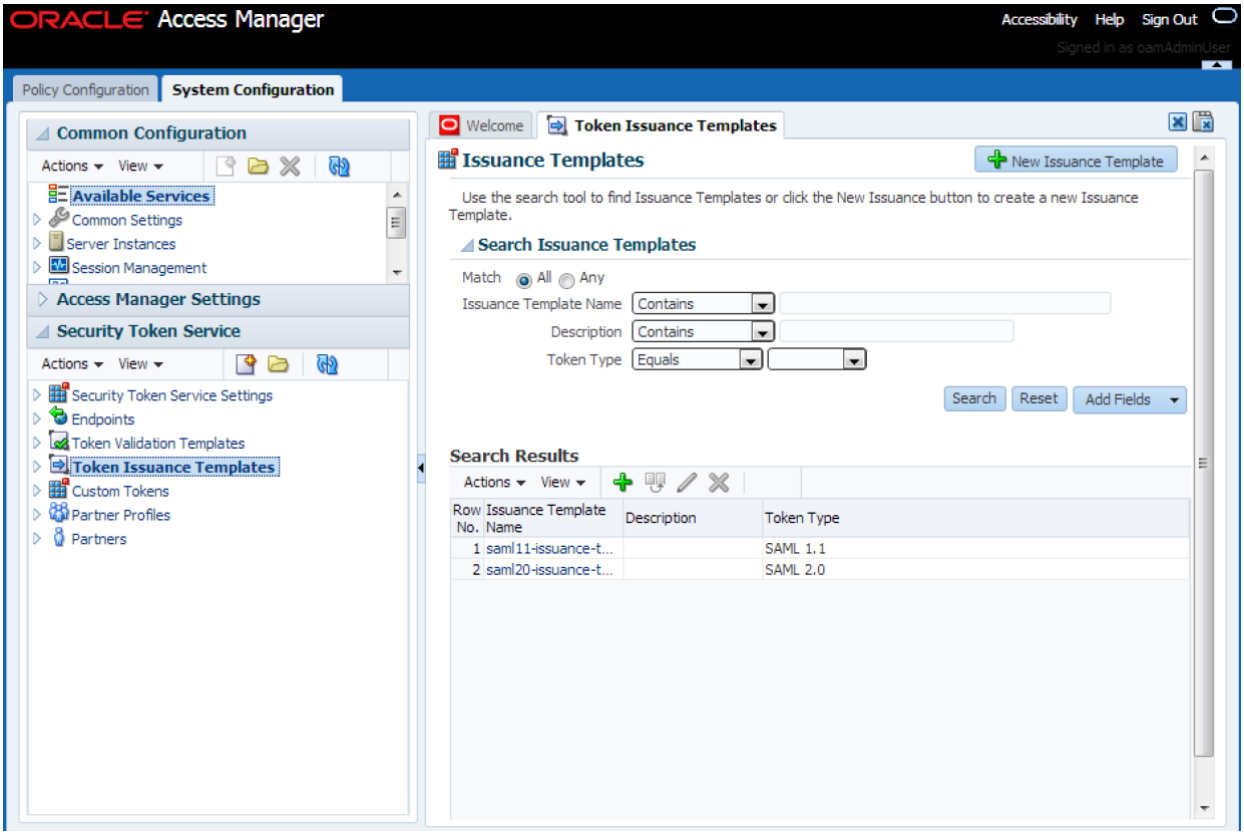
An STS issuance template contains the settings describing how a token should be issued by STS. This section describes how to create and configure a SAML 2.0 issuance template for a user located in the LDAP directory.

Perform the following steps:

- Go to the OAM Admin Console: http://wlsadminhost:wlsadminport/oamconsole
- Log in as the OAM Administrator.
- Navigate to **System Configuration >Security Token Service >Security Token Service Settings:**
    - Click the **System Configuration** tab.
    - Click the Security Token Service accordion group.
    - Double click the **Security Token Service Settings** list item.
- Click the **Create** button in the Keystore table (green plus sign in the top right of the table).
- Enter **stssignid** as the **Template ID**.
- Select **stssign** as the alias from the drop down list.

- Enter the value for **STS_SIGN_KEY_PASSWORD** used during the generation of the **stssign** key entry.
- Click the **Create** button in the Keystore table.
- Enter **stsencid** as the **Template ID**.
- Select **stsenc** as the alias from the drop down list.
- Enter the value for **STS_ENC_KEY_PASSWORD** used during the generation of the **stsenc** key entry.
- Click **Apply**.
- Select **stsencid** as the **Default Encryption Template** (third field from the top).
- Click **Apply**.

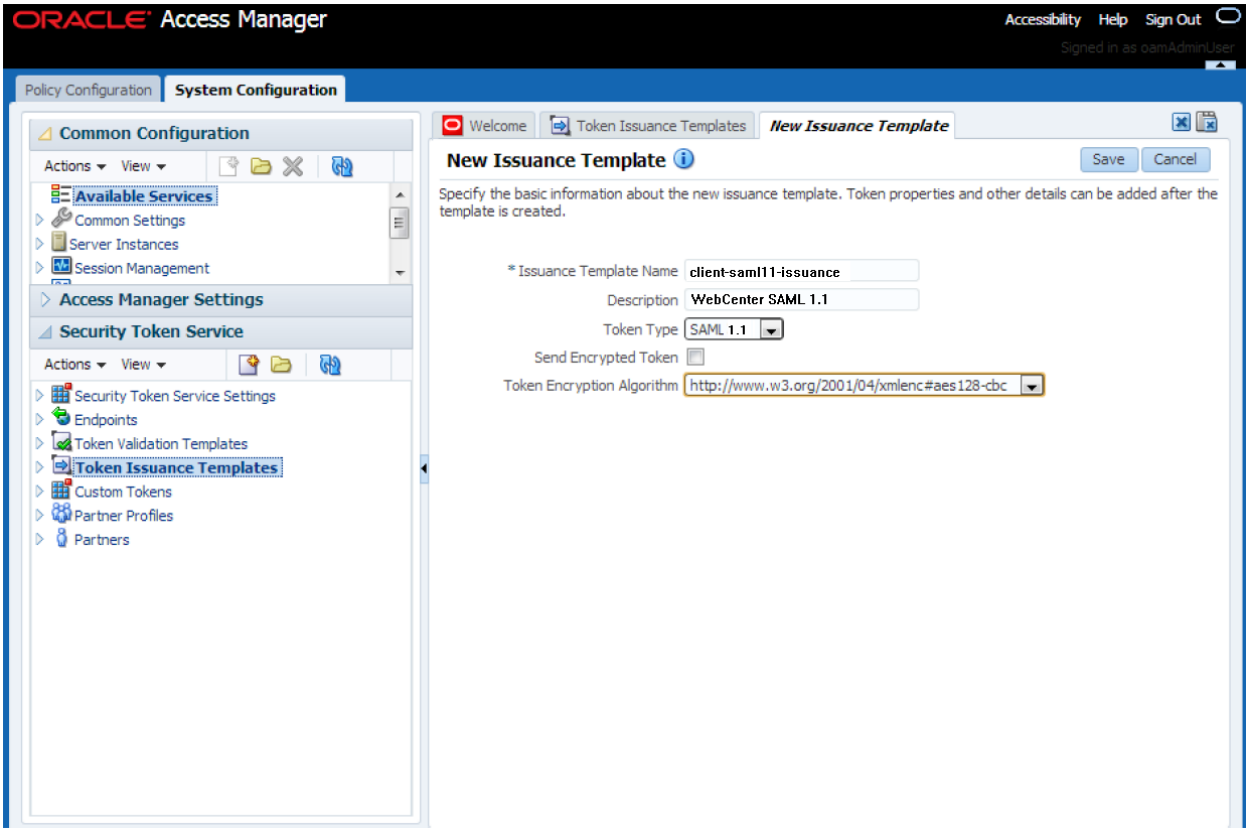
  NOTE: if **stsencid** doesn't appear in the drop down list, log out and then log back in, and it should appear.



- Click the **New Issuance Template** button.
- Enter the following data in the various fields:
  - **Issuance Template Name:** client-saml11-issuance
  - **Description:** Client

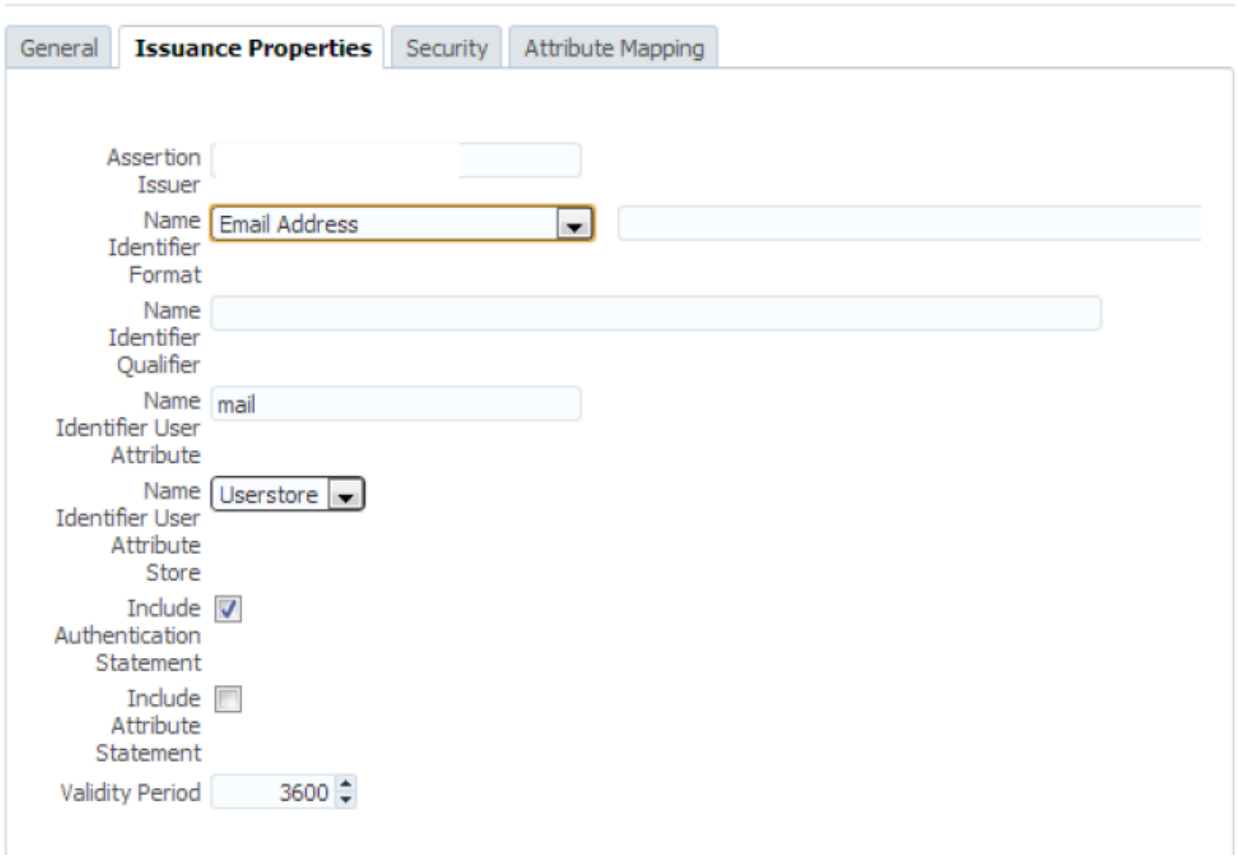
- **Token Type:** SAML 1.1
- **Send Encrypted Token:** unchecked
- **Token Encryption Algorithm:** http://www.w3.org/2001/04/xmlenc#aes128cbc

- Click **Save**.

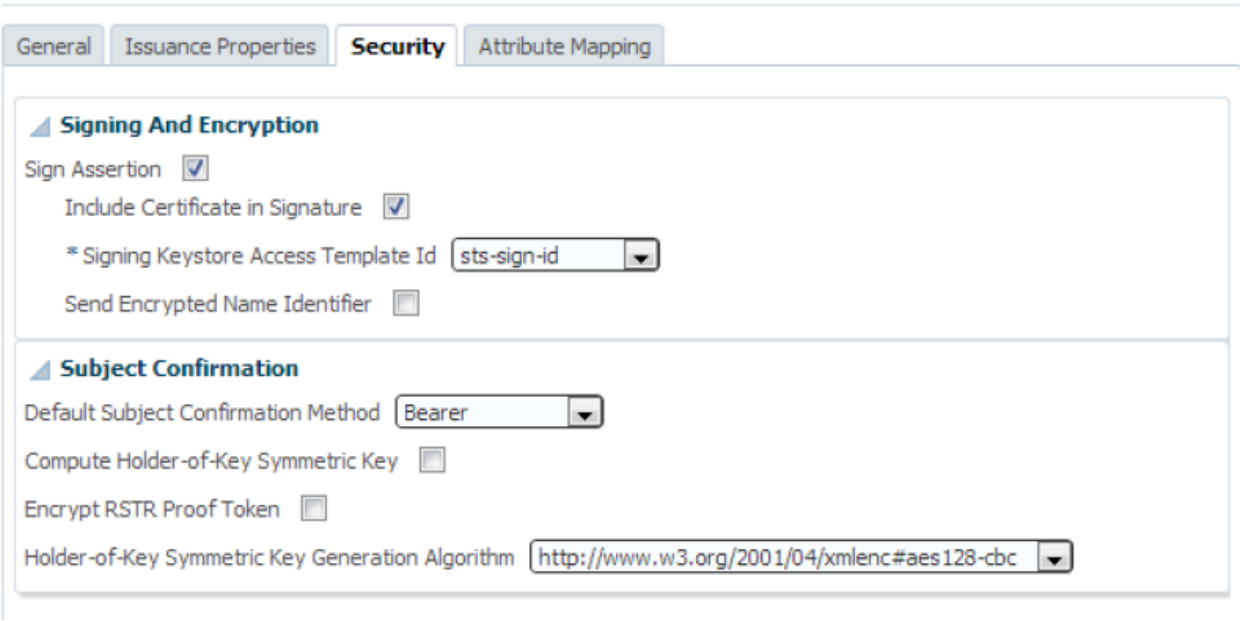


- Navigate to **System Configuration > Security Token Service > Token Issuance Templates.**
  - Click **Search**.
  - Click **client-saml11-issuance**.
  - Click **Issuance Properties**.
  - Enter the following data in the various fields:
    - Set the **Assertion's Issuer** to the public hostname of OAM/STS.
    - Leave the **Name Identifier Qualifier** field empty.
    - Select **Userstore** as the **Name Identifier User Attribute Store**.
    - Select **Include Authentication Statement**.
    - Deselect **Include Attribute Statement**.
    - Set **Validity Period** to 3600 seconds.
    - If the SAML Assertion's subject contains the user ID:
      - Set the **Name Identifier Format** as **Unspecified**.

        - Set the **Name Identifier User Attribute** to **uid**.
    - If the SAML Assertion's subject contains the user's email address:
        - Set the **Name Identifier Format** as **Email Address**.
        - Set the **Name Identifier User Attribute** to **mail**.



- Click the **Security** tab.
- Enter the following data in the various fields:
    - **Sign Assertion:** selected
    - **Include Certificate in Signature:** selected
    - **Signing Keystore Access Template ID:** stssignid
    - **Send Encrypted Name Identifier:** not selected
    - **Default Subject Confirmation Method:** Bearer
    - **Compute HolderofKey Symmetric Key:** not selected
    - **Encrypt RSTR Proof Token:** not selected
    - **HolderofKey Symmetric Key Generation Algorithm:** http://www.w3.org/2001/04/xmlenc#aes128cbc
- Click **Apply**.



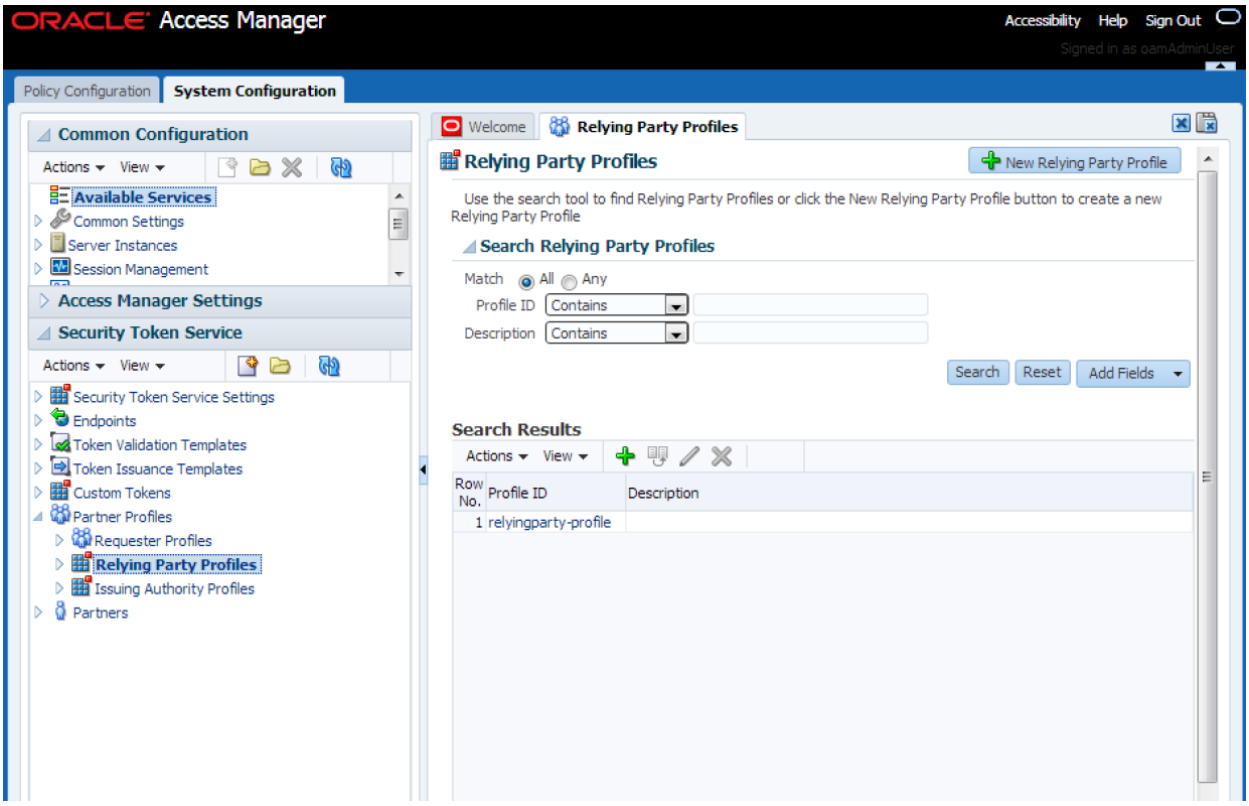
Create a Relying Party Profile

The Client Web Service mast be registered as a Relying Party partner in OSTS, and it is bound to a Relying Partner profile. This section describes how to create and configure the Relying Party Profile.

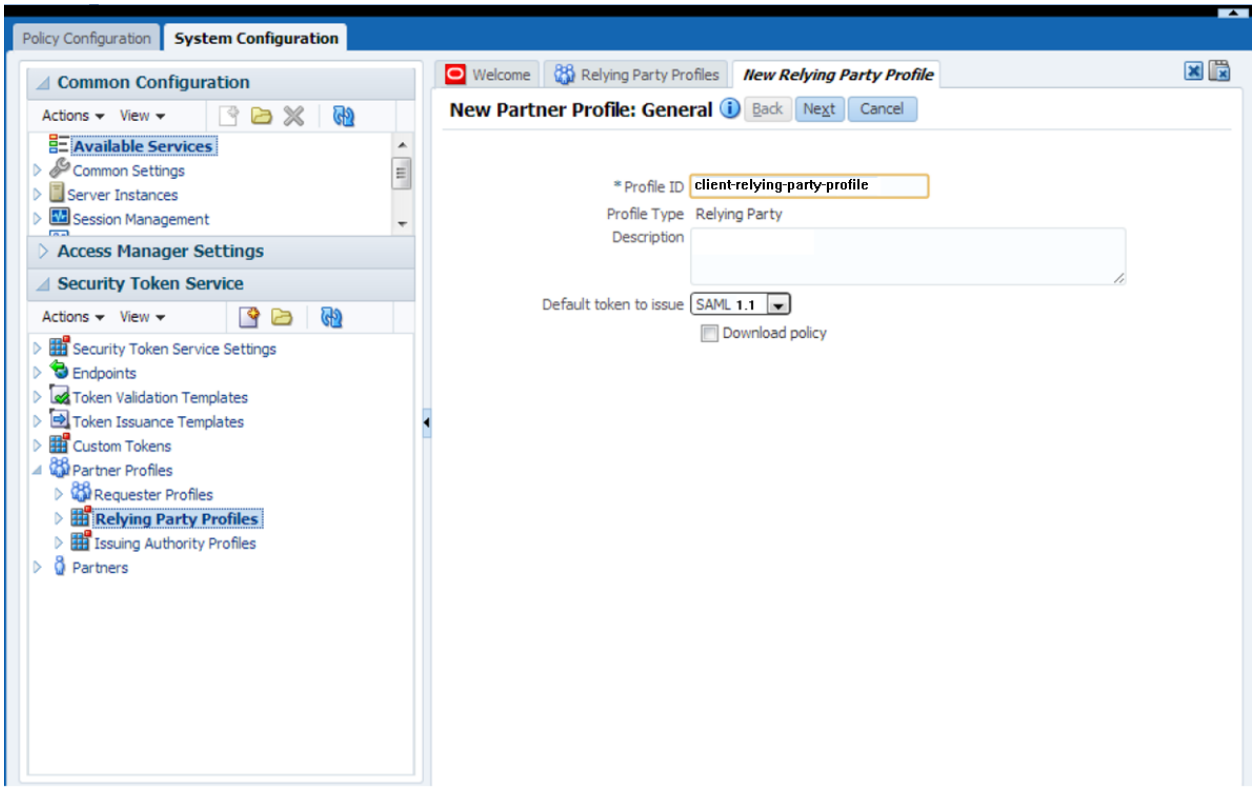
Perform the following steps:

- Go to the OAM Admin Console: http://wlsadminhost:wlsadminport/oamconsole
- Log in as the OAM Administrator.
- Navigate to **System Configuration > Security Token Service > Partner Profiles > Relying Party Profiles.**

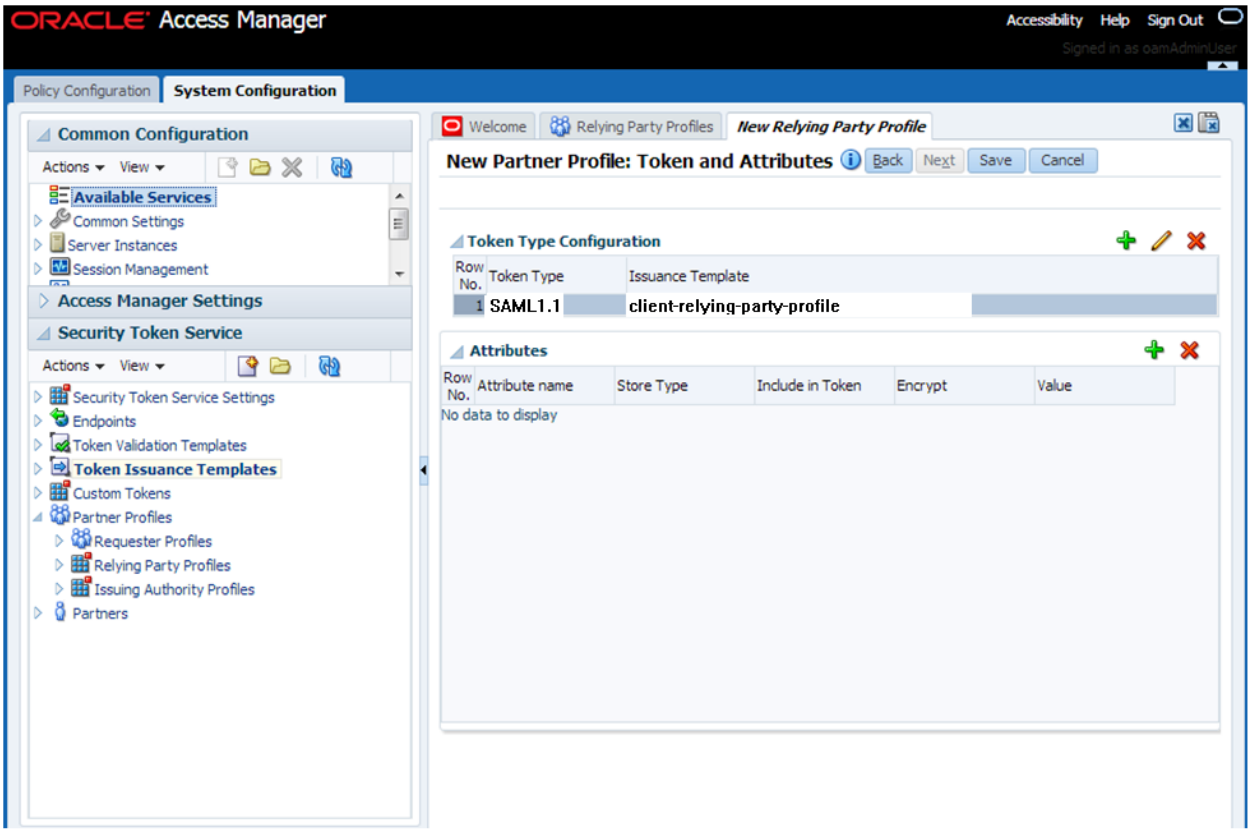
  NOTE: Relying Party Profiles in located under the Partner Profiles node.



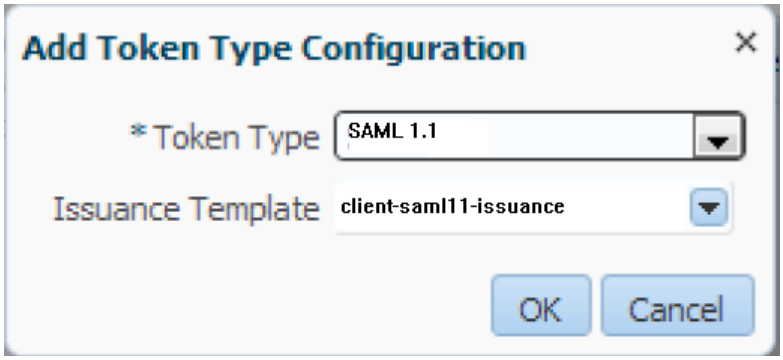
- Click the **New Relying Party Profile** button.



- Enter the following data in the various fields:
  - **Profile ID:** client-relying-party-profile
  - **Description:** Client
  - **Default Token to issue:** SAML 1.1
  - **Download Policy:** not selected
- Click **Next**.



- Click **Add** in the Token Type Configuration.



- In the popup window:
  - Select **SAML 1.1** as the **Token Type**.
  - Select **client-saml11-issuance** as the **Issuance Template**.

- Click **OK**.

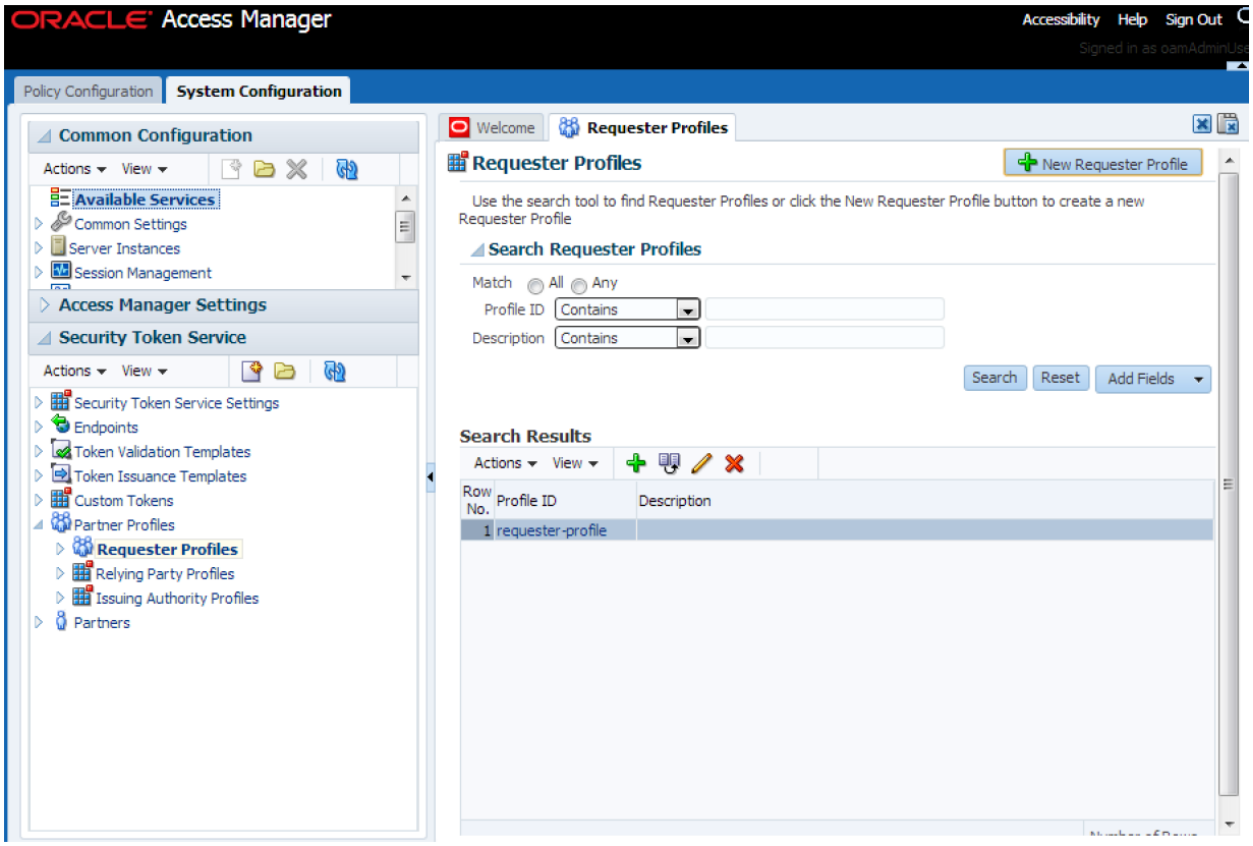
- Click **Save**.

Create a Requester Profile

The clients are not registered as OSTS Requester partners, since those clients are users stored in the LDAP directory. OSTS applies a default requester profile for those clients that will be evaluated at runtime, and that default profile contains configuration data used for processing the incoming request. This section describes how to create the default requester profile.

Perform the following operations:

- Go to the OAM Admin Console: http://wlsadminhost:wlsadminport/oamconsole
- Log in as the OAM Administrator
- Navigate to **System Configuration > Security Token Service > Partner Profiles > Requester Profiles.**



- Click the **New Requester Profile** button.
- Enter the following data in the various fields:
  - **Profile ID:** WebCenterPortal-requester-profile
  - **Description:** Client
  - **Default Relying Party Profile:** client-relying-party-profile

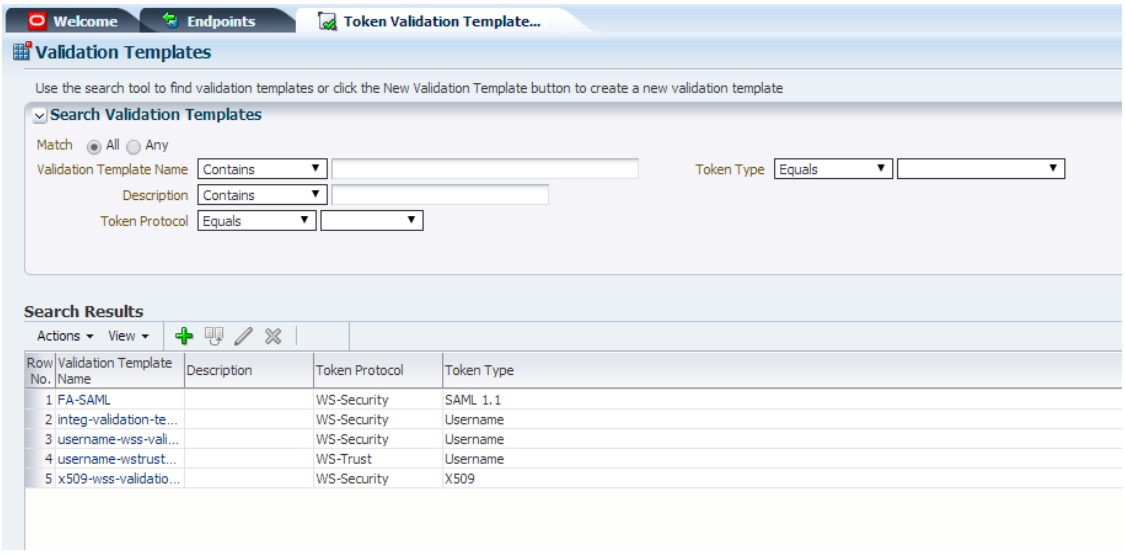
- Click **Next**.
- Click **Save**.

Create the SAML 1.1. Validation Template

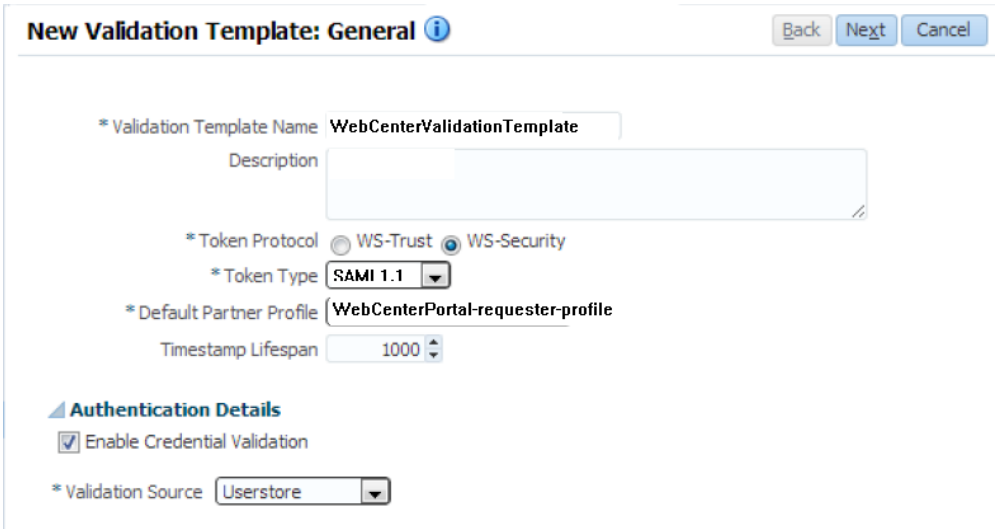
This section describes how to create and configure the validation template that is used to verify the SAML assertion sent by WebCenter Portal.

Perform the following operations:

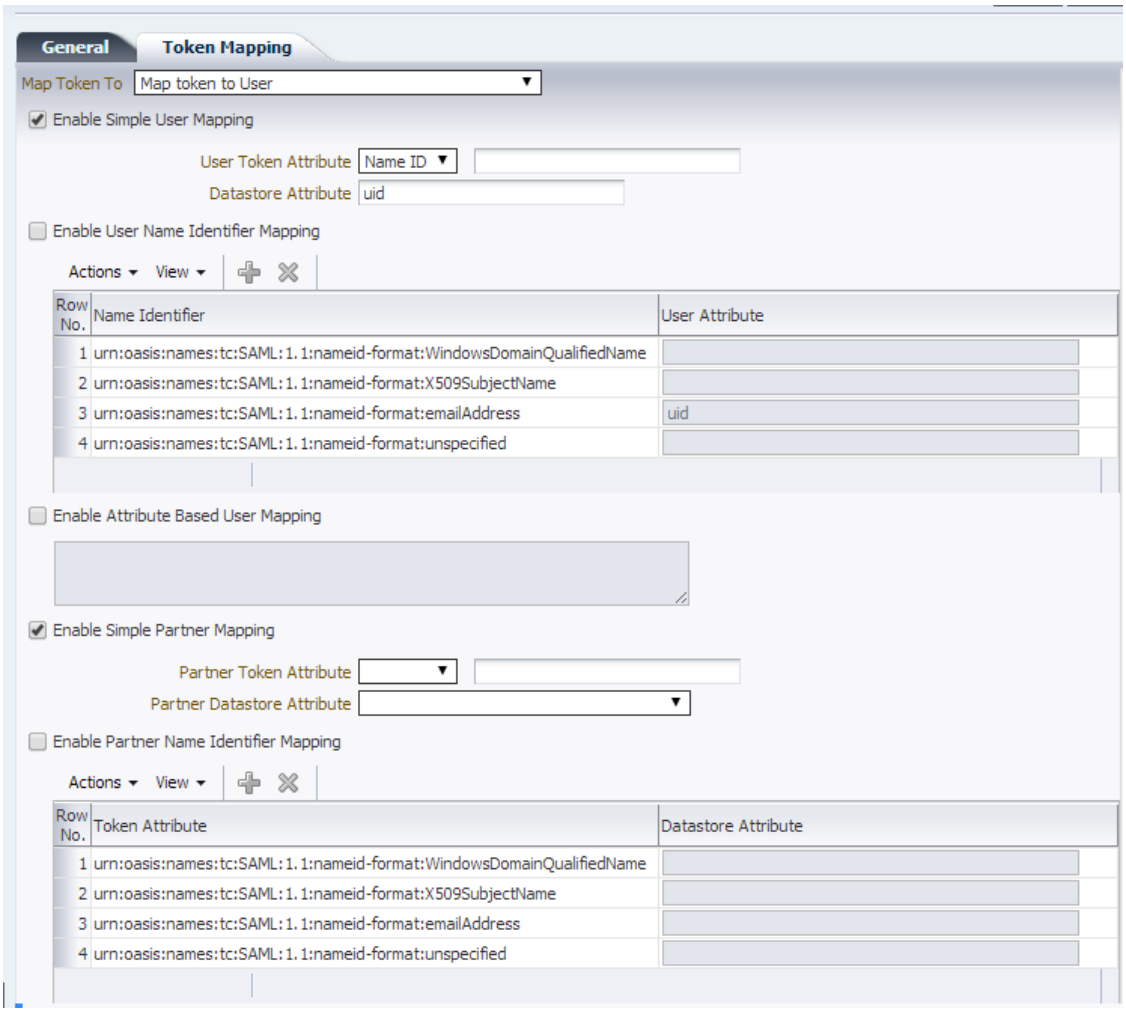
- Go to the OAM Admin Console: http://wlsadminhost:wlsadminport/oamconsole
- Log in as the OAM Administrator.
- Navigate to **System Configuration > Security Token Service > Token Validation.**



- Click the **New Validation Template** button.



- Enter the following data in the various fields:
  - **Validation Template Name:** WebCenterValidationTemplate
  - **Description:** Client
  - **Token Protocol:** WSSecurity
  - **Token Type:** SAML1.1
  - **Default Partner Profile:** WebCenterPortal-requester-profile
  - **Timestamp Lifetime:** 1000 seconds
  - **Enable Credential Validation:** checked
  - **Validation Source:** Userstore
- Click **Next** to get to following screen.



On this screen, it is important to specify the user name attribute for OID. If the WebCenter Portal user is using **uid** as the login name then specify the database attribute

name as **uid**. If the user name attribute is different from **uid** then update this field accordingly.

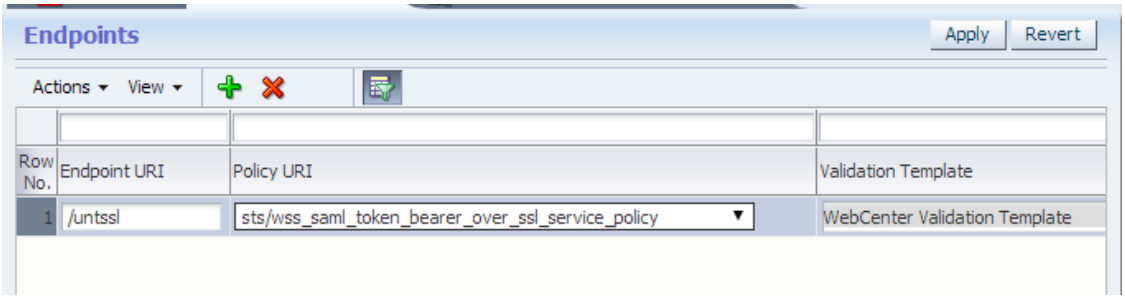
- Click **Apply**.

Configure the OSTS Endpoints

This section describes how to configure the OSTS endpoints, that is, by adding the new endpoint that should be used by the WebCenter Portal to invoke OSTS at https://oamhost:oamport/sts/untssl

Perform the following operations:

- Go to the OAM Admin Console: http://wlsadminhost:wlsadminport/oamconsole
- Log in as the OAM Administrator.
- Navigate to **System Configuration > Security Token Service > Endpoints.**



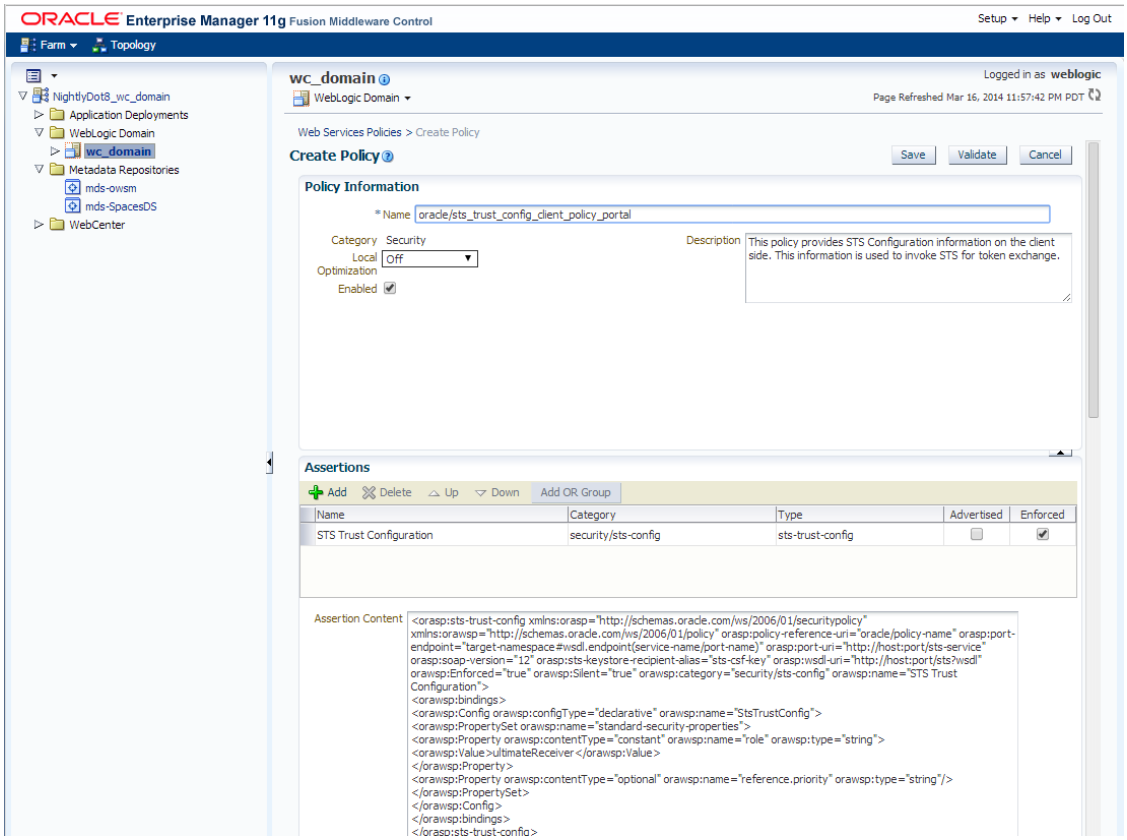
Perform the following operations:

- Click **Add**
- Enter **/untssl** as the endpoint URI
- Set **Policy URI** to **sts/wss_saml_token_bearer_over_ssl_service_policy**.
- Set **WebCenter Validation Template** to the template created in the previous section as the WebCenter Validation Template.

This completes the configuration on Oracle STS.

Configuring WebCenter Portal to Use Oracle STS

In this section, we create a custom OWSM policy to configure the STS configuration policy from the web service client. Use Fusion Middleware Control to create a new policy called **oracle/sts_trust_config_client_policy_portal** from an existing **oracle/sts_trust_config_client_policy** policy (see "Creating a Web Service Policy from an Existing Policy" in the *Oracle Fusion Middleware Security and Administrator's Guide for Web Services*).



Provide the following information:

- Add the actual endpoint URI of the STS **for port-uri.** For example: https://sts-host:sts-port/sts/untssl
- Add the client policy URI to be used by the client to communicate with the STS for **policy-reference-uri**. Set this to **oracle/wss_saml_token_over_ssl_client_policy**.
- Add the endpoint of the web service, specified as **target-namespace#wsdl.endpoint(service-name/port-name)** for **wsdl-uri**.
- Add **sts-keystore-recipient-alias** that is added to keystore for Alias of STS Certificate. The default alias name is **sts-csf-key**.

For example:

```
<orasp:sts-trust-config
 xmlns:orasp="http://schemas.oracle.com/ws/2006/01/securitypolicy"
 xmlns:orawsp="http://schemas.oracle.com/ws/2006/01/policy"
 orasp:policy-reference-
uri="oracle/wss10_username_token_with_message_protection_client_policy"
 orasp:port-endpoint="target-namespace#wsdl.endpoint(service-name/port-name)"
 orasp:port-uri="http://host:port/sts-service" orasp:soap-version="12"
 orasp:sts-keystore-recipient-alias="sts-csf-key"
 orasp:wsdl-uri="http://host:port/sts?wsdl" orawsp:Enforced="true"
 orawsp:Silent="true" orawsp:category="security/sts-config" orawsp:name="STS Trust
Configuration">
<orawsp:bindings>
```

```
<orawsp:Config orawsp:configType="declarative" orawsp:name="StsTrustConfig">
<orawsp:PropertySet orawsp:name="standard-security-properties">
<orawsp:Property orawsp:contentType="constant" orawsp:name="role"
orawsp:type="string">
<orawsp:Value>ultimateReceiver</orawsp:Value>
</orawsp:Property>
</orawsp:PropertySet>
</orawsp:Config>
</orawsp:bindings>
</orasp:sts-trust-config>
```

Click **Save**.

Conclusion

After reading this white paper you have learned about the options available for integrating WebCenter Portal and Cloud-based Oracle Applications to enable portal components to access data stored in the application. Step-by-step instructions provided in the paper explained how to configure security and integrate WebCenter Portal deployed on-premise with Cloud based Oracle Applications using either of the following two approaches:

- On-premise WebCenter Portal is integrated with Oracle Sales Cloud application using a fixed System Account
- On-premise WebCenter Portal is integrated with Oracle Sales Cloud application using Identity Federation

The documented sample uses a Sales Opportunity web service, but the same approach can be used with any service exposed in the Oracle Sales Cloud API.

White Paper Title
[March] 2014
Author: Nitin Shah and Igor Polyakov
Contributing Authors: Ray Polk

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Oracle is committed to developing practices and products that help protect the environment

Copyright © 2014, Oracle and/or its affiliates. All rights reserved.

This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 0114

Hardware and Software, Engineered to Work Together