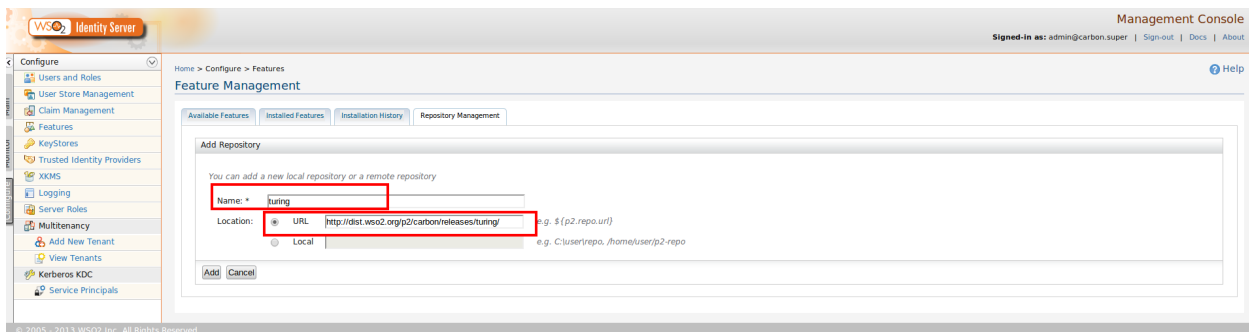


Configuring IS

1. Download pack form <http://wso2.com/products/identity-server/>
2. Run the pack (Refer : <http://docs.wso2.org/display/IS460/Getting+Started>)
3. After the server is started install the Key Manager feature. For more information refer: <http://docs.wso2.org/display/IS460/Installing+Features>
 - a. Go to the feature manager and add the following feature repository as shown below.
P2 Repo: <http://dist.wso2.org/p2/carbon/releases/turing/>



- b. After adding, find the feature in that repository. (Click on “find features” button).
 - i. Select “Key Manager feature” from “Api Manager 1.6.0” features category.
 - ii. Select “Registry Extension Feature” feature from “Governance registry” features category. (both are shown in the below diagram)

WSO2 Identity Server Management Console

Signed-in as: admin@carbon.super | Sign-out | Docs | About

Home > Configure > Features

Feature Management

Available Features | Installed Features | Installation History | Repository Management

Available Features

Find new features or updates to installed features in available repositories

Repository: [Add Repository](#)

Filter by feature name:

☐ Show only the latest versions ☒ Group features by category

[Find Features](#)

Select all in this page | Select none [Install](#)

Features	Version	Actions
<input type="checkbox"/> API Manager 1.5.0		
<input type="checkbox"/> API Manager 1.6.0		
<input type="checkbox"/> Gateway Manager		
<input type="checkbox"/> Gateway Worker		
<input checked="" type="checkbox"/> Key Manager		
<input type="checkbox"/> Publisher		
<input type="checkbox"/> Store		
<input type="checkbox"/> Application Server 5.2.0		
<input type="checkbox"/> Application Server 5.2.1		
<input type="checkbox"/> Application Server Runtimes (Optional)		
<input type="checkbox"/> Complex Event Processor		
<input type="checkbox"/> Data Services Server 3.1.0		
<input type="checkbox"/> Data Services Server 3.1.1		
<input type="checkbox"/> Enterprise Service Bus		
<input type="checkbox"/> Governance Registry		
<input type="checkbox"/> Application Management		
<input type="checkbox"/> Asset Store		
<input type="checkbox"/> Gadget Dashboard		
<input type="checkbox"/> governance Styles		
<input type="checkbox"/> Miscellaneous		
<input type="checkbox"/> Monitoring		
<input type="checkbox"/> Registry		
<input type="checkbox"/> Event Common	4.2.0	More Info
<input type="checkbox"/> Event Server	4.2.0	More Info
<input type="checkbox"/> Event Server	4.2.0	More Info
<input type="checkbox"/> Governance Metadata	4.2.1	More Info
<input type="checkbox"/> Registry Activities	4.2.0	More Info
<input type="checkbox"/> Registry Associations/Dependencies	4.2.0	More Info
<input type="checkbox"/> Registry Community Features	4.2.1	More Info
<input checked="" type="checkbox"/> Registry Extensions Feature -	4.2.0	More Info
<input type="checkbox"/> Registry Handler Config	4.2.0	More Info
<input type="checkbox"/> Registry JCR	4.2.0	More Info
<input type="checkbox"/> Registry Social	4.2.0	More Info
<input type="checkbox"/> Registry Task	4.2.0	More Info
<input type="checkbox"/> Registry UDDI	4.2.0	More Info
<input type="checkbox"/> Registry UI Menu for Governance	4.2.0	More Info
<input type="checkbox"/> Registry WS API	4.2.0	More Info
<input type="checkbox"/> WS-Discovery Core	4.2.1	More Info
<input type="checkbox"/> WS-Discovery Core	4.2.1	More Info
<input type="checkbox"/> WS-Discovery Managed Mode Proxy	4.2.1	More Info
<input type="checkbox"/> Registry/Repository		
<input type="checkbox"/> Reporting		
<input type="checkbox"/> Repository		
<input type="checkbox"/> Single Sign On(SAML2)		
<input type="checkbox"/> Tenancy Management		
<input type="checkbox"/> User Management		
<input type="checkbox"/> Identity Server		
<input type="checkbox"/> Product Styles		
<input type="checkbox"/> Storage Server 1.1.0		

Select all in this page | Select none [Install](#)

© 2005 - 2013 WSO2 Inc. All Rights Reserved.

- c. Click on “Install” button and go through the wizard to complete the installation. Finally you should see the following successful message.

WSO2 Identity Server Management Console

Signed-in as: admin@carbon.super | Sign-out | Docs | About

Home > Configure > Features

Feature Management

Available Features | Installed Features | Installation History | Repository Management

Installation Complete

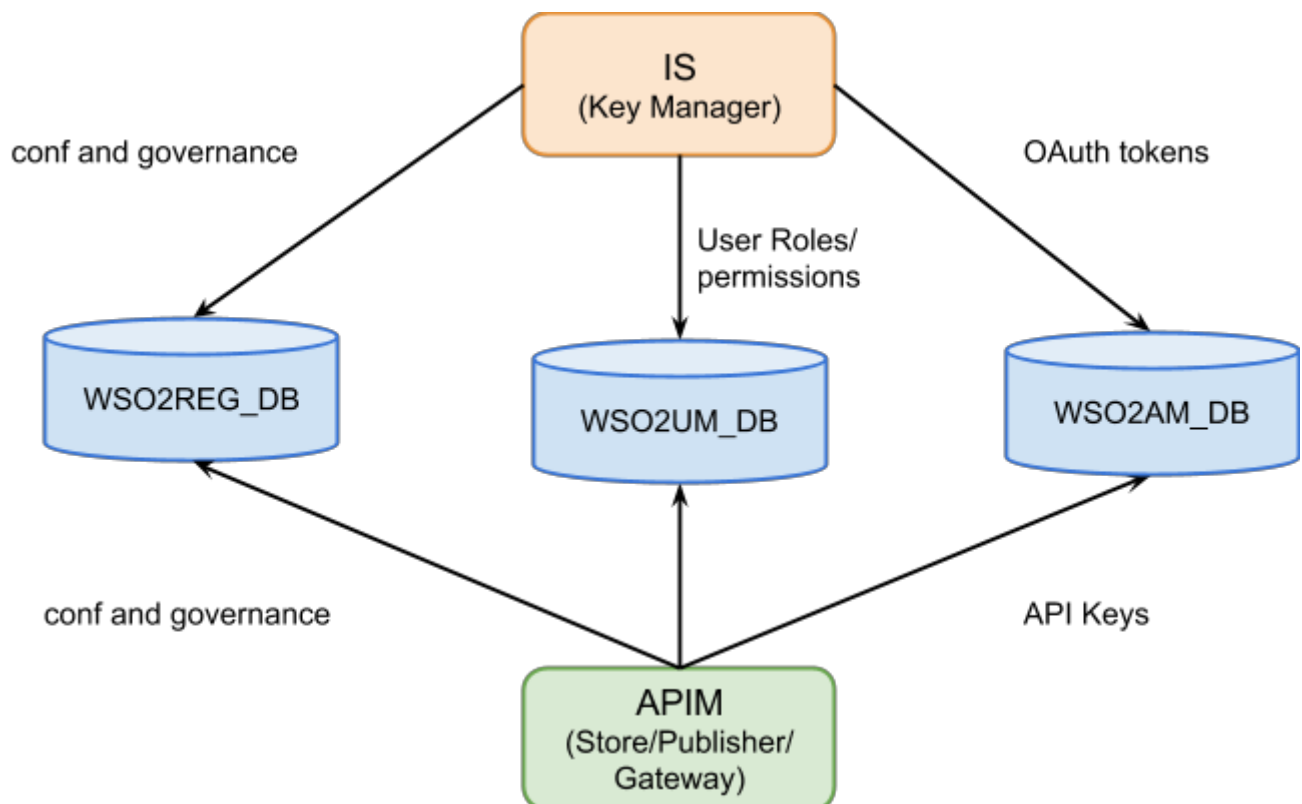
Successfully installed the selected features.

Restart the server.

[Finish](#)

- d. Copy the api-manager.xml from an API Manager 1.6.0 pack into IS_HOME/repository/conf (replace the one available there)
- e. Then change the GatewayType property as following,
`<GatewayType>None</GatewayType>`
- f. Restart Identity Server

Following diagram illustrate how databases are shared between IS and APIM



- *WSO2REG_DB* - This is used to keep the registry information
- *WSO2UM_DB* - This is used to keep the permissions (i.e permission store). Also this will be used to keep the Internal roles.
- *WSO2AM_DB* - This will be used to keep the identity data and api related data. OAuth tokens, Keys will be also in this. When serving key-validation requests, key manager validates whether there are subscriptions made by the particular key. For this *WSO2AM_DB* should be accessed.

4. Open the IS/repository/conf/datasources/master-datasource.xml file and add the following data sources. This is where we define the above databases as data sources so that they can be used in the servers.

```

<datasource>
  <name>WSO2AM_DB</name>
  <description>The datasource used for API Manager database</description>
  <jndiConfig>
    <name>jdbc/WSO2AM_DB</name>
  </jndiConfig>
  <definition type="RDBMS">
    <configuration>
<url>jdbc:mysql://localhost:3306/apimgt?autoReconnect=true&relaxAutoCommit=true</url>
      <username>apiuser</username>
      <password>apimanager</password>
      <driverClassName>com.mysql.jdbc.Driver</driverClassName>
      <maxActive>50</maxActive>
      <maxWait>60000</maxWait>
      <testOnBorrow>true</testOnBorrow>
      <validationQuery>SELECT 1</validationQuery>
      <validationInterval>30000</validationInterval>
    </configuration>
  </definition>
</datasource>

<datasource>
  <name>WSO2REG_DB</name>
  <description>The datasource used for registry and user manager</description>
  <jndiConfig>
    <name>jdbc/WSO2REG_DB</name>
  </jndiConfig>
  <definition type="RDBMS">
    <configuration>
<url>jdbc:mysql://localhost:3306/registry?autoReconnect=true&relaxAutoCommit=true</url>
      <username>apiuser</username>
      <password>apimanager</password>
      <driverClassName>com.mysql.jdbc.Driver</driverClassName>
      <maxActive>50</maxActive>
      <maxWait>60000</maxWait>
      <testOnBorrow>true</testOnBorrow>
      <validationQuery>SELECT 1</validationQuery>
      <validationInterval>30000</validationInterval>
    </configuration>
  </definition>
</datasource>

<datasource>
  <name>WSO2UM_DB</name>
  <description>The datasource used for registry and user manager</description>
  <jndiConfig>
    <name>jdbc/WSO2UM_DB</name>
  </jndiConfig>
  <definition type="RDBMS">
    <configuration>
<url>jdbc:mysql://localhost:3306/userstore?autoReconnect=true&relaxAutoCommit=true
      </url>
      <username>apiuser</username>
      <password>apimanager</password>

```

```

        <driverClassName>com.mysql.jdbc.Driver</driverClassName>
        <maxActive>50</maxActive>
        <maxWait>60000</maxWait>
        <testOnBorrow>true</testOnBorrow>
        <validationQuery>SELECT 1</validationQuery>
        <validationInterval>30000</validationInterval>
    </configuration>
</definition>
</datasource>

```

5. Replace the IS_HOME/repository/conf/registry.xml by the registry.xml in APIM_HOME/repository/conf/ folder. Handler to evaluate xacml media type is not present in the copied registry.xml. Therefore add the following handler

```

<handler
class="org.wso2.carbon.identity.entitlement.policy.finder.registry.RegistryPolicyHandler">
    <filter
class="org.wso2.carbon.identity.entitlement.policy.finder.registry.RegistryPolicyMediaTypeMatcher">
        <property name="mediaType">application/xacml-policy+xml</property>
    </filter>
</handler>

```

6. Create the registry mounts. Open IS/repository/conf/registry.xml and insert the following sections.

```

<dbConfig name="govregistry">
    <dataSource>jdbc/WSO2REG_DB</dataSource>
</dbConfig>

<remoteInstance url="https://localhost">
    <id>gov</id>
    <dbConfig>govregistry</dbConfig>
    <readOnly>false</readOnly>
    <enableCache>true</enableCache>
    <registryRoot></registryRoot>
</remoteInstance>

<mount path="/_system/governance" overwrite="true">
    <instanceId>gov</instanceId>
    <targetPath>/_system/governance</targetPath>
</mount>

<mount path="/_system/config" overwrite="true">
    <instanceId>gov</instanceId>
    <targetPath>/_system/config</targetPath>
</mount>

```

7. Change the datasource in identity.xml (IS/repository/conf/identity.xml) from jdbc/WSO2CarbonDB → jdbc/WSO2AM_DB
8. Open the user-mgt.xml and change the permission datassource from

jdbc/WSO2CarbonDB → jdbc/WSO2UM_DB

Note: Add the correct LDAP configuration

9. Open api-manager.xml and change the ThriftClientPort (under APIKeyManager section) to 10398.
10. Creating Databases: Create the following databases in the MySQL database server;
 - a. userstore
 - b. registry
 - c. apimgt

For creating ustore and regdb, use the script inside IS_HOME/dbscripts/mysql.sql.

When creating apimgt db, both the scripts; APIM_HOME/dbscripts/apimgt/mysql.sql and IS_HOME/dbscripts/identity/mysql.sql should be used. Script inside APIM/dbscripts/apimgt/ only have tables needed to manage Oauth access tokens. To make all the other identity related features working, the script inside IS/dbscripts/identity too should be used. Therefore create a combined script from both these two and use it.

Create a user 'apiuser' with password 'apimanager' . Grant all accesses to above three data bases to this user.

ex:

```
grant all on apimgt.* TO apiuser@localhost identified by "apimanager";
grant all on userstore.* TO apiuser@localhost identified by "apimanager";
grant all on registry.* TO apiuser@localhost identified by "apimanager";
```

11. JWT configuration must be done at IS. For **JWT Token generation** please follow :
<http://docs.wso2.org/pages/viewpage.action?pageId=32350971>

Configuring API Manager

1. Add the three data sources into master-datasources.xml
2. Open user-mgt.xml and change the permission datassource from
jdbc/WSO2CarbonDB → jdbc/WSO2UM_DB

Note: Add the correct LDAP configuration (same one as in IS)

3. Open registry.xml and add the following;

```
<dbConfig name="govregistry">
  <dataSource>jdbc/WSO2REG_DB</dataSource>
</dbConfig>

<remoteInstance url="https://localhost">
  <id>gov</id>
```

```

    <dbConfig>govregistry</dbConfig>
    <readOnly>false</readOnly>
    <enableCache>true</enableCache>
    <registryRoot></registryRoot>
</remoteInstance>

<mount path="/_system/governance" overwrite="true">
    <instanceId>gov</instanceId>
    <targetPath>/_system/governance</targetPath>
</mount>

<mount path="/_system/config" overwrite="true">
    <instanceId>gov</instanceId>
    <targetPath>/_system/config</targetPath>
</mount>

```

4. Open api-manager.xml

- a. Change the *ServerUrl* of the *AuthManager* to point to IS

Change

```
<ServerURL>https://${carbon.local.ip}:${mgt.transport.https.port}/services/</ServerURL>
```

to

```
<ServerURL>https://${carbon.local.ip}:{port}/services/</ServerURL>
```

- b. Change the *ServerUrl* of the *APIKeyManager* to point to IS

Change

```
<ServerURL>https://${carbon.local.ip}:${mgt.transport.https.port}/services/</ServerURL>
```

to

```
<ServerURL>https://${carbon.local.ip}:{port}/services/</ServerURL>
```

- c. Change the *KeyValidatorClientType* from ThriftClient to WSCClient
- d. Change the *ThriftServerPort* (under *APIKeyManager* section) to 10398
- e. Change the *ThriftServerHost* to point to Identity servers IP address
- f. Change *EnableThriftServer* to true

Note: Add MySQL jdbc driver to both servers:

i.e. put the jar into CARBON_HOME/repository/components/lib