

# CA Identity Portal

## Installation Guide

1.5.1

# CA Technologies Product References

---

This document references the following CA Technologies products:

- CA Identity Governance
- CA Identity Manager
- CA Single Sign On
- CA User Activity Reporting
- CA Service Desk Manager
- CA IAM Connector Server

## Contact CA Technologies

---

### Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At <http://ca.com/support>, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

### Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to [techpubs@ca.com](mailto:techpubs@ca.com).

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at <http://ca.com/docs>.

# CONTENTS

---

<b>INSTALLATION OVERVIEW .....</b>	<b>4</b>
Components .....	4
Software Requirements .....	5
Hardware Requirements .....	7
Network Requirements .....	7
DNS Requirements .....	8
Cluster Requirements.....	8
<b>INSTALLING CA IDENTITY PORTAL ON AN APPLICATION SERVER .....</b>	<b>11</b>
Installation Overview .....	11
Installation Prerequisites.....	11
Installing Using the CA Identity Portal Installer.....	19
Manual Installation .....	21
Installing CA Identity Portal in a Cluster.....	33
<b>POST INSTALLATION .....</b>	<b>34</b>
CA Identity Manager Environment Validation .....	34
Import CA Identity Portal Roles and Tasks into CA Identity Manager Environment.....	34
Task Configuration in CA Identity Manager Environment.....	34
Copy Workpoint Client JARS (from CA Identity Manager Application Server to CA Identity Portal) .....	34
Set Endorsed Libraries Override (JBoss Only) .....	36
<b>CA IDENTITY PORTAL AND SINGLE-SIGN-ON (SSO).....</b>	<b>38</b>
Supported SSO Products .....	38
Background .....	38
SSO Prerequisites .....	40
CA Identity Governance, CA Identity Portal and SSO .....	44
<b>APPENDIX A – ENABLING XA TRANSACTIONS FOR MS SQL .....</b>	<b>45</b>
Configuring XA Transactions for SQL Server 2008 R2.....	45
<b>APPENDIX B – TEWS SETTINGS FOR CA IDENTITY PORTAL WITHOUT SSO .....</b>	<b>47</b>
<b>APPENDIX C – CONFIGURING JBOSS WITH A MSSQL DATASOURCE .....</b>	<b>48</b>
<b>APPENDIX D – CA IDENTITY PORTAL &amp; IDM WITHOUT PROVISIONING .....</b>	<b>50</b>

# Installation Overview

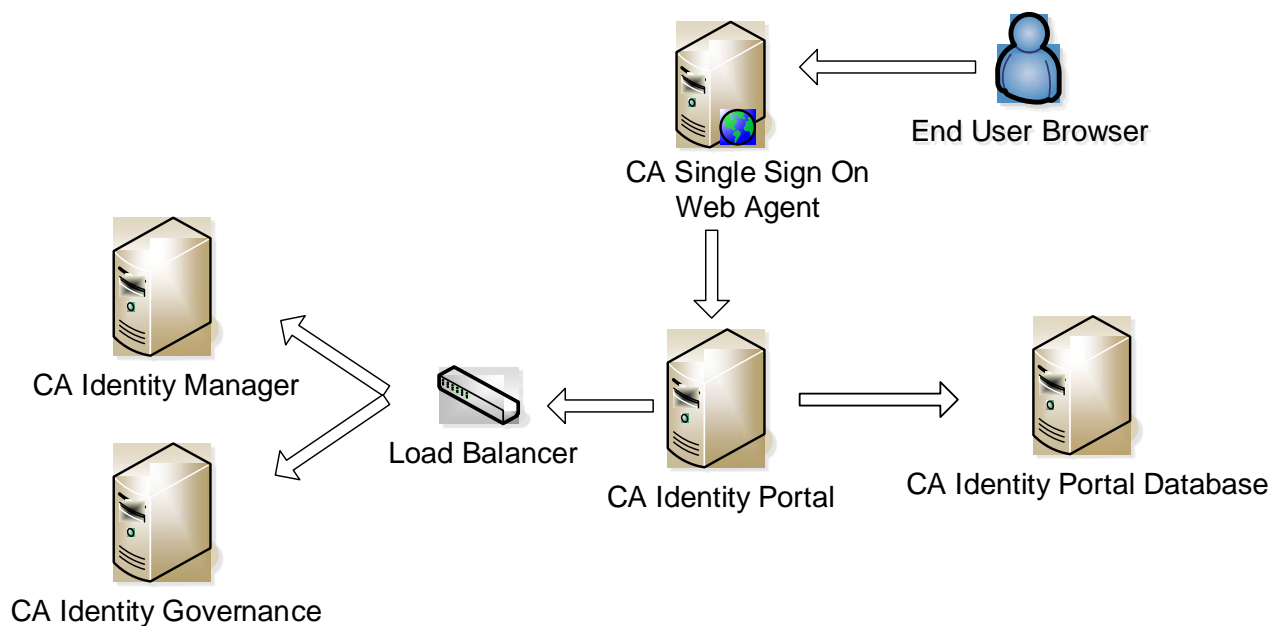
CA Identity Portal is a web-based business-ready identity and access management application. CA Identity Portal provides a business logic layer that leverages and aggregates functionality from existing Identity Management products, such as CA Identity Manager (CA IDM) and CA Identity Governance (CA GM). CA Identity Portal is designed for the non-technical business end user and delivers an intuitive all-inclusive interface in the form of a single page web application.

From a components perspective, CA Identity Portal is a java web application that is deployed on a supported application server or servlet container. CA Identity Portal requires a database for its configuration and persistence stores. CA Identity Portal interfaces with the organization's existing IDM platforms (such as CA Identity Manager) through CA Identity Portal backend connectors. CA Identity Portal communicates with the IDM backend platforms using the exposed public APIs of these backend systems (for example, Web Services (TEWS) & Workpoint APIs for CA IDM, and web services API for CA GM).

CA Identity Portal can be deployed in a single node configuration or in a multi node cluster configuration. A CA Identity Portal cluster configuration does not depend on the application server cluster abilities and can exist even if the application server itself is not deployed in a cluster mode.

CA Identity Portal can be deployed with basic authentication, where user credentials (user id, password) are validated to a main CA Identity Portal backend connector (for example CA IDM). Alternatively, CA Identity Portal can be integrated with Single Sign On to deliver an SSO experience to the end user.

## Components



## Software Requirements

This section describes the supported software prerequisites for installing CA Identity Portal.

### Operating Systems

OS	Version	Notes
Microsoft Windows Server	2008 R2 (SP1,SP2)	64 bit
Red Hat Enterprise Linux	6.x	64 bit

### Application Servers

The following are supported application servers on which CA Identity Portal can be deployed. These servers are supported on all the Operating Systems described above.

Application Server	Version	Java Version	Notes
Apache Tomcat	7.0.50	Sun JDK 1.6.x update 40 and above	On a Windows OS, install Apache Tomcat using the Apache Tomcat automatic installer (apache- tomcat-7.0.50.exe), which also registers Apache Tomcat as a Windows Service (Do not use the ZIP distribution).
Oracle WebLogic	11g R1 (10.3.5,10.3.6), 12c (12.1.1 Only)	Sun JDK 1.6.x update 40 and above.	Only Native Oracle WebLogic Cluster configuration is supported (Single node cluster or more).
Red Hat (RH) JBoss	EAP 6.1.0 GA (Application Server 7.2)	Sun JDK 1.6.x update 40 and above.	Only Standalone server is supported (RH) JBoss Native Cluster is not supported)



**Note:** General notes for supported application servers:

- Only 64 bit application servers are supported.
- Only 64bit Java **JDKs** are supported. JREs are not supported (CA Identity Portal includes runtime compile elements).

## Databases

CA Identity Portal supports the following databases used for its runtime and persistent stores.

Vendor	Version	Notes
Oracle	11g R2	RAC is supported.
MySQL	5.5.x	
MS SQL	2008 R2	XA Transactions must be configured on the Database (see Appendix A for more information)

## CA Identity Manager and CA Identity Governance Back-ends

Vendor	Version	Notes
CA Identity Manager	12.5 (SP6-SP15), 12.6 (SP1,SP2,SP3,SP4)	Supported IDM Application servers: (RH) JBoss, Oracle WebLogic, WebSphere
CA Identity Governance	12.5 (SP6 and above) 12.6 (SP0, SP1)	



**Note:** In case CA Identity Manager or CA Identity Governance are deployed in a cluster, a NLB (Network Load Balancer) VIP is required for CA Identity Portal to leverage all IDM/GM cluster nodes.

## CA Single Sign On Option

Vendor	Version	Notes
CA Single Sign On	r12.5 r6.0 SP6 CR9 r12.0 SP3 CR11 and above	If CA Identity Portal is integrated with CA Single Sign On, IDM TEWS must also be integrated with CA Single Sign On.

## Web Clients (Browsers)

Browser	Version	Notes
Internet Explorer	8,9,10,11,12	On Windows Desktop OS
Mozilla Firefox	3.6 and above	On Windows Desktop OS
Google Chrome	All versions	On Windows Desktop OS
Safari	6.1 and above	On Mac OS



**Note:** The recommended screen resolution is 1280x800 (pixels)

## Hardware Requirements

The following are recommended PRODUCTION hardware specifications for the CA Identity Portal application server nodes. For fault tolerance and performance considerations, CA Identity Portal needs to be deployed in at least a two node cluster (two distinct servers).

Component (per node)	Minimum	Recommended
<b>CPU</b>	Dual Core Intel (or compatible) 2.0 GHz Xeon or similar (64 bit)	Quad Core Intel (or compatible) 2.0 GHz Xeon or similar (64 bit)
<b>RAM</b>	16 GB	32 GB
<b>Local Storage</b>	160 GB	160 GB
<b>Database Storage</b>	1GB Initial Size	5 GB Initial Size
<b>Shared Storage (for uploaded files)</b>	50 GB	100 GB

## Network Requirements

The following table summarizes the Firewall/Communications requirements between CA Identity Portal and various solution components.

From	To	Port and Protocol	Notes
<b>Web Servers (SSO web agents)</b>	<b>CA Identity Portal application servers</b>	CA Identity Portal Application Server HTTP port	For example: 8080 for Apache Tomcat

CA Identity Portal Application Servers	CA Identity Portal Database	Database port	
CA Identity Portal Application Servers	CA Identity Manager Servers	ALL TCP Ports	HTTP and RMI Traffic
CA Identity Portal Application Servers	CA Identity Governance Servers	TCP/8080 (HTTP)	
Identity Manager Servers	CA Identity Portal Application Servers	CA Identity Portal Application Server HTTP port	



**Note:** If CA Identity Manager or CA Identity Governance are deployed in a cluster an NLB (Network Load Balancer) VIP is required for CA Identity Portal to leverage all IDM/GM cluster nodes. CA Identity Portal will be configured to point to the VIP (Virtual IP) representing the CA IDM, CA GM clusters. NLB VIP Characteristics are as follows:

- **Relay:** All TCP ports.
- **Load Balancing Scheme:** Round Robin (No IP-stickiness).
- **Health Monitor:** Basic HTTP on the IDM/GM application server HTTP port (for example 8080 on (RH) JBoss).

## DNS Requirements

The CA Identity Manager application servers FQDNs should be resolvable from all the CA Identity Portal Application server nodes. Resolution should be performed either via DNS or a local host's file override.

## Cluster Requirements

When CA Identity Portal is deployed in a cluster, CA Identity Portal nodes use Java Groups technology to communicate and replicate configuration and state. CA Identity Portal does this in order to enhance performance and simplify the process of committing/announcing a configuration change to all the nodes in the CA Identity Portal cluster.

This is not a mandatory requirement (although it is a recommended best practice). In case the requirement is not addressed in a given CA Identity Portal cluster deployment, please see the note at the end of this section for guidelines regarding running CA Identity Portal in such an environment.

Java Groups discovery relies on UDP Multicast. For UDP Multicast to be possible, the CA Identity Portal cluster nodes should reside on the same network switch. In case the CA Identity Portal nodes reside on different network switches, layer 2 Multicast spoofing must be enabled on these switches.



## Choosing a Multicast Address

By default, the CA Identity Portal installer uses the following multicast address:

**228.6.7.9**

You need to make sure this address is not currently being used. Use the testing procedure described in the next section to check this.

However, in case you are installing several CA Identity Portal environments on the same physical network (for example a Development and a QA environment), you should use a different multicast address for each installation/CA Identity Portal cluster.

For example:

The CA Identity Portal Development environment nodes will use the multicast address: **228.6.7.9**

The CA Identity Portal QA environment nodes will use the multicast address: **228.6.7.10**

Otherwise, you run the risk of CA Identity Portal nodes from the QA environment, joining the CA Identity Portal Development cluster and vice versa.

## Testing JGroups Multicast

1. Download the following "[jgroups-3.3.1.Final.jar](http://sourceforge.net/projects/javagroups/files/JGroups/3.3.1.Final/)" JAR file from the JGroups web site.
2. Copy the "jgroups-3.3.1.Final.jar" file to each of the CA Identity Portal application servers to a folder of your choice.
3. Open a command prompt to that folder.
4. Run the jgroups Receiver Test on the 1<sup>st</sup> node (Java Runtime needs to be installed):

```
java -cp jgroups-3.3.1.Final.jar org.jgroups.tests.McastReceiverTest -mcast_addr 228.6.7.9 -port 46656
```

5. Run the jgroups Sender Test on the 2<sup>nd</sup> node:

```
java -cp jgroups-3.3.1.Final.jar org.jgroups.tests.McastSenderTest -mcast_addr 228.6.7.9 -port 46656
```

6. Type something in the sender console and press enter.  
The message should display in the receiver console on the other node.
7. Switch receiver and sender sides and try again to validate both directions work.

## What to Do If Multicast Is Not Available

When the CA Identity Portal cluster requirements, detailed in the section above, cannot be met, the following should be taken into consideration:

1. After performing configuration changes via the CA Identity Portal Admin UI, you must then connect to the CA Identity Portal Admin UI on each node in the cluster and flush (Clear) all the CA Identity Portal caches (using the Tools/Cache section in the Admin UI).

2. Cache based optimization in CA Identity Portal will be available on a per node basis. For example, if a certain user search has been performed on a specific node in the cluster, the result set will be cached only on the node (and not replicated to the other cluster nodes).



**Note:** In some cases, IPv6 addresses might interfere with Java Groups operations. If you experience issues with the CA Identity Portal cluster, add the following parameter to the server startup for each CA Identity Portal node:

`Djava.net.preferIPv4Stack=true`

---

# Installing CA Identity Portal on an Application Server

---

## Installation Overview

- 1) Install and prepare a Database
- 2) Install a JDK
- 3) Install and prepare the Application Server
- 4) Add 3<sup>rd</sup> Part Jars to the Application Server
- 5) Install CA Identity Portal (using the Installer or using a Manual procedure)
- 6) Perform Post Installation Steps

## Installation Prerequisites

### 1 - Install and Prepare a Database

#### Oracle Database

- a) Install a supported version of the Oracle database. It is recommended that the database will run on a separate server than the CA Identity Portal application server.
- b) Create a dedicated schema for CA Identity Portal. The schema user should have the following DB Roles:
  - CONNECT
  - RESOURCE
- c) Record the database user and password to be supplied to the CA Identity Portal installer.



**Note:** The database should be configured to support XA transactions. Consult your Database documentation on how to configure this.

---

#### MySQL Database

- a) Install a supported version of the MySQL database.
- b) It is recommended that the database will run on a separate server than the CA Identity Portal application server.
- c) Create a dedicated database instance for CA Identity Portal. Create a user with all privileges on the CA Identity Portal schema and grant remote access to this user.

- d) Record the database user and password to be supplied to the CA Identity Portal installer.

### MS SQL Database

- a) Install a supported version of the MS SQL database (It is recommended that the database will run on a separate server than the CA Identity Portal application server).
- b) MS SQL Server Security should be set to: *"SQL Server and Windows Authentication mode"*.
- c) Create a dedicated database instance for CA Identity Portal.
- d) Create a Login User (SQL Server authentication) with DBO privilege on the CA Identity Portal Database schema (record the user name and password).
- e) Configure TCP/IP connectivity for the SQL Server (record the TCP port to which the server is bound).
- f) Enable XA Transactions for the MS SQL Server (see "Appendix A" for more information).

## 2 - Install a JDK

Install a supported **Java Development Kit (JDK)**.



**Note:** If installing on Linux, make sure the JDK bin folder is the user's environment path. To verify this, type "java -version" on the command prompt and make sure the JDK runtime is indeed invoked.

---

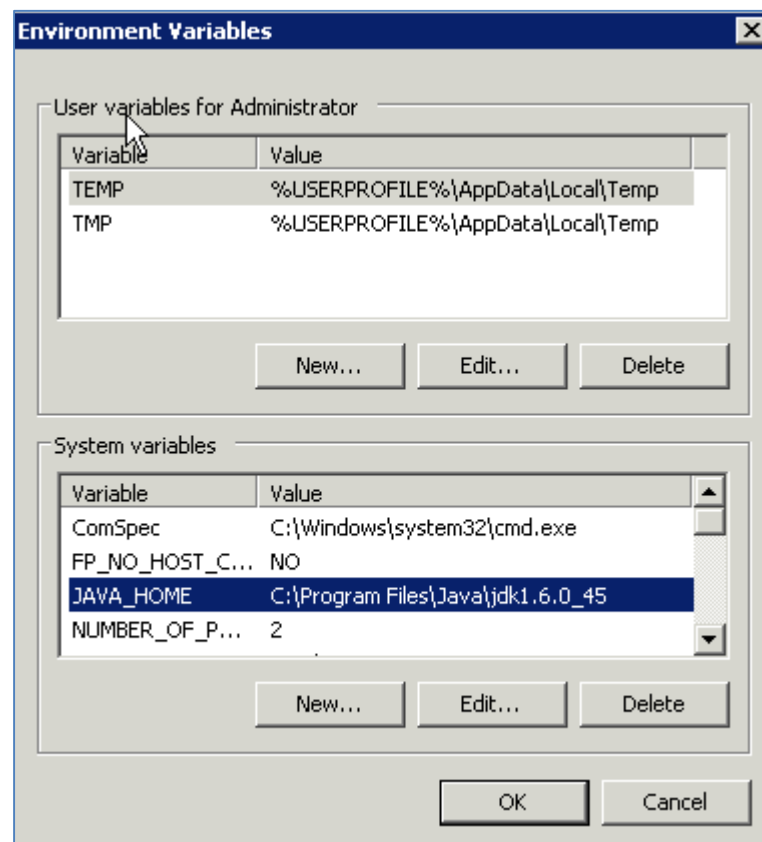
## 3 - Install and Prepare the Application Server

### (RH) JBoss Application Server

- a) Install a supported application server.
- b) Make sure the application server is configured to run with the **JDK** you installed (not the JRE).

One way to make sure (RH) JBoss starts with a JDK is to set a system wide environment variable: JAVA\_HOME and point it to the JDK home.

For example (on Windows):



Alternatively, you can modify the JBoss standalone.bat/.sh script and declare the JAVA\_HOME variable at the beginning of the file:

For example (on Windows):

```
set JAVA_HOME=c:\Java\jdk1.6.0_45
```

or on Linux:

```
export JAVA_HOME=c:\Java\jdk1.6.0_45
```

- c) Verify the application server starts correctly (using the standalone server start script). By default, (RH) JBoss only binds to the loopback interface (127.0.0.1). To bind (RH) JBoss to all interfaces (so it can be accessed from outside the hosting server) invoke the standalone startup scrip with the following parameters:

```
-b=0.0.0.0
```

```
-bmanagement=0.0.0.0
```

For example:

```
Standalone.bat -b=0.0.0.0 -bmanagement=0.0.0.0
```

- d) Verify (RH) JBoss started with the JDK. (RH) JBoss log should show a line pointing to your JDK. For example:

```
java.home = c:\Program Files\Java\jdk1.6.0_45\jre
```

- e) Record the application server base directory to be supplied to the CA Identity Portal installer.



**Note:** The CA Identity Portal installer supports installation on (RH) JBoss in "Standalone" server mode. To achieve a CA Identity Portal cluster configuration, you will need to run the CA Identity Portal installer on separate (RH) JBoss deployments in "Standalone" mode.

---

## Oracle WebLogic Application Server

- a) Install a supported application server.
- b) Make sure the application server is configured to run with the **JDK** you installed (not the JRE).



**Note:** The CA Identity Portal installer requires Oracle WebLogic to be configured in a cluster configuration with **at least** one admin server and one managed server. In addition Oracle WebLogic Node Manager must be configured and used to start and stop the managed server/s on which CA Identity Portal is to be installed.

---

- c) Verify the application server starts correctly when started using Oracle WebLogic Node Manager (by starting the managed server from the Oracle WebLogic Administration Console).
- d) Record the application server base directory to be supplied to the CA Identity Portal installer.

## Apache Tomcat Server

- a) Install a supported application server.

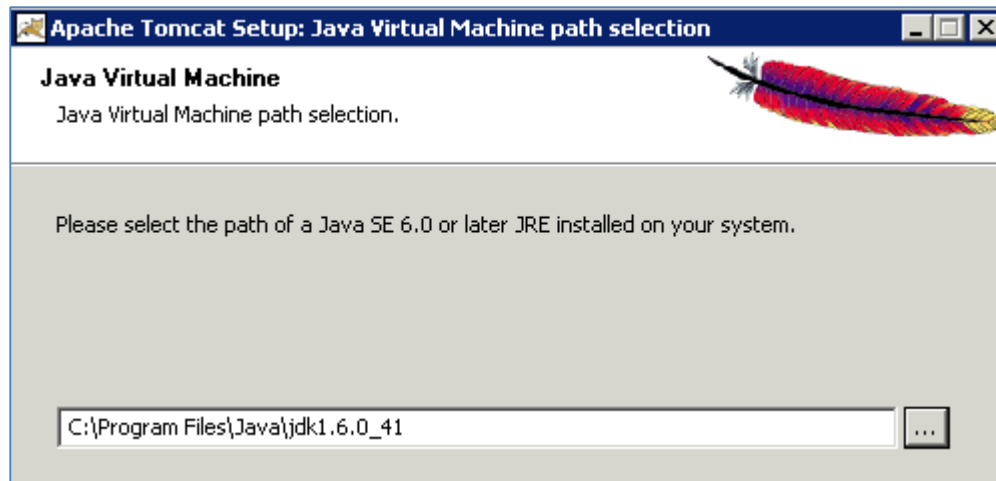


**Note:** On a Windows OS, install Apache Tomcat using the Apache Tomcat automatic installer (apache-tomcat-7.0.50.exe), which also registers Apache Tomcat as a Windows Service (Do not use the ZIP distribution).

---

- b) Make sure the application server is configured to run with the **JDK** you installed (not the JRE).

Example: When installing Apache Tomcat, during the installation process you are required to supply the path to a Java SE 6.0 JRE. At this point you should point the installer to the Java JDK home directory rather than to the JRE home directory.



- c) Verify the application server starts correctly.
- d) Record the application server base directory to be supplied to the CA Identity Portal installer.

## 4 – Add 3rd Party Jars to the Application Server

### Download the Database JDBC Driver for Your Database

You need to download a JDBC driver specific to your database vendor and place this driver in the application server lib folder. Follow the instructions below to accomplish this for your database and application server types.

#### MySQL Database (version 5.1x)

CA Identity Portal requires driver version 5.1.x and above (for example: mysql-connector-java-5.1.25-bin).

Download this driver from the following web site:

<http://dev.mysql.com/downloads/connector/j/>

#### Oracle Database

CA Identity Portal requires the latest Oracle JDBC driver for Java Runtime v.6 (ojdbc6.jar).

Download this driver distribution from the Oracle web site.

#### MS SQL Database (version 4)

CA Identity Portal requires MS SQL JDBC Driver v4.

Download this driver distribution from the Microsoft web site:

<http://www.microsoft.com/en-us/download/details.aspx?displaylang=en&id=11774>

## Apache Tomcat Server 3<sup>rd</sup> Party Jars Installation

### Installing Database JDBC Driver on Apache Tomcat

- a) Locate the JDBC driver jar in the JDBC driver distribution you downloaded, and place this jar in the Apache Tomcat "lib" folder.

For example: *C:\Tomcat 7.0\lib*

- b) Restart the Apache Tomcat application server.

### Installing BITRONIX (JTA) on Apache Tomcat

Apache Tomcat application server does not supply JTA transaction Manager functionality that is required by CA Identity Portal. When using Apache Tomcat, you need to download and install Bitronix, which is an open-source JTA transaction Manager. CA Identity Portal supports the Bitronix Transaction Manager (BTM) version 2.1.3.

- a) Download the BTM distribution from the following web site:

<http://bitronix.be/downloads/btm-dist-2.1.3.zip>

or

Browse to the Bitronix web site (<http://bitronix.be/>) and search for the 2.1.3 distribution version in the Download Archives section.

- b) Unzip the BTM distribution and copy to following jars to your Apache Tomcat **lib** folder

- i. <BTM Distro Root>\lib\slf4j-api-1.6.4.jar
- ii. <BTM Distro Root>\lib\slf4j-jdk14-1.6.4.jar
- iii. <BTM Distro Root>\lib\geronimo-jta\_1.1\_spec-1.1.1.jar
- iv. <BTM Distro Root>\integration\btm-tomcat55-lifecycle-2.1.3.jar
- v. <BTM Distro Root>\btm-2.1.3.jar

Example Tomcat destination directory: *C:\Tomcat 7.0\lib*

## JBoss Application Server 3<sup>rd</sup> Party Jars Installation

### Installing Database JDBC Driver on (RH) JBoss

For (RH) JBoss, CA Identity Portal requires that the database JDBC driver be deployed as a (RH) JBoss Module.

Deployment of the JDBC Driver as a module differs slightly between database vendors.

Follow the steps below to deploy the JDBC driver for your database vendor type



For more information, consult the JBoss Application Server documentation on how to deploy a JDBC driver on JBoss:

[https://community.jboss.org/wiki/DataSourceConfigurationInAS7#jive\\_content\\_id\\_Installing\\_a\\_JDBC\\_driver\\_as\\_a\\_module](https://community.jboss.org/wiki/DataSourceConfigurationInAS7#jive_content_id_Installing_a_JDBC_driver_as_a_module)

---





In case you want to use a MSSQL database with JBoss, you need to configure both MySQL and MSSQL JDBC drivers on JBoss (Follow the instructions in this section for both MySQL and MSSQL JDBC drivers).

## Oracle JDBC Driver

- a) Create the following directory tree inside the “<jboss-home>/modules” directory:

*oracle/jdbc/main/*

For example:

```
cd C:\jboss-eap-6.1\modules mkdir oracle\jdbc\main\
```

- b) Copy your JDBC driver jar (For Oracle this is: “ojdbc6.jar”) to the “<jboss-home>/modules/oracle/jdbc/main” directory.
- c) Using a text editor, create a file called “module.xml” inside “<jboss-home>/modules/oracle/jdbc/main” as following:

```
<?xml version="1.0" encoding="UTF-8"?>
<module xmlns="urn:jboss:module:1.0" name="oracle.jdbc">
  <resources>
    <resource-root path="ojdbc6.jar"/>
  </resources>
  <dependencies>
    <module name="javax.api"/>
    <module name="javax.transaction.api"/>
  </dependencies>
</module>
```

- d) Now open your <jboss-home>/standalone/configuration/standalone.xml file and add the driver declaration tag referring to the new module as following (look for the default driver declaration tag already contained in the file <driver name="h2? module="com.h2database.h2?> and add your driver declaration before it):

```
<driver name="oracle" module="oracle.jdbc">
  <driver-class>oracle.jdbc.OracleDriver</driver-class>
  <xa-datasource-class>oracle.jdbc.xa.client.OracleXADataSource</xa-datasource-
class>
</driver>
```

- e) Restart the application server.

## MYSQL JDBC Driver

- 1) Create the following directory tree inside the “<jboss-home>/modules” directory:

*com/mysql/main/*

For example:

```
cd C:\jboss-eap-6.1\modules
mkdir com\mysql\main\
```

- 2) Copy your JDBC driver jar (For MSSQL this is: “mysql-connector-java-5.1.25.jar”) to the “<jboss-home>/modules/com/mysql/main” directory.
- 3) Using a text editor, create a file called “module.xml” inside “<jboss-home>/modules/com/mysql/main” as following:

```
<module xmlns="urn:jboss:module:1.0" name="com.mysql">
  <resources>
    <resource-root path="mysql-connector-java-5.1.25.jar"/>
  </resources>
  <dependencies>
    <module name="javax.api"/>
    <module name="javax.transaction.api"/>
  </dependencies>
</module>
```

- 4) Now open your “<jboss-home>/standalone/configuration/standalone.xml” file and add the driver declaration tag referring to the new module as following (look for the default driver declaration tag already contained in the file <driver name="h2? module="com.h2database.h2?> and add your driver declaration before it):

```
<driver name="mysql" module="com.mysql">
  <driver-class>com.mysql.jdbc.Driver</driver-class>
  <xa-datasource-class>com.mysql.jdbc.jdbc2.optional.MysqlXADataSource</xa-
datasource-class>
</driver>
```

5. Restart the application server.

## Oracle WebLogic Server 3<sup>rd</sup> Party Jars Installation

### Installing database JDBC driver on Oracle WebLogic Server

Oracle WebLogic comes with pre-loaded JDBC drivers for Oracle, MySQL and MS SQL databases.

There is no need to load 3<sup>rd</sup> party drivers to Oracle WebLogic.

## 5 – (JBoss only) Disable application server modules

For JBoss only, you need to disable the native JBoss "WELD" module.

1. Make a backup of the JBoss standalone.xml configuration file (for example: <jboss-home>\standalone\configuration\standalone.xml)
2. Open the standalone.xml for editing, look for and delete the following sections:
  - a. `<extension module="org.jboss.as.weld"/>`
  - b. `<subsystem xmlns="urn:jboss:domain:weld:1.0"/>`
3. Save the file.

## Installing Using the CA Identity Portal Installer

Use the following procedure to install CA Identity Portal on Windows or Linux.



**Note:** If installing on Linux, ensure that you run the CA Identity Portal installer using the same user that was used to install the Application Server, and that the JDK bin folder is in the user's environment path. To verify type "java -version" on the command prompt and make sure the JDK runtime is indeed invoked.

---

- 1: Verify the installation pre-requisites described in the previous chapter have been met.
- 2: For Apache Tomcat, stop the application server on which you intend to deploy CA Identity Portal (for Oracle WebLogic, (RH) JBoss keep the application server started and make sure you can access the application server admin console).
- 3: Run the CA Identity Portal installer (sigma.exe/sigma.bin) on the computer where the CA Identity Portal application server is installed.
- 4: Choose the installation mode: "Extract Only" or "Install All". For a Wizard based installation, choose "Install All". For a manual installation, choose "Extract Only" and skip to the next chapter in this guide.
- 5: Accept the license agreement.
- 6: Supply the path to the installed JDK home folder.  
For example: C:\Program Files\Java\jdk1.6.0\_41
- 7: Select the Application Server type (Apache Tomcat, Oracle WebLogic, (RH) JBoss) to which CA Identity Portal will be deployed.  
In case of Apache Tomcat, enter the Apache Tomcat Windows Service name (as viewed in the windows services mmc snap-in). For example: Tomcat7.
- 8: Choose and enter a UserID and Password to be used as the CA Identity Portal Administrator.



**Note:** This username and password pair will be created in the Application Server security realm. In case of (RH) JBoss, make sure you choose a complex password (at least nine characters long, including a capital letter, number and a non-alphanumeric character. For example: Pizass1!).

---

- 9: Choose a folder location for the CA Identity Portal client log files.  
For example: C:\SIGMA\Logs



**Note:** This folder will be created, in case it does not exist.

---

10: Select a database type to be used for the CA Identity Portal configuration and runtime store (MySQL, MS SQL, Oracle).



**Note:** In the case of JBOSS, the installer currently only supports MySQL and Oracle databases. If you need to use MSSQL with JBoss, please select "MySQL" as the database type (After the installer completes the installation, follow the instructions in Appendix C to configure JBoss for a MSSQL database). If not done already, before you continue with the installation, you should also prepare the MySQL JDBC driver on the JBoss server (even if you are not planning to use it). See 4 – Add 3rd Party Jars to the Application Server in the Installation Prerequisites section.

---

11: Supply database connection and credential information.

*Note: The installer does not verify connectivity and credentials to the database. Please make sure the connection details and credentials are valid.*

12: Choose to install the included CA Identity Portal release version or install a patched CA Identity Portal version.

13: Choose a CA Identity Portal home folder (where the tools and sample files will be installed).



**Note:** This folder will be created, in case it does not exist.

---

14: Review and approve the summary of installation. Click "Install" to perform the actual installation.

15: Validate installation results:

- a. If the application server has not been started by the installer, start the application server now.
- b. Review application server log file for startup errors.
- c. Check that the CA Identity Portal Administration UI is up.
  - i. Browse to: <http://<application server host>:port/sigma/admin>  
For example: <http://localhost:8080/sigma/admin>
  - ii. Provide the CA Identity Portal Administrator Username and Password you set during the installation.

Proceed to the "Post Installation" section in this document.



**Note:** For a CA Identity Portal Apache Tomcat or (RH) JBoss cluster installation, rerun this installation procedure on each CA Identity Portal application server node. In case of Oracle WebLogic, the Oracle WebLogic cluster takes care of deploying CA Identity Portal to the managed cluster nodes.

---

## Manual Installation

### Overview

To install CA Identity Portal manually, you must first have all the pre-requisites in place. See and review the pre-requisites section in this document for more information.

Once the pre-requisites are fulfilled, you will need to perform the following steps to install CA Identity Portal manually on the application server:

1. Extract the CA Identity Portal application and sample configuration files from the installer.
2. Generate a CA Identity Portal keystore file.
3. Set CA Identity Portal specific JVM options in the application server.
4. Configure a database data source in the application server.
5. Configure Bitronix Transaction Manager (for Apache Tomcat Only).
6. Deploy the CA Identity Portal Web Archive to the Application Server.
7. Create a user to access the CA Identity Portal Admin UI.
8. Verify the installation.

### Extract the CA Identity Portal Application and Sample Configuration Files

Run the CA Identity Portal installer in "Extract Only" mode. This will extract required files and samples to a location of your choice in the file system (This location will be referenced as the CA Identity Portal Home Folder).

Preferably, run the installer on the application server host machine (otherwise, you will need to copy sample configuration files from the extracted installer to the application server).

### Generate a CA Identity Portal Keystore File

CA Identity Portal requires a key to encrypt sensitive configuration data in the database. This key is stored in a CA Identity Portal specific Java Keystore file.

Use the following procedure to create a key and Keystore file for CA Identity Portal. In the next step you will configure JVM parameters that point to the Keystore file you created.

1. Open a command prompt and change directories to the CA Identity Portal "sigma-keystore-tool" folder located in the CA Identity Portal home folder (where you extracted the CA Identity Portal installer in the previous step).

For example:

```
<SIGMA_HOME>\SIGMA\sigma-keystore-tool\
```

- Run the following command to generate the CA Identity Portal Keystore:

```
java -jar CreateSigmaKeystore.jar <keystore_file_path> <keystore_password> <key_name>
<key_password>
```

For example:

```
java -jar CreateSigmaKeystore.jar mysigmaks.ks kssecret sigmakey keysecret
```

- Record the full path to the keystore you created and all the other parameters you provided. In the next section you will create CA Identity Portal specific JVM options that point the keystore and its parameters.



**Note:** In case you plan to install CA Identity Portal in a cluster, you will need to copy this Keystore file to all nodes in the cluster.

## Set Application Server Startup JVM Options

Add the following JVM options to the application server startup:

Mandatory JVM Option (For all Application Servers)	
Database Related	
<b>-Dsigma.persistence.xml.location=&lt;path to persistence xml&gt;</b>  Example Values: <i>file:d:\SIGMA\3rd-party-config-files\tomcat-persistence-files\sigma-persistence.xml</i> <i>file:d:\SIGMA\3rd-party-config-files\tomcat-persistence-files\sigma-persistence-oracle.xml</i>	Sample sigma persistence files are included with CA Identity Portal in the CA Identity Portal home folder under the "3rd-party-config-files" folder. Sample files are provided for all application server types and database types. For Oracle DB, use the "sigma-persistence-oracle.xml" file. For MySQL and MS SQL use the "sigma-persistence.xml" file.
<b>-Dhibernate.id.new_generator_mappings=true</b>	Set only in case Oracle is used as a Database for CA Identity Portal.
<b>-Dsigma.spring.jpa.context=file:d:\SIGMA\3rd-party-config-files\tomcat-persistence-files\sigma-jpa-context.xml</b>	Sample sigma jpa context files are included with CA Identity Portal in the CA Identity Portal home folder under the "3rd-party-config-files" folder. Sample files are provided for all application server types.

Mandatory JVM Option (For all Application Servers)	
Database Related	
Cluster Related	
<b>-Djgroups.mping.mcast_addr=228.6.7.9</b> <b>-Djgroups.mping.mcast_port=46656</b> <b>-Djgroups.bind_addr=&lt;server ipv4 address&gt;</b>	<p>Needed for CA Identity Portal cluster communications. See Cluster Requirements section for more information.</p> <p>Substitute your server IP address. Each CA Identity Portal node is a cluster must be set with its own IP address.</p>
<b>-</b> <b>Dsigma.infinispan.configuration.location=file:D:\SIGMA\conf\infinispan-config.xml</b>	<p>Path to the CA Identity Portal caching properties file.</p> <p>Use this if you want to override the default caching properties. Sample infinispan config files are included with CA Identity Portal in the CA Identity Portal home folder under the "3rd-party-config-files" folder.</p> <p>Sample files are provided for all application server types.</p>
Log Related	
<b>-Dlog4j.configuration=file:D:\SIGMA\conf\log4j.properties</b>	<p>Path to the CA Identity Portal log properties file.</p> <p>Use this if you want to override the default log properties. A sample log4j.properties file is provided in the CA Identity Portal home folder in "sigma-docs\log4j.properties.sample". The file system path is an example.</p>
<b>-Dlog4j.logpath=&lt;path to log directory&gt;</b> <b>-Dorg.apache.cxf.Logger=org.apache.cxf.common.logging.Log4jLogger</b>	<p>Designate an operating system path (for example: d:\sigma\logs) to store CA Identity Portal related logs.</p>
Memory Related	

Mandatory JVM Option (For all Application Servers)	
Database Related	
<b>-XX:MaxPermSize=512m</b>	Needed for runtime compilation of TEWS classes.
<b>-Xms4g</b> <b>-Xmx8g</b>	Minimum and Maximum JVM Heap Memory limits. These values should reflect the expected load and available RAM on the CA Identity Portal server. The example given is for a production server in a large enterprise organization with 32GB of RAM. Memory benchmarks should always be performed and memory prams fine-tuned on regular basis (for example, using JConsole).
Encryption Related (CA Identity Portal Keystore)	
<b>-Dsigma.encryption.keystoreLocation=D:\SIGMA\conf\sigma.keystore</b> <b>-Dsigma.encryption.keystorePassword=&lt;secret_password&gt;</b> <b>-Dsigma.encryption.keyPassword=&lt;secret_key_password&gt;</b> <b>-Dsigma.encryption.keyName=&lt;key_name&gt;</b>	Sets the file system path and passwords for the CA Identity Portal Keystore. The keystore file must already exist (See "Generating a CA Identity Portal Keystore file" in this guide).

JVM Option (for Apache Tomcat Only)	
<b>-Dbtm.root=&lt;Tomcat Home Path&gt;</b> <b>-Dbitronix.tm.configuration=&lt;Tomcat Home Path&gt;\conf\btm-config.properties</b>	Set when Bitronix is used with Tomcat. <i>Note: make sure you use short names on the Apache Tomcat Home Path (for example, use "Progra~1" instead of "Program Files")</i>





**Note:** How to set JVM options in different Application Server

#### Oracle WebLogic

- If you are using an Oracle WebLogic Node Manager to start the nodes, use the Oracle WebLogic Administration Console to add the above JVM options to each node "Server Start" section.
- In case you are not using the Oracle WebLogic Node Manager to start the nodes (you start them manually from scripts), add the above JVM options to Domain wide setDomainEnv.cmd/ setDomainEnv.sh startup script.

#### Apache Tomcat

On MS Windows, use the "Configure Tomcat" utility (Tomcat7w.exe). Update the startup script (for Linux based deployments), to add these JVM options.

#### Red Hat JBoss

Set the JVM options in the batch file you use to start Red Hat JBoss (for example: standalone.bat or standalone.sh). Alternatively, you can use the Red Hat JBoss Admin Console.

## Configure a Data Source in the Application Server

This step is relevant only for Oracle WebLogic and (RH) JBoss (for Apache Tomcat, skip to the next step "INSTALLING BITRONIX (JTA) ON TOMCAT").

Configure a database **XA Data Source** in the Application server to connect to the CA Identity Portal database. Make sure to select the correct Database type/driver.

For **WebLogic**, set the following Data source JNDI Name: **jdbc/sgmadb**

For **JBoss**, set the following Data source JNDI Name: **java:jboss/datasources/jdbc/sgmadb**

*Example 1: WebLogic using a MySQL Database*

The screenshot shows the Oracle WebLogic Administration Console interface. The 'Configuration' tab is selected, and the 'Identity Options' sub-tab is active. A 'Save' button is at the top left. Below it, a descriptive paragraph explains that applications get a database connection from a data source by looking up the data source on the JNDI tree. The main configuration area is titled 'Name: DS' with a description: 'A unique name that identifies this data source in the WebLogic domain. More Info...'. Below this, the 'JNDI Name:' field is set to 'jdbc/sgmadb' with a description: 'The JNDI path to where this data source is bound. By default, the JNDI name is the name of the data source. More Info...'. Further down, there are three settings: 'Row Prefetch Enabled' (unchecked), 'Row Prefetch Size' (set to 48), and 'Stream Chunk Size' (set to 256). Each setting has a description and a 'More Info...' link. A 'Save' button is at the bottom left.

The screenshot shows the JBoss Configuration console with the 'Connection Pool' tab selected. The 'URL' field is set to 'jdbc:mysql://10.0.0.70:3306/sigmacc' and the 'Driver Class Name' is 'com.mysql.jdbc.Driver'. The 'Properties' section contains 'user=sigma'. The 'System Properties' section is empty.

**Configuration** Targets Monitoring Control Security Notes

General **Connection Pool** Transaction Diagnostics Identity Options

Save

The connection pool within a JDBC data source contains a group of JDBC connections that applications res connections within it are created when the connection pool is registered, usually when starting up WebLoq

Use this page to define the configuration for this data source's connection pool.

**URL:** jdbc:mysql://10.0.0.70:3306/sigmacc

**Driver Class Name:** com.mysql.jdbc.Driver

**Properties:**

user=sigma

**System Properties:**

*Example 2: (RH) JBoss and an Oracle database:*

Use the JBoss admin console to configure a new XA Data Source for CA Identity Portal.

The screenshot shows the JBoss Enterprise Application Platform 6.1.0.GA admin console. The 'Datasources' tab is selected, and the 'JDBC XA Datasources' section is active. The 'Available XA Datasources' table shows a single entry 'sigma-ds' with the JNDI name 'java:jboss/datasources/jdbc/sigmadb' and is enabled. The 'Selection' section shows the 'Attributes' tab, and the 'Edit' button is visible. The configuration details for 'sigma-ds' are shown below, including the Name, JNDI, XA Data Source Class, Driver, and other properties.

**JBoss Enterprise Application Platform 6.1.0.GA** (1) Messages Profile Runtime

Subsystems

- Connector
  - JCA
  - Datasources**
  - Resource Adapters
  - Mail
- Container
- Core
- Security
- Web

General Configuration

- Interfaces
- Socket Binding
- Paths
- System Properties

**Datasources** XA Datasources

**JDBC XA Datasources**

JDBC XA datasource configuration.

**Available XA Datasources**

Name	JNDI	Enabled?
sigma-ds	java:jboss/datasources/jdbc/sigmadb	✓

Add Remove Disable

Selection

Attributes Connection Security Properties Pool Validation

Edit

**Name:** sigma-ds **JNDI:** java:jboss/datasources/jdbc/sigmadb

**XA Data Source Class:** **Is enabled?:** true

**Driver:** oracle **Share Prepared Statements:** false

**Statement Cache Size:** 0

[Need Help?](#)

*Example 3: (RH) JBoss and a MSSQL database:*

Use the JBoss admin console to configure a new XA Data Source for CA Identity Portal.

The screenshot shows the JBoss Enterprise Application Platform 6.1.0.GA Admin Console. The left sidebar shows the navigation tree with 'Datasources' selected under 'Subsystems'. The main panel is titled 'JDBC XA Datasources' and shows the configuration for the 'SIGMA' datasource. The 'Available XA Datasources' table lists the 'SIGMA' datasource with JNDI 'java:jboss/datasources/jdbc/sigmadb' and 'Enabled?' checked. Below the table, the 'Selection' tabs include 'Attributes', 'Connection', 'Security', 'Properties', 'Pool', and 'Validation'. The 'Attributes' tab is active, showing the 'Name' as 'SIGMA' and 'JNDI' as 'java:jboss/datasources/jdbc/sigmadb'. Other attributes include 'XA Data Source Class: com.microsoft.sqlserver.jdbc.SQLServer', 'Is enabled?: true', 'Driver: sqlserver', 'Share Prepared Statements: false', and 'Statement Cache Size: 0'.

Name	JNDI	Enabled?
SIGMA	java:jboss/datasources/jdbc/sigmadb	✓

Key	Value	Option
DatabaseName	sigma	Remove
SelectMethod	cursor	Remove
ServerName	localhost	Remove

Make sure to set these three properties on the datasource: DatabaseName, SelectMethod, ServerName

The screenshot shows the JBoss Enterprise Application Platform 6.1.0.GA Admin Console, specifically the 'Properties' tab for the 'SIGMA' datasource. The 'Available XA Datasources' table is the same as in the previous screenshot. The 'Selection' tabs include 'Attributes', 'Connection', 'Security', 'Properties', 'Pool', and 'Validation'. The 'Properties' tab is active, showing a table with three properties: 'DatabaseName' (value: sigma), 'SelectMethod' (value: cursor), and 'ServerName' (value: localhost). Each property has a 'Remove' button next to it.

Key	Value	Option
DatabaseName	sigma	Remove
SelectMethod	cursor	Remove
ServerName	localhost	Remove

## Installing BITRONIX (JTA) on Apache Tomcat Only

Apache Tomcat application server does not supply JTA transaction Manager functionality that are necessary for CA Identity Portal. On Apache Tomcat you must use Bitronix (BTM), which is an open-source JTA transaction Manager. CA Identity Portal supports the Bitronix version 2.1.3.

1) Download the BTM distribution from:

<http://bitronix.be/downloads/btm-dist-2.1.3.zip>

2) Unzip the BTM distribution and copy to following jars to your Apache Tomcat lib folder.

- i. <BTM Distro Root>\lib\slf4j-api-1.6.4.jar
- ii. <BTM Distro Root>\lib\slf4j-jdk14-1.6.4.jar
- iii. <BTM Distro Root>\lib\geronimo-jta\_1.1\_spec-1.1.1.jar
- iv. <BTM Distro Root>\integration\btm-tomcat55-lifecycle-2.1.3.jar
- v. <BTM Distro Root>\btm-2.1.3.jar

Create a file in the Apache Tomcat configuration folder called "*resources.properties*" with the following configuration (this is an example for a MySQL database configuration). A sample file is included in the CA Identity Portal Home Folder under "*\SIGMA\3rd-party-config-files\bitronix\resources.properties*".

```
resource.ds.className=com.mysql.jdbc.jdbc2.optional.MysqlXADataSource
resource.ds.uniqueName=jdbc/sigmadb
resource.ds.minPoolSize=10
resource.ds.maxPoolSize=10
resource.ds.driverProperties.URL=jdbc:mysql://10.0.0.70/sigmadb
resource.ds.driverProperties.user=sigma
resource.ds.driverProperties.password=xxxxxx
resource.ds.allowLocalTransactions=true
resource.ds.driverProperties.pinGlobalTxToPhysicalConnection=true
```



**Note:** Substitute the database type and connectivity information in the above example with parameters relevant to your database.

Create a file in tomcat configuration folder called "*btm-config.properties*" with the following configuration:

```
bitronix.tm.serverId=tomcat-btm-node0
bitronix.tm.journal.disk.logPart1Filename=${btm.root}/work/btm1.tlog
bitronix.tm.journal.disk.logPart2Filename=${btm.root}/work/btm2.tlog
bitronix.tm.resource.configuration=${btm.root}/conf/resources.properties
```

Edit "*server.xml*" file in Apache Tomcat conf directory, under the line:

```
<Listener className="org.apache.catalina.mbeans.GlobalResourcesLifecycleListener"/>
```

Add this line:

```
<Listener className="bitronix.tm.integration.tomcat55.BTMLifecycleListener"/>
```

Edit "Context.xml" file in Apache Tomcat conf directory, under the line:

```
<WatchedResource>WEB-INF/web.xml</WatchedResource>
```

Add this line:

```
<Transaction factory="bitronix.tm.BitronixUserTransactionObjectFactory" />
<Resource name="jdbc/sigmadb" auth="Container" type="javax.sql.DataSource"
factory="bitronix.tm.resource.ResourceObjectFactory" uniqueName="jdbc/sigmadb"/>
<Resource name="TransactionSynchronizationRegistry" auth="Container"
type="javax.transaction.TransactionSynchronizationRegistry"
factory="bitronix.tm.BitronixTransactionSynchronizationRegistryObjectFactory"/>
```

## Deploy the CA Identity Portal Application Web Archive to the Application Server

Locate the CA Identity Portal web archive file "**sigma.war**" in the CA Identity Portal Home Folder.

Deploy the CA Identity Portal web archive to your application server.

### For Apache Tomcat

- 1) Stop the application server.
- 2) Copy the sigma war file to the tomcat "webapps" folder (<tomcat home\webapps):

For example:

C:\tomcat7\webapps\

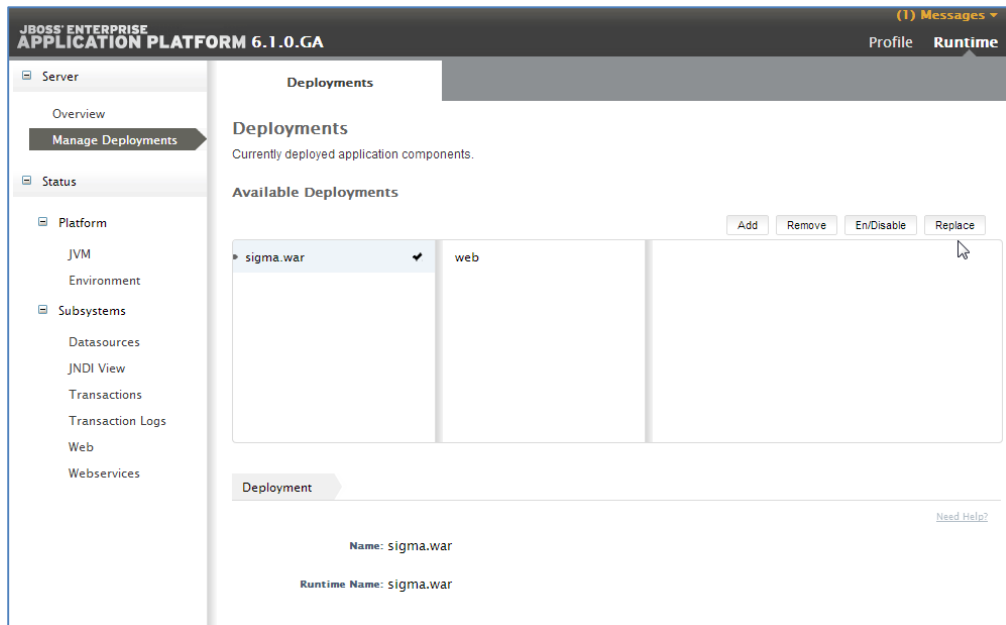
### For Oracle WebLogic

Use the Oracle WebLogic Administration Console to deploy the sigma war as a web application.

In case you have an Oracle WebLogic cluster, make sure to deploy the application to the Cluster.

### For Red Hat JBoss

Use the JBoss Admin Console to deploy the CA Identity Portal Application Web Archive to the server runtime.



## Create a User to Access the CA Identity Portal Administration UI

Create an application server user and group for accessing the CA Identity Portal Administration UI.

To access the CA Identity Portal Admin UI you need to create the appropriate security group and users in the application server.

### For Apache Tomcat

Add the following to "tomcat-users.xml"

```
<role rolename="SigmaAdministrators"/>
<user username="sigma" password="secret123" roles="SigmaAdministrators"/>
```

### For Oracle WebLogic

- 1) Using the Oracle WebLogic Admin Console, create an Oracle WebLogic security group called "SigmaAdmins".
- 2) Create a user called "sigma" or any other user and place that user in the "SigmaAdmins" group.

The screenshot shows the 'Groups' configuration page in the CA Identity Portal. The left sidebar contains the 'Domain Structure' tree with 'Security Realms' highlighted. The main content area is titled 'Settings for myrealm' and shows a table of groups. The 'SigmaAdmins' group is highlighted with a red circle.

**Domain Structure:**

- IDMDevDomain
  - Environment
  - Deployments
  - Services
  - Security Realms**
  - Interoperability
  - Diagnostics

**How do I...:**

- Manage users and groups
- Create groups
- Modify groups
- Delete groups

**System Status:**

Health of Running Servers

Name	Description	Provider
AdminChannelUsers	AdminChannelUsers can access the admin channel.	DefaultAuthenticator
Administrators	Administrators can view and modify all resource attributes and start and stop servers.	DefaultAuthenticator
AppTesters	AppTesters group.	DefaultAuthenticator
CrossDomainConnectors	CrossDomainConnectors can make inter-domain calls from foreign domains.	DefaultAuthenticator
Deployers	Deployers can view all resource attributes and deploy applications.	DefaultAuthenticator
Monitors	Monitors can view and modify all resource attributes and perform operations not restricted by roles.	DefaultAuthenticator
Operators	Operators can view and modify all resource attributes and perform server lifecycle operations.	DefaultAuthenticator
OracleSystemGroup	Oracle application software system group.	DefaultAuthenticator
<b>SigmaAdmins</b>	Sigma Administrators	DefaultAuthenticator

The screenshot shows the 'Users' configuration page in the CA Identity Portal. The left sidebar contains the 'Domain Structure' tree with 'Security Realms' highlighted. The main content area is titled 'Settings for myrealm' and shows a table of users. The 'sigma' user is highlighted with a red circle.

**Domain Structure:**

- IDMDevDomain
  - Environment
  - Deployments
  - Services
  - Security Realms**
  - Interoperability
  - Diagnostics

**How do I...:**

- Manage users and groups
- Create users
- Modify users

**Settings for myrealm:**

This page displays information about each user that has been configured in this security realm.

**Customize this table**

Name	Description	Provider
OracleSystemUser	Oracle application software system user.	DefaultAuthenticator
<b>sigma</b>	Sigma Administrator	DefaultAuthenticator
weblogic	This user is the default administrator.	DefaultAuthenticator

The screenshot shows the 'Groups' configuration page for the 'sigma' user. The left sidebar contains the 'Domain Structure' tree with 'Security Realms' highlighted. The main content area is titled 'Settings for sigma' and shows a list of parent groups. The 'SigmaAdmins' group is highlighted with a red circle.

**Domain Structure:**

- IDMDevDomain
  - Environment
  - Deployments
  - Services
  - Security Realms**
  - Interoperability
  - Diagnostics

**How do I...:**

Use this page to configure group membership for this user.

**Parent Groups:**

Available:

- AdminChannelUsers
- Administrators
- AppTesters
- CrossDomainConnectors
- Deployers
- Monitors
- Operators
- OracleSystemGroup

Chosen:

- SigmaAdmins**

## For (RH) JBoss

Use the JBoss `add_user.bat/add_user.sh` script supplied with JBoss (under the `<JBoss_Home>/bin` folder). Follow these guidelines (sample run of the `add_user` utility):

```
What type of user do you wish to add?
  a) Management User (mgmt-users.properties)
  b) Application User (application-users.properties)
(a): b
Enter the details of the new user to add.
Realm (ApplicationRealm) :
Username : sigma
Password : xxxxxxxx
Re-enter Password : xxxxxxxx
What roles do you want this user to belong to? (Please enter a comma separated
list, or leave blank for none)[ ]: SigmaAdministrators
About to add user 'sigma' for realm 'ApplicationRealm'
Is this correct yes/no? yes
Added user 'sigma' to file 'C:\jboss-eap-
6.1\standalone\configuration\application-users.properties'
Added user 'sigma' to file 'C:\jboss-eap-6.1\domain\configuration\application-
users.properties'
Added user 'sigma' with roles testers to file 'C:\jboss-eap-
6.1\standalone\configuration\application-roles.properties'
Added user 'sigma' with roles testers to file 'C:\jboss-eap-
6.1\domain\configuration\application-roles.properties'
Is this new user going to be used for one AS process to connect to another AS
process?
e.g. for a slave host controller connecting to the master or for a Remoting
connection for server to server EJB calls.
yes/no? yes
```

## Validate Installation Results

- 1) Start the application server.
- 2) Review CA Identity Portal log file for startup errors (CA Identity Portal logs are written to the application server stdout and stderr logs).
- 3) Check the CA Identity Portal Administration UI is up.
  - a) Browse to: `http://<application server host>:port/sigma/admin`  
For example: `http://localhost:8080/sigma/admin`
  - b) Provide the User ID and Password you defined for the CA Identity Portal Administrator.



**Note:** In case the userid and password for the Admin UI fail, reset the password for that application server user using the application server native tools.

---

- 4) Go to the Post Installation section in this document.



## Installing CA Identity Portal in a Cluster

CA Identity Portal can be deployed in a cluster. A CA Identity Portal cluster is essentially a collection of two or more application server nodes running the CA Identity Portal application and sharing a common CA Identity Portal store (database).

To deploy CA Identity Portal in a cluster, follow the installation steps outlined in either *Installing Using the CA Identity Portal Installer* or *Manual Installation*. To create the CA Identity Portal cluster, simply repeat the procedure on each application server node in the cluster.

After you complete the installation on all cluster nodes, follow these special Cluster Post Install steps:

### Copy the CA Identity Portal Keystore File from the 1<sup>st</sup> Node to All the Other Server Nodes

CA Identity Portal uses a symmetric encryption key to encrypt sensitive values in the configuration store. The encryption key is generated by the CA Identity Portal installer, or manually during the installation of CA Identity Portal. All nodes in the CA Identity Portal cluster must use the same key. Follow the following procedure to ensure all nodes use the same key.

- 1) Locate the sigma keystore file "sigma.keystore" on the first node on which you installed sigma.  
This is usually located under: "<SIGMA\_HOME>\SIGMA\sigma-keystore-tool\sigma.keystore".
- 2) Copy that file to all the other nodes, overwriting the files on those nodes (in that same location).
- 3) Restart the nodes.

# Post Installation

---

## CA Identity Manager Environment Validation

Use the IM Management Console and make sure you can export the IM Environment Role and Task Settings. Save this export as a backup of the environment before proceeding to the next step.

## Import CA Identity Portal Roles and Tasks into CA Identity Manager Environment

1. Locate the SIGMA-CORE-RoleDefinitions.xml file in the CA Identity Portal Home Folder (for example: C:\SIGMA\sigma-docs\SIGMA-CORE-RoleDefinitions.xml).
2. Connect to the CA Identity Manager Management Console.
3. Select the IME you want to integrate with CA Identity Portal.
4. Click Roles and Task Settings.
5. Click Import, Browse and select the SIGMA-CORE-RoleDefinitions.xml.
6. Restart the environment.



If the IDM environment you plan to integrate CA Identity Portal with does not have Provisioning configured (a provisioning directory is not configured for the environment), modify some of the CA identity Portal service tasks that you imported with the CA identity Portal Role Definitions XML. Follow the instructions in Appendix D – CA Identity Portal & IDM without Provisioning.

---

## Task Configuration in CA Identity Manager Environment

1. Using the CA Identity Manager Management Console under "Environment Advanced Settings - Web Services", Enable Execution for "Web Services". Restart the environment.  
See "Appendix B" in this document for a screen shot of how TEWS should be configured.
2. Using the User Console, modify the "View My Work List" admin task and enable it for web services in the task profile definition screen.

## Copy Workpoint Client JARS (from CA Identity Manager Application Server to CA Identity Portal)

Copy the following Jars from the CA Identity Manager Workpoint lib directory to a local directory on the CA Identity Portal application server (in case of a CA Identity Portal cluster, copy these jars to all the CA Identity Portal cluster nodes). This local folder directory will be referenced once, configuring a CA Identity Manager connector in the CA Identity Portal administration UI:

Post InstallationCopy Workpoint Client JARS (from CA Identity Manager Application Server to CA Identity Portal)

### For CA Identity Manager on (RH) JBoss Server

1. <JBoss IAM.im Deployment Folder>\library\wpClient.jar
2. <JBoss IAM.im Deployment Folder>\library\wpCommon.jar
3. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\jbossall-client.jar
4. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jboss-client.jar
5. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jboss-common-core.jar <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jboss-integration.jar
6. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jboss-javaee.jar
7. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jboss-logging-spi.jar
8. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jboss-remoting.jar
9. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jboss-security-spi.jar
10. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jboss-serialization.jar
11. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jbossx-client.jar
12. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\jnp-client.jar
13. <CA identity base installation dir>\IAM Suite\Identity Manager\tools\workpoint\lib\JBoss\policy.jar

### For CA Identity Manager on Oracle WebLogic Server

1. <Oracle WebLogic Server Home>\lib\wlclient.jar  
*For example: D:\Oracle\wlserver\server\lib\*
2. <WebLogic iam\_im.ear Deployment Folder>\APP-INF\lib\wpClient.jar
3. <WebLogic iam\_im.ear Deployment Folder>\APP-INF\lib\wpCommon.jar  
*For example: D:\Oracle\wlserver\server\lib\*

## For CA Identity Manager on IBM WebSphere Server

1. <WAS\_SERVER\_HOME>\runtimes\com.ibm.ws.ejb.thinclient\_7.0.0.jar
2. <WAS\_SERVER\_HOME>\runtimes\com.ibm.ws.orb\_7.0.0.jar
3. From the <WAS\_SERVER\_HOME>\WebSphere-ear\Identity Manager\WAS\_IMr12.ear file fetch the following jars:
  - a. Library\wpCommon.jar
  - b. Library\wpClient.jar

## Set Endorsed Libraries Override (JBoss Only)

If you are using (RH) JBoss, you need to configure an override for the Java Endorsed Directory, before you start the CA Identity Portal application server. Follow the procedure below:

### Extract CA Identity Portal Specific JAR Files from the CA Identity Portal Web Archive

Locate the CA Identity Portal web archive "sigma.war". It should be present in the CA Identity Portal home folder.

For example: C:\SIGMAQA\SIGMA\sigma.war

From the folder where the "**sigma.war**" is located Run the following commands to extract the CA Identity Portal specific JAR files:

```
"JAVA_HOME/bin/jar" xf sigma.war WEB-INF/lib/jaxb-api-2.2.6.jar
```

```
"JAVA_HOME/bin/jar" xf sigma.war WEB-INF/lib/geronimo-jaxws_2.2_spec-1.1.jar
```

For example (on windows):

```
"C:\Java\jdk1.6.0_45\bin\jar" xf sigma.war WEB-INF/lib/jaxb-api-2.2.6.jar
```

```
"C:\Java\jdk1.6.0_45\bin\jar" xf sigma.war WEB-INF/lib/geronimo-jaxws_2.2_spec-1.1.jar
```

These commands will extract the required JAR files to the following folder under the CA Identity Portal home folder:

<SIGMA\_HOME>\SIGMA\WEB-INF\lib\geronimo-jaxws\_2.2\_spec-1.1.jar

<SIGMA\_HOME>\SIGMA\WEB-INF\lib\jaxb-api-2.2.6.jar

### Copy CA Identity Portal Specific JARS to the JDK Endorsed Directory

1. Create an Endorsed Lib folder under your JDK lib folder:

For example:

Locate your JDK Lib folder:

c:\Program Files\Java\jdk1.6.0\_45\jre\lib\

Create a folder named "endorsed" under the \lib\ folder:

c:\Program Files\Java\jdk1.6.0\_45\jre\lib\endorsed\

2. Copy the two JARs you extracted in the previous step to the JDK endorsed folder. After you complete this step, the JDK endorsed folder should have the following content:

c:\Program Files\Java\jdk1.6.0\_45\jre\lib\endorsed\geronimo-jaxws\_2.2\_spec-1.1.jar

c:\Program Files\Java\jdk1.6.0\_45\jre\lib\endorsed\jaxb-api-2.2.6.jar

3. Verify that (RH) JBoss uses the JDK endorsed directory.

Look for the following line in the (RH) JBoss log file (the path should point to your JDK endorsed directory):

```
java.endorsed.dirs = c:\Program Files\Java\jdk1.6.0_45\jre\lib\endorsed
```

# CA Identity Portal and Single-Sign-On (SSO)

---

## Supported SSO Products

CA Identity Portal currently supports SSO with CA Single Sign On. See "Supported Single-Sign-On Options" in this document, for specific CA Single Sign On versions.

## Background

When CA Identity Portal is used by an end user, all actions performed on the user's behalf in CA Identity Manager (via TEWS) need to run in the user's security context in CA Identity Manager. In case CA Identity Portal is not protected by a web SSO solution (like CA Single Sign On), the end user supplies a user ID and password to CA Identity Portal. CA Identity Portal, in turn, supplies these credentials to TEWS<sup>1</sup>, thus running the CA Identity Manager tasks in the user context.

In case CA Identity Portal is protected by a web SSO solution, such as CA Single Sign On, the end user's password is unknown to CA Identity Portal. All that is known to CA Identity Portal is the user ID and hopefully the user's DN in the CA Identity Manager User directory. CA Identity Portal, without the user's password, now needs to invoke actions in CA Identity Manager on behalf of this user.

Several SSO scenarios can exist in a customer's environment, but in order for CA Identity Portal to support SSO the TEWS security framework MUST be configured for CA Single Sign On authentication, and without "Admin Password required" (see Option 1 in Table 1 below). In addition, CA Identity Portal must pass the appropriate CA Single Sign On HTTP headers with the TEWS SOAP call in order for TEWS to accept the user context.

---

<sup>1</sup> In this case the TEWS Security properties need to be set to: "Enable admin\_id (allow impersonation)" and "Admin Password is required" (See Screenshot 2 in Appendix B).

The following table lists the most common combinations of initial conditions (before CA Identity Portal deployment) and the effects of CA Identity Portal's Single Sign On deployment for existing TEWS clients.

**Table 1**

*CA Identity Portal/CA Identity Manager Single Sign On (SSO) Options (Before CA Identity Portal deployment)*

<i>Option</i>	<i>CA Identity Manager</i>	<i>TEWS</i>	<i>CA Identity Portal</i>	<i>Notes</i>
<b>1</b>	SSO Protected	SSO Protected (No Admin_password)	SSO Protected	This is the most desired/least complex initial conditions for a CA Identity Portal SSO deployment. Here the customer's TEWS setup is already configured for CA Identity Portal SSO support.
<b>2</b>	SSO Protected	Not used at all	SSO Protected	This is next desired/least complex initial conditions for a CA Identity Portal SSO deployment. Here TEWS is not used by anyone or any process (i.e. bulk load) and its configuration for CA Identity Portal will not affect other processes.
<b>3</b>	SSO Protected	Admin_Id & Password Protected	SSO Protected	In this case, existing customer TEWS clients (like the bulk loader client) will need to migrate from using Admin_ID and password to using SSO authentication. Either this or CA Identity Portal will not be SSO protected.
<b>4</b>	No SSO	Admin_Id & Password Protected	SSO Protected	In this case, existing customer TEWS clients (like the bulk loader client) will need to migrate from using Admin_ID and password to using SSO authentication. Either this or CA Identity Portal will not be SSO protected.
<b>5</b>	SSO Protected	SSO Protected (Admin_password is also used)	SSO Protected	The TEWS setup of the customer must be set to not use the Admin_password while using CA Single Sign On. This may affect existing TEWS clients. If this is not set CA Identity Portal cannot use SSO.

## SSO Prerequisites

### Enable Support for HTTP "delete" verb in CA Single Sign On

Out of the box, a CA Single Sign On deployment is not configured to allow the HTTP DELETE verb. CA Identity Portal requires that CA Single Sign On allows the DELETE verb for the CA Single Sign On Realm protecting CA Identity Portal. Follow this procedure to enable the DELETE verb.

- 1) Create a Delete Action for the CA Single Sign On Web Agent.  
By default, the WebAgent has only the Get, Post, and Put Actions available.  
To add the Delete Action, complete the following steps:
  - a) In the CA Single Sign On Administration Console, click **View** and select **Agent Types**.
  - b) Select Agent Types in the Systems pane.
  - c) Double-click Web Agent in the Agent Type list.
  - d) In the Agent Type Properties dialog box, click Create.
  - e) Enter Delete in the New Agent Action dialog box and click OK.
  - f) Click OK again to save the new Action.

The screenshot shows the 'Modify Agent Type: Web Agent' dialog box in the CA Single Sign On Administration Console. The 'General' tab is selected, showing the 'Name' field set to 'Web Agent'. The 'Agent Type Definition' section shows 'RADIUS Device' as a checkbox and 'IETF Vendor ID' as 2552. The 'Actions' section is expanded, showing a list of actions: Delete, Get, ImpersonateStart, ImpersonateStartUser, OnAccessAccept, OnAccessReject, OnAuthAccept, OnAuthAttempt, OnAuthChallenge, OnAuthReject, OnAuthUserNotFound, Post, ProcessSOAP, ProcessXML, and Put. The 'Delete' action is highlighted. A 'Create' button is at the bottom.



- 2) In the Rule configured to protect the CA Identity Portal realm make sure to select the **Delete, Get, Post, Put** verbs in the Agent Actions section:

The image shows two screenshots of the CA Identity Portal configuration interface.

**Top Screenshot: Modify Realm: SIGMA Application**

- General**
  - Name: SIGMA Application
  - Domain: SIGMA APP Domain
- Resource**
  - Agent: im\_sigma
  - Resource Filter: /sigma/
  - Effective Resource: im\_sigma/sigma/
  - Default Resource Protection: ☒ Protected ☐ Unprotected
  - Authentication Scheme: SIGMA Form Authentication
- Rules**
  - Table with columns: Name, Description
  - Row: SIGMA Allow all resources
  - Create button

**Bottom Screenshot: Modify Rule: SIGMA Allow all resources**

- Navigation: Infrastructure | Policies | Reports | Administration
- Breadcrumbs: Applications > Domains > Expressions > Global > Password
- General
  - Name: SIGMA Allow all resources
  - Domain: SIGMA APP Domain
- Attributes**
  - Realm and Resource**
    - Resource: \*
    - Effective Resource: im\_sigma/sigma/\*
    - Regular Expression: ☐
  - Allow/Deny and Enable/Disable**
    - ☒ Allow Access
    - ☐ Deny Access
    - Enabled ☒
  - Action**
    - Web Agent actions
    - Authentication events
    - Authorization events
    - Impersonation events
    - Action: **Delete**, Get, Post, ProcessSOAP

## CA Identity Portal Realm Protection

Protect CA Identity Portal in CA Single Sign On as you would any other web application.

The CA Identity Portal URI to protect is `"/sigma/"`.

However, CA Single Sign On needs to authenticate users for CA Identity Portal using the **same** authentication directory used to protect the CA Identity Manager realm (for the environment CA Identity Portal is integrating with).

For example, in case CA Single Sign On protects the CA Identity Manager realm with AD SSO and authorization directory mapping between AD and the IDM user store, protect the CA Identity Portal realm in exactly the same manner (using the same authentication directory and authorization mapping).

## Required CA Single Sign On Headers

CA Identity Portal relies on the out of the box CA Single Sign On authentication and authorization HTTP headers. In case your CA Single Sign On deployment was modified from its OOTB configuration, make sure the CA Identity Portal realm is configured to forward these specific SSO headers:

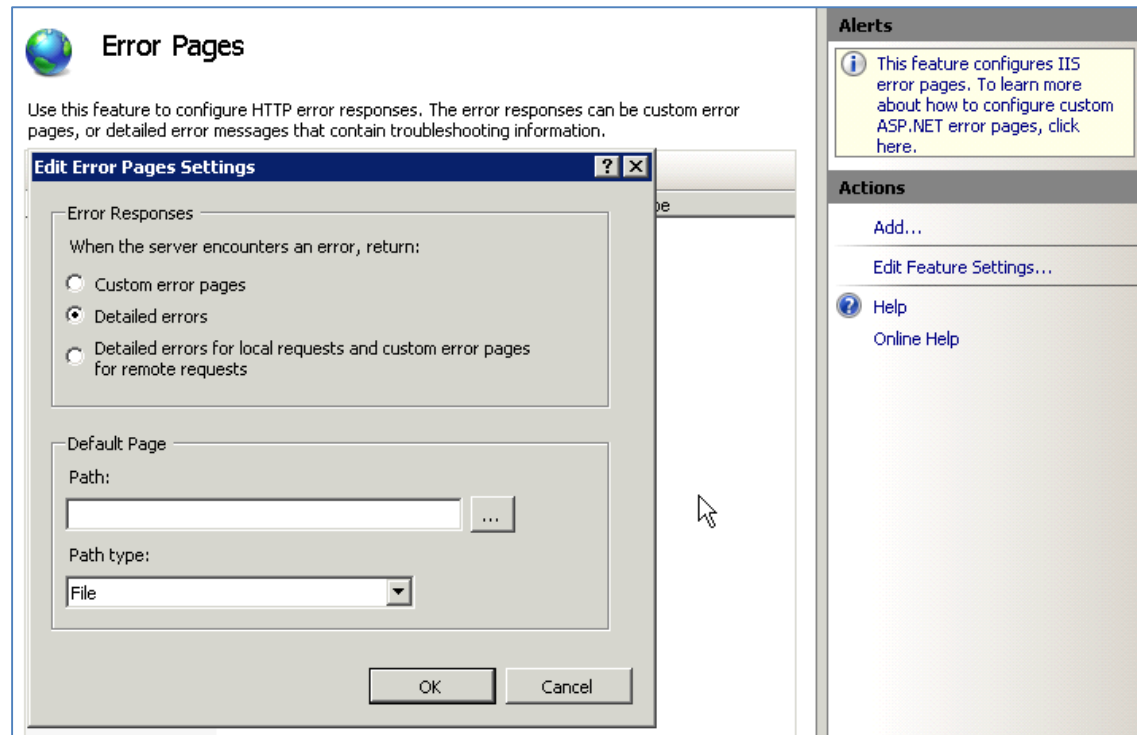
- `sm_user`
- `sm_userdn`
- `sm-authtype`
- `sm_serversessionspec`

## IIS Error Page Handling

In case the CA Single Sign On agent protecting CA Identity Portal is installed on Microsoft Internet Information Services (IIS), you need to enable "Detailed errors" in the IIS "Error Pages" settings.

The following instructions are for IIS v.7.5:

1. Select the Site protecting CA Identity Portal.
2. Open the "Error Pages" Feature.
3. Click on "Edit Feature Settings".
4. Select the "Detailed error" option and click OK.



## TEWS Security Settings

When CA Identity Portal is protected by CA Single Sign On, set exactly the following properties for TEWS in the CA Identity Manager environment serving *CA Identity Portal* (Either "Basic" or "Other" can be selected<sup>2</sup>).

*TEWS settings for CA Identity Portal with SSO*

Home > Environments > Sigma > Advanced Settings > Web Services

**Web Services Properties**

Property	
Enable Execution	<input checked="" type="checkbox"/>
Enable WSDL Generation	<input checked="" type="checkbox"/>
Enable admin_id (allow impersonation)	<input checked="" type="checkbox"/>
Admin password is required	<input type="checkbox"/>
SiteMinder Authentication	<input type="radio"/> (None) <input checked="" type="radio"/> Basic Authentication <input type="radio"/> Other
WSS Username Token (Password Text)	<input type="checkbox"/>
Generate WSDL in WS-I form (Note: your existing TEWS code may need to be modified).	<input type="checkbox"/>



CA Identity Portal does not support "TEWS configured with WSS Username Token" and "WS-I WSDL Format".

## CA Identity Governance, CA Identity Portal, and SSO

CA Identity Governance Web Services do not implement support for CA Single Sign On authentication (as of 12.5 SP7). CA Identity Governance Web services security implementation checks the WSS security header (UserNameToken) and authenticates (either to AD/LDAP or the Eurekify configuration). In order for Sigma to support CA Single Sign On with GM as an endpoint the following configuration is required for the CA Identity Governance Portal:

- 1) AD/LDAP authentication MUST be disabled in CA Identity Governance.
- 2) CA Single Sign On authentication needs to be enabled in CA Identity Governance (otherwise users will be able to access CA Identity Governance with an incorrect password).

<sup>2</sup> "Basic" means that CA Identity Manager will automatically configure the realm and protection in the SSO policy server when the environment is started.

"Other" means, the SSO admin will need to configure the protection of TEWS in SSO.

# Appendix A – Enabling XA Transactions for MS SQL

---

If you use a MS SQL database for the CA Identity Portal persistence store, enable XA transactions on that database server. Follow the procedure below (also see: [http://technet.microsoft.com/en-us/library/aa342335\(v=sql.105\).aspx](http://technet.microsoft.com/en-us/library/aa342335(v=sql.105).aspx) for more information).

## Configuring XA Transactions for SQL Server 2008 R2

### Enable the MS DTS Service on the Windows OS

The MS DTC service should be marked Automatic in Service Manager to make sure that it is running when the SQL Server service is started.

#### On Windows Server 2003:

- 1: Select Control Panel > Administrative Tools > Component Services.
- 2: Select Component Services > Computers and right-click My Computer, and select Properties.
- 3: Click the MSDTC tab, and then click Security Configuration.
- 4: Select the Enable XA Transactions check box, and then click OK.
- 5: Click OK again to close the Properties window, and then close Component Services.
- 6: Restart SQL Server to ensure that it syncs up with the MS DTC changes.

#### On Windows Server 2008:

- 1: Select Control Panel > Administrative Tools > Component Services.
- 2: Select Component Services > Computers > My Computer > Distributed Transaction Coordinator.
- 3: Right-click Local DTC and then select Properties.
- 4: Click the Security tab on the Local DTC Properties window.
- 5: Select the Enable XA Transactions check box, and click OK.
- 6: Click OK again to close the Properties window, and then close Component Services.
- 7: Restart SQL Server to ensure that it syncs up with the MS DTC changes.

## Configure the JDBC Distributed Transaction on the Database Server

In this section you will need files that come with the MS SQL JDBC v4 driver you downloaded before.

- 1: Locate the following files in the JDBC Driver package:
  - sqljdbc\_xa.dll
  - xa\_install.sql
- 2: Copy the **sqljdbc\_xa.dll** file from the JDBC package to the Binn directory (for a default SQL Server install, the location is C:/Program Files/Microsoft SQL Server/MSSQL10\_50.MSSQLSERVER/MSSQL/Binn) of SQL Server.



**Note:** If you are using XA transactions with a 32-bit SQL Server, use the sqljdbc\_xa.dll file in the x86 folder, even if the SQL Server is installed on an x64 processor. If you are using XA transactions with a 64-bit SQL Server on the x64 processor, use the sqljdbc\_xa.dll file in the x64 folder.

---

- 3: Run the **xa\_install.sql** database script on SQL Server. Run the SQL script as a SQL server administrator (sa) on the MASTER database.



**Note:** This script installs the extended stored procedures that are called by sqljdbc\_xa.dll. These extended stored procedures implement distributed transaction and XA support for the Microsoft SQL Server JDBC Driver. You can ignore errors about unable to drop procedures that don't exist.

---

## Grant the CA Identity Portal DB User the XA Role Permission

You need to grant the CA Identity Portal user login to run XA transaction on the database.

- 1: Open the SQL Server Management Studio to locate the security folder under the master database.
- 2: Grant permissions to the CA Identity Portal user to participate in distributed transactions with the JDBC driver:

Add the user to the **SqlJDBCXAUser** role in the master database (for example, for a sigmadb user add master database in User mappings and check SqlJDBCXAUser role).



**Note:** Failure to follow the procedure above to configure XA transactions on the MS SQL server can result in the following error in the CA Identity Portal server log: *"javax.transaction.xa.XAException: com.microsoft.sqlserver.jdbc.SQLServerException: Failed to create the XA control connection. Error: 'Could not find stored procedure 'master..xp\_sqljdbc\_xa\_init\_ex'.'"*


---

## Appendix B – TEWS Settings for CA Identity Portal without SSO

### TEWS settings for CA Identity Portal without SSO

[Home](#) > [Environments](#) > [Sigma](#) > [Advanced Settings](#) > Web Services

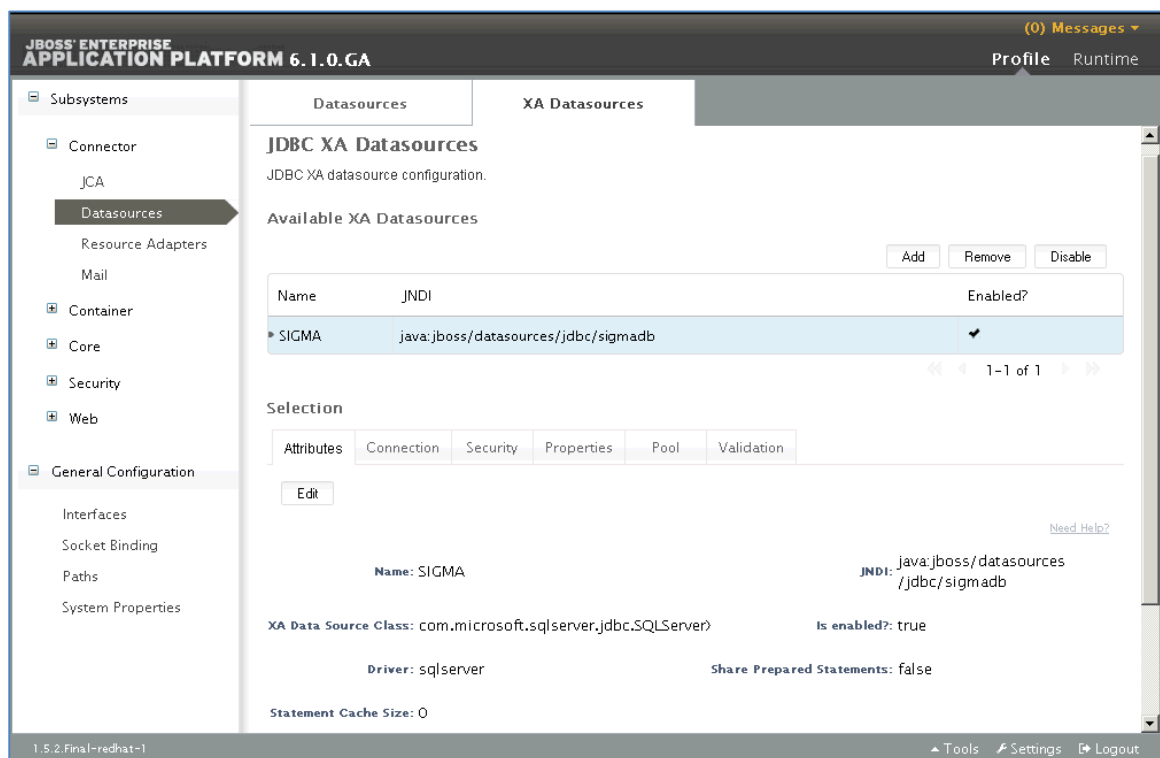
**Web Services Properties**

Property	
Enable Execution	<input checked="" type="checkbox"/>
Enable WSDL Generation	<input checked="" type="checkbox"/>
Enable admin_id (allow impersonation)	<input checked="" type="checkbox"/>
Admin password is required	<input type="checkbox"/>
 SiteMinder Authentication	<input checked="" type="radio"/> (None) <input type="radio"/> Basic Authentication <input type="radio"/> Other
WSS Username Token (Password Text)	<input type="checkbox"/>
Generate WSDL in WS-I form (Note: your existing TEWS code may need to be modified).	<input type="checkbox"/>

## Appendix C – Configuring JBoss with a MSSQL Datasource

When using CA Identity Portal with JBoss and a MSSQL database, select the MySQL database during the CA Identity Portal installation. After the CA Identity Portal installation is complete, follow the procedure below to reconfigure JBoss to use a MSSQL XA datasource.

1. Make sure the MSSQL JDBC driver is prepared on the JBoss server. See "Add 3rd Party Jars to the Application Server" in the "Installation Pre-requisites" section. Follow the instructions for configuring the MSSQL JDBC driver as a JBoss module.
2. Start the JBoss application server.
3. Open the JBoss administration console and go to the Profile->Datasources->XA Datasources Section.
4. Remove the MySQL XA Datasource definition.
5. Create a MSSQL XA datasource:
6. Use the JBoss admin console to configure a new XA Data Source for CA Identity Portal. Set the JNDI name to: ***java:jboss/datasources/jdbc/sigmadb***



Make sure to set these three properties on the datasource:

- DatabaseName
- SelectMethod
- ServerName



## Appendix C – Configuring JBoss with a MSSQL DatasourceConfiguring XA Transactions for SQL Server 2008 R2

The screenshot displays the JBoss Enterprise Application Platform 6.1.0.GA configuration console. The left sidebar shows the navigation tree with 'Subsystems' expanded, and 'Datasources' selected under the 'Connector' section. The main content area is titled 'JDBC XA Datasources' and shows the configuration for available XA Datasources. A table lists one available datasource named 'SIGMA' with the JNDI name 'java:jboss/datasources/jdbc/sigmadb' and is enabled. Below this, the 'Selection' tab is active, showing a table of properties for the selected datasource. The properties table includes 'DatabaseName' (sigma), 'SelectMethod' (cursor), and 'ServerName' (localhost), each with a 'Remove' option. The console also shows tabs for 'Attributes', 'Connection', 'Security', 'Properties', 'Pool', and 'Validation'. The bottom status bar indicates the version '1.5.2.Final-redhat-1' and provides links for 'Tools', 'Settings', and 'Logout'.

JBoss Enterprise Application Platform 6.1.0.GA

(0) Messages

Profile Runtime

Subsystems

Connector

JCA

Datasources

Resource Adapters

Mail

Container

Core

Security

Web

General Configuration

Interfaces

Socket Binding

Paths

System Properties

Datasources

XA Datasources

JDBC XA Datasources

JDBC XA datasource configuration.

Available XA Datasources

Add Remove Disable

Name	JNDI	Enabled?
SIGMA	java:jboss/datasources/jdbc/sigmadb	<input checked="" type="checkbox"/>

1-1 of 1

Selection

Attributes Connection Security Properties Pool Validation

Add

Need Help?

Key	Value	Option
DatabaseName	sigma	Remove
SelectMethod	cursor	Remove
ServerName	localhost	Remove

1-3 of 3

1.5.2.Final-redhat-1

Tools Settings Logout

## Appendix D – CA Identity Portal & IDM without Provisioning

In case the IDM environment you plan to integrate CA Identity Portal with does not have Provisioning configured (a provisioning directory is not configured for the environment), you will need to modify some of the CA Identity Portal service tasks that you imported with the CA Identity Portal Role Definitions XML.

1. Modify the "Sigma View User" Admin Task Tabs section and **remove** the "Provisioning Roles" and "Provisioning Roles Indirect" tabs.

**Modify Admin Task: Sigma - View User**

Profile Search **Tabs** Fields Events Role Use

Which tab controller should be used for this task?  
Standard Tab Controller

Which tabs should appear in this task?

Tab	Tag	Type	
Profile	Profile	Profile	
Access Roles	AccessRoles	Access Roles	
Admin Roles	AdminRoles	Admin Roles	
Provisioning Roles	ProvisioningRoles	Provisioning Roles	
Provisioning Roles Indirect	ProvisioningRolesIndirect	Provisioning Roles	
Groups	Groups	Groups	
Delegate Work Items	Delegation	Delegation	

2. Modify the "SIGMA Admin Scope" Admin Task Profile Screen.  
Select: Tabs->Profile->Browse Screen (SigmaAdminScope)->Edit

**Modify Admin Task: Sigma - Admin Scope**

Profile Search **Tabs** Fields Events Role Use

**Configure Profile**

**Select Screen Definition**

**Configure Standard Profile Screen**

• = Required

• Name

• Tag

Use  columns for layout. ➡

☐ |adminTasks|

☐ |provisioningRoles|

☐ |adminOfGroups|

3. Edit the "|provisioningRoles|" field.
4. Delete the "Initialization Javascript".

Initialization JavaScript

*This JavaScript must contain a function with the signature "function init(FieldContext)".*

```
function init(FieldContext) {
    var admin = FieldContext.getAdministrator();
    var provRoleAdmin = admin.getProvisioningRolesAdministrator();
    var uniques = new java.util.Vector();
    for (var i = 0; i < provRoleAdmin.size(); i++) {
        uniques.add(provRoleAdmin.get(i).getFriendlyName());
    }
    FieldContext.setMultiValue(uniques);
}
```

5. Apply and submit the changes.
6. Modify the "SIGMA – TEWS Tasks" Admin Role.  
Remove the existing Member Policy.

**Modify Admin Role: SIGMA - TEWS Tasks**

Profile Tasks **Members** Administrators Owners

*Members are able to use the tasks in a role.*

**Member Policies**

Member Rule	Scope Rules
	<b>Group</b>
	(all)
	<b>Provisioning Role</b>
	(all)
	<b>User</b>
	(all)



7. Create a new Member Policy with only Groups and User scoping.

**Modify Admin Role: SIGMA - TEWS Tasks**

**Profile** **Tasks** **Members** **Administrators** **Owners**

*Members are able to use the tasks in a role.*

**Member Policies**

	Member Rule	Scope Rules	
	(all)	<b>Group</b>	
		(all)	
		<b>User</b>	
		(all)	

8. Submit the changes.
9. Stop and start the CA Identity Portal application server.