CA Identity Manager

Installation Guide (JBoss)

r12.5 SP16

CA Technologies Product References

This document references the following CA products:

- CA IdentityMinder
- CA SiteMinder®
- CA Directory
- CA User Activity Reporting
- CA GovernanceMinder

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.

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

Chapter 1: Installation Overview

This guide provides instructions for installing CA Identity Manager and also includes information about optional components for installation such as Provisioning and CA SiteMinder.

This section contains the following topics:

Sample CA Identity Manager Installations

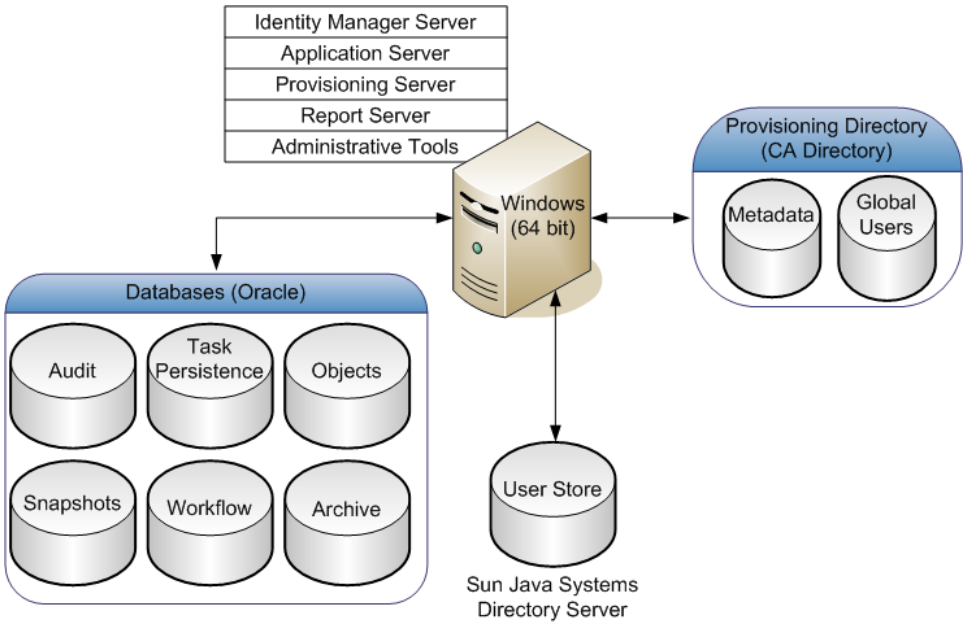
With CA Identity Manager, you can control user identities and their access to applications and accounts on endpoint systems. Based on the functionality you need, you select which CA Identity Manager components to install.

In all CA Identity Manager installations, the CA Identity Manager Server is installed on an application server. You use the CA Identity Manager Installer to install the other components you need.

The following sections illustrate some examples of CA Identity Manager implementations at a high level.

Example: Single Node Installation

In a single node installation, the CA Identity Manager Server is installed on one application server node. Also, one copy of each provisioning component is installed, but components can be on different systems. The following figure is an example of a single node CA Identity Manager installation with a Provisioning Server on the same system and a Provisioning Directory on another system:



This example also illustrates choices for platforms. In this case:

- The CA Identity Manager server is installed on Windows.
- The user store is on the Sun Java Systems Directory server.
- The databases are on Oracle

These platforms are only examples. You can select other platforms instead.

CA Identity Manager Server

Executes tasks within CA Identity Manager. The J2EE CA Identity Manager application includes the Management Console (for configuring environments), and the User Console (for managing an environment).

CA Identity Manager Administrative Tools

Provides tools and samples for configuring and using CA Identity Manager. The tools include Connector Xpress, the Java Connector Server SDK, configuration files, scripts, utilities, and JAR files that you use to compile custom objects with CA Identity Manager APIs and API samples. The Provisioning Manager and WorkPoint Designer are also included with the Administrative Tools.

The default installation location for most Administrative Tools follows:

- **Windows:** C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools
- **UNIX:** /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools

However, the default location for Provisioning Manager, which is only installed on Windows, follows:

C:\Program Files\CA\Identity Manager\Provisioning Manager

Note: The Tools\db directory also includes a document that describes the database schema.

Report Server

Uses CA Business Intelligence 3.2. You use this server to include data from the Snapshot Database, which contains information from the CA Identity Manager Object Store and the CA Identity Manager user store. An example of a Snapshot Report is the User Profile report. You can also create reports with a disabled snapshot applied, which include data from other data sources, such as the Audit Database.

CA Identity Manager Databases

Store data for CA Identity Manager. The databases store information for auditing, task persistence, snapshots (reporting), workflow, and CA Identity Manager objects. Each database must be a relational database.

Note: For a complete list of supported relational databases, see the CA Identity Manager support matrix on the CA Support Site.

CA Identity Manager User Store

Contains users and their information. This store can be a pre-existing user store already in use by the company. This user store can be LDAP or a relational database.

Note: For more information about setting up a user store for CA Identity Manager, see the *Configuration Guide*.

CA Identity Manager Provisioning Server

Manages accounts on endpoint systems. On the same system or another system, you can also install Connector Servers, which manage Java or C++ based connectors to endpoints.

CA Identity Manager Provisioning Directory

Specifies the Provisioning Directory schema to CA Directory. This schema sets up the Directory System Agents (DSAs) within CA Directory. The CA Identity Manager user store can also be the Provisioning Directory.

CA Identity Manager Provisioning Manager

Manages the Provisioning Server through a graphical interface. This tool is used for administrative tasks such as synchronizing accounts with account templates. The Provisioning Manager is installed as part of the CA Identity Manager Administrative Tools or can be installed separately from those tools.

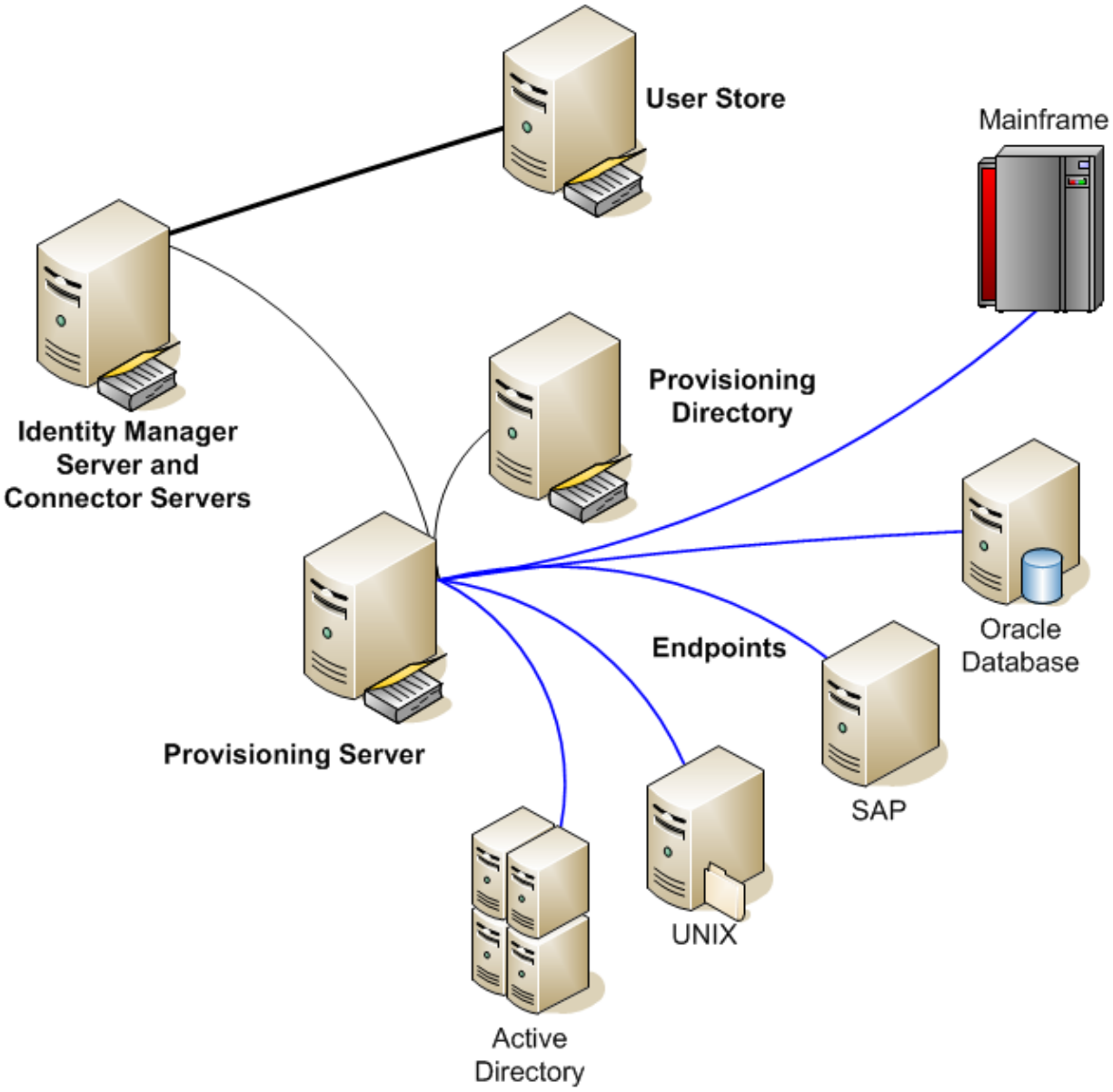
Note: This application runs on Windows only.

Example: Installation with Multiple Endpoints

Installing a Provisioning Server allows administrators to provision accounts on endpoints such as email servers, databases, and other applications to end users. To communicate with the endpoint systems, you install connector servers for endpoint-specific connectors, such as an SAP connector.

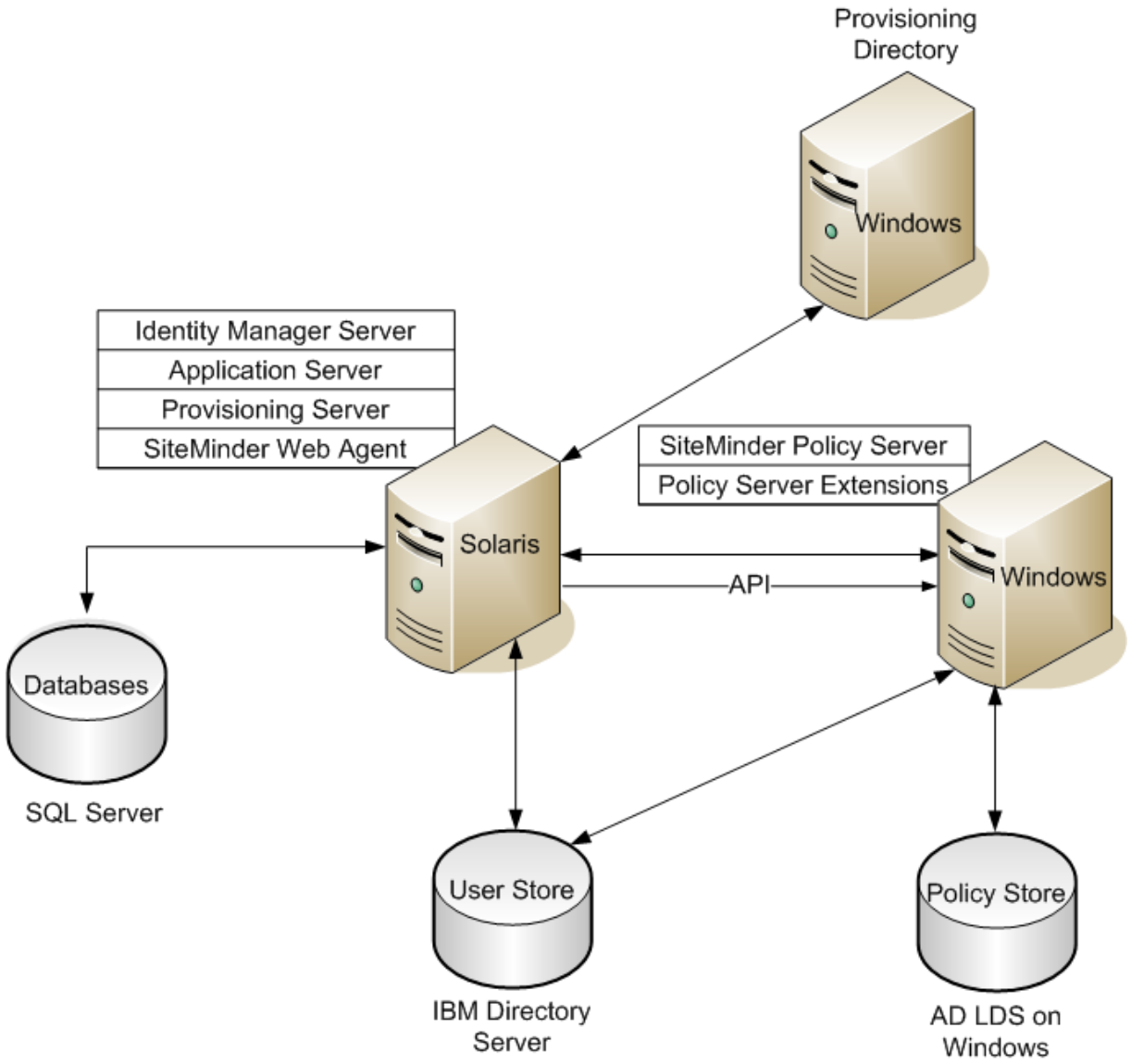
A typical installation scenario involves separate systems for the user store and the Provisioning Directory, which remained synchronized.

This example illustrates the use of CA Identity Manager to provide access to accounts on Active Directory, UNIX, SAP, Oracle, and mainframe systems.



Example: SiteMinder and CA Identity Manager Installation

CA Identity Manager can be integrated with a SiteMinder Policy Server, which provides advanced authentication and protection for your environments. The following figure is an example of a CA Identity Manager installation that uses a CA SiteMinder Policy Server for authentication and authorization:



The SiteMinder elements are defined as follows:

SiteMinder Web Agent

Works with the SiteMinder Policy Server to protect the User Console. Installed on the system with the CA Identity Manager Server.

SiteMinder Policy Server

Provides advanced authentication and authorization for CA Identity Manager and facilities such as Password Services, and Single Sign-On.

SiteMinder Policy Server Extensions

Enable a SiteMinder Policy Server to support CA Identity Manager. Install the extensions on each SiteMinder Policy Server system in your CA Identity Manager implementation.

The CA Identity Manager components are defined in the previous example on a single node installation; however, in this example, the components are installed on different platforms. The CA Identity Manager databases are on Microsoft SQL Server and the user store is on the IBM directory Server. The SiteMinder Policy Store is on AD LDS on Windows, which is one of several supported platforms for a policy store.

High Availability Installation

Before you install CA Identity Manager, consider the goals for your implementation. For example, one goal could be a resilient implementation that consistently provides good performance. Another goal could be scalability.

A high-availability implementation provides the following features:

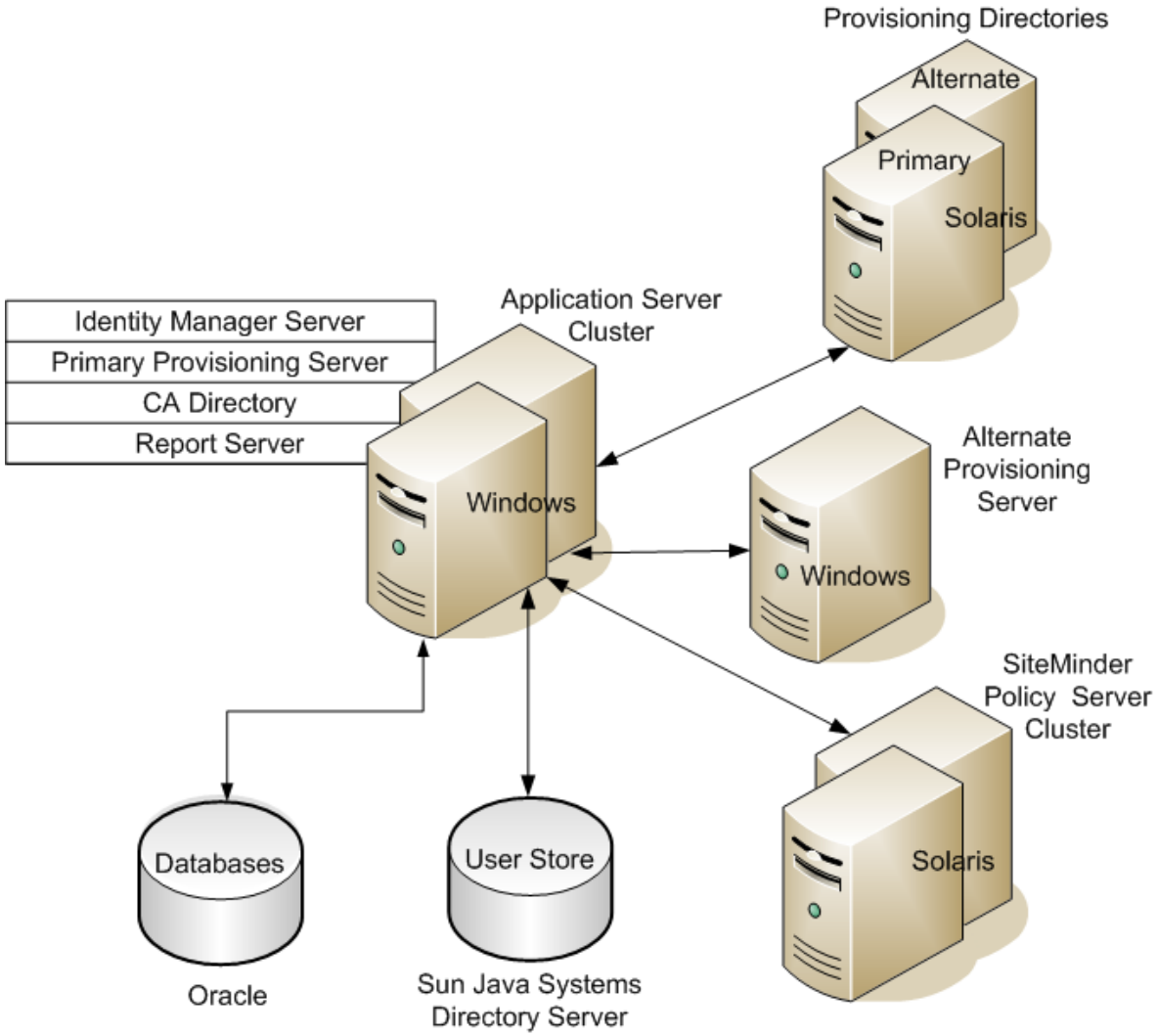
- Failover—Switches to another system automatically if the primary system fails or is temporarily offline for any reason.
- Load balancing—Distributes processing and communications activity evenly across a computer network so that performance remains good and no single device is overwhelmed.
- Various deployment tiers that provide the flexibility to serve dynamic business requirements.

To provide these high-availability features, the following implementation options exist:

- The CA Identity Manager Server can be installed on an application server cluster to allow the failover to any of the node in the cluster, providing uninterrupted access to users. The application server can be a 64-bit format, which provides better performance than a 32-bit application server.
- The Provisioning Server uses a CA Directory router to route traffic to a Provisioning Directory.
- CA Identity Manager includes connector servers that you configure per-directory or per-managed systems. Installing multiple connector servers increases resilience. Each connector server is also an LDAP server, similar to the Provisioning Server.

Example: High Availability Installation

The following diagram is an example that provides high availability for the CA Identity Manager Server, Provisioning Server, Provisioning Directory, and SiteMinder Policy Server. The use of alternate components and clusters provide the high availability features.



In addition to illustrating high availability, this figure shows the different platforms that are used for the components comparing to the SiteMinder (see page 14) illustration. For example, the database uses Oracle instead of Microsoft SQL Server, which appeared in the previous illustration.

CA Identity Manager Server Architecture

A CA Identity Manager implementation may span a multi-tiered environment that includes a combination of hardware and software, including three tiers:

- Web Server tier
- Application Server tier
- Policy Server tier (optional)

Each tier may contain a cluster of servers that perform the same function to share the workload for that tier. You configure each cluster separately, so that you can add servers only where they are needed. For example, in a clustered CA Identity Manager implementation, a group of several systems may all have a CA Identity Manager Server installed. These systems share the work that CA Identity Manager Server has performed.

Note: Nodes from different clusters may exist on the same system. For example, an application server node can be installed on the same system as a Policy Server node.

Provisioning Components Architecture

Provisioning provides high availability solutions in the following three tiers:

- Client tier

  The clients are the CA Identity Manager User Console, CA Identity Manager Management Console and the Provisioning Manager. You can group clients that are together based on their geographic locations, organizational units, business functions, security requirements, provisioning workload, or other administration needs. Generally, we recommend keeping clients close to the endpoints they manage.

- Provisioning Server tier

  Clients use primary and alternate Provisioning Servers, in order of their failover preference. Client requests continue to be submitted to the first server until that server fails. In other words, the connection stays active until the server fails. If a failure occurs, the client reviews the list of configured servers in order of preference to find the next available server.

The Provisioning Server can have multiple connector servers in operation. Each connector server handles operations on a distinct set of endpoints. Therefore, your organization could deploy connector servers on systems that are close in the network to the endpoints. For example, assume that you have many UNIX /etc endpoints. In such case, install one connector server on each server so that each connector server controls only the endpoints on the server where it is installed.

Installing Connector Servers close to the endpoints also reduces delays in managing accounts on endpoints.

- CA Directory tier (Provisioning Directory)

  Provisioning Servers uses a CA Directory router to send requests to primary and alternate Provisioning Directories in order of preference.

Overall Installation Process

To install CA Identity Manager, perform the following steps:

1. Install the prerequisite hardware and software and configure your system as required.
2. Install the CA Identity Manager Server on a single node or an application server cluster.
3. (Optional) Configure separate databases.
4. (Optional) Install the report server.
5. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers for high availability provisioning capabilities.

Note: In this document, each chapter includes a checklist of the steps to install or configure a CA Identity Manager feature or component. That section begins with a How To title.

Chapter 2: Installation Prerequisites

This section contains the following topics:

Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
X	**1. Install prerequisite hardware and software and configure your system as required.**
	2. Perform one of these installations: ■ Single node installation ■ Installation on an application server cluster.
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5.(Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support the failover and load balancing.

Prerequisite Knowledge

This guide is intended for users who are familiar with Java, J2EE standards, and application server technology. This guide assumes that you have the following technical knowledge:

- An understanding of J2EE application servers and multi-tier architecture
- Experience with installing and managing the application server, including tasks such as the following:
  - Starting the application server
  - Installing a single node
  - Installing cluster to support high availability
- Experience with managing a relational database
- (Optional) Familiarity with SiteMinder concepts, terms, and Policy Server configuration tasks

How to Install Prerequisite Components

To install the prerequisite hardware and software for CA Identity Manager required for either a standalone or cluster installation:

✓	Step
	1. Verify that your system meets the hardware requirements.
	2. Install CA Directory.
	3. (Optional) Create a FIPS key.
	4. (Optional) Integrate with SiteMinder.
	5. Create a database.
	6. Set up the application server.
	7. Meet IPv6 requirements if installing on IPv6 systems.
	8. Verify the Provisioning requirements if installing on Solaris.
	9. Complete the Installation Worksheets with information you need for the CA Identity Manager installation program.

Check Hardware Requirements

CA Identity Manager Server

These requirements take into account the requirements of the application server installed on the system where you install the CA Identity Manager Server.

Component	Minimum	Recommended
CPU	Intel (or compatible) 2.0 GHz (Windows or Red Hat Linux), SPARC 1.5 GHz (Solaris) or POWER4 1.1 GHz (AIX)	Dual core Intel (or compatible) 3.0 GHz (Windows or Red Hat Linux), Dual core SPARC 2.5 GHz (Solaris) POWER5 1.5 GHz (AIX)
Memory	4 GB	8 GB
Available Disk Space	4 GB	8 GB
Temp Space	2 GB	4 GB
Swap/Paging Space	2 GB	4 GB
Processor	32-bit processor and operating system for small deployments 64-bit processor and operating system for intermediate and large deployments, dual core	64-bit processor and operating system, quad core

Provisioning Server or a Standalone Connector Server

Component	Minimum	Recommended
CPU	Intel (or compatible) 2.0 GHz (Windows) SPARC 1.5 GHz (Solaris)	Dual core Intel (or compatible) 3.0 GHz (Windows) SPARC 2.0 GHz (Solaris)
Memory	4 GB	8 GB
Available Disk Space	4 GB	8 GB

Component	Minimum	Recommended
Processor	32-bit processor and operating system for small deployments 64-bit processor and operating system for intermediate and large deployments, dual core	64-bit processor and operating system, quad core

Provisioning Directory

Component	Minimum	Recommended
CPU	Intel (or compatible) 1.5 GHz (Windows) SPARC 1.0 GHz (Solaris)	Dual core Intel (or compatible) 2.5 GHz (Windows) SPARC 1.5 GHz (Solaris)
Memory	4 GB	8 GB
Available Disk Space	2 to 10 GB, depending on the number of endpoint accounts ■ Compact—Up to 10,000 accounts, 0.25 GB per data file (total 1 GB) ■ Basic—Up to 400,000 accounts, 0.5 GB per data file (total 2 GB) ■ Intermediate (64 bit only)—Up to 600,000 accounts, 1 GB per data file, total 4 GB ■ Large (64 bit only)—Over 600,000 accounts, 2 GB per data file, total 8 GB	2 to 10 GB, depending on the number of endpoint accounts ■ Compact—Up to 10,000 accounts, 0.25 GB per data file (total 1 GB) ■ Basic—Up to 400,000 accounts, 0.5 GB per data file (total 2 GB) ■ Intermediate (64 bit only)—Up to 600,000 accounts, 1 GB per data file, total 4 GB ■ Large (64 bit only)—Over 600,000 accounts, 2 GB per data file, total 8 GB
Processor	32-bit processor and operating system for small deployments 64-bit processor, 64-bit operating system, and CA Directory (64- bit version) for intermediate and large deployments	64-bit processor and operating system

All Components on One System

Hosting the entire CA Identity Manager product on a single physical system is not recommended for production environments. However, to do so, the hardware requirements are as follows:

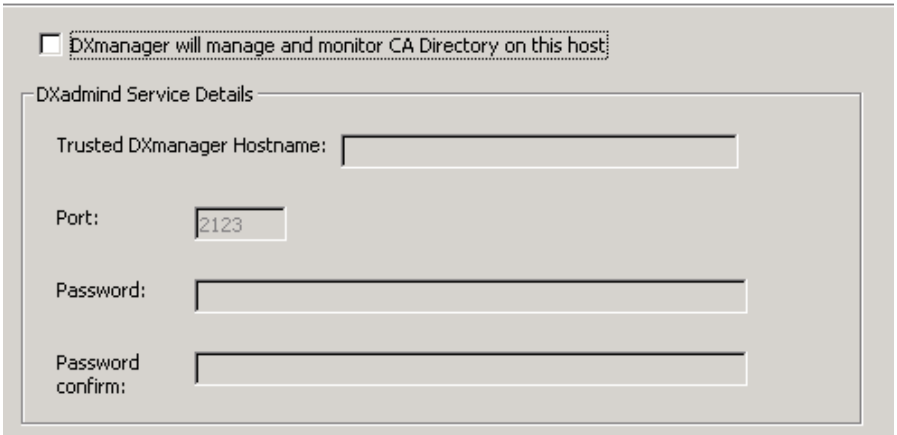
Component	Minimum
CPU	Intel (or compatible) 3.1 GHz (Windows) SPARC 2.5 GHz (Solaris)
Memory	8 GB
Available Disk Space	6 to 14 GB depending on the number of accounts
Processor	64-bit processor and operating system for intermediate and large deployments, quad core
Swap/Paging Space	6 GB

Install CA Directory

Before you install CA Identity Manager, install CA Directory using the following steps:

1. Install CA Directory on the system where you plan to install the Provisioning Directory. A supported version of CA Directory is included on your installation media. For details on installation, download the CA Directory documentation from the support site.

   Note: When the installer asks about installing dxadmind for DXManager, you can safely clear this option. The Provisioning Directory does not use DXManager.



2. Install a second copy of CA Directory on the system where you plan to install the Provisioning Server. This installation is for routing purposes, so that the Provisioning Server can communicate with the remote Provisioning Directory.

Important! We recommend that you disable all antivirus software before installation. During the installation, If antivirus software is enabled, problems can occur. Verify that you enable your antivirus protection again after you complete the installation.

Create a FIPS 140-2 Encryption Key

When you run the CA Identity Manager installer, you are given the option of enabling FIPS 140-2 compliance mode. For CA Identity Manager to support FIPS 140-2, all components in a CA Identity Manager environment must be FIPS 140-2 enabled. You need a FIPS encryption key to enable FIPS 140-2 during installation. A Password Tool for creating a FIPS key is located in the installation media at PasswordTool\bin.

Important! Use the same FIPS 140-2 encryption key in all installations. Verify that you safeguard the Password Tool generated key file immediately.

If you are using SiteMinder, be sure to set the ra.xml file correctly after CA Identity Manager installation. See the procedure Adding SiteMinder to an Existing CA Identity Manager Deployment in the *Configuration Guide*.

(Optional) Integrate with SiteMinder

A SiteMinder Policy Server is an optional component that you install as described in the *CA SiteMinder Installation Guide*. If you plan to make the Policy Server highly available, you configure it as a Policy Server cluster. You also install JCE libraries to enable communication with CA Identity Manager.

To install a Policy Server:

1. Install the SiteMinder Policy Server. For details, see the *CA SiteMinder Policy Server Installation Guide*.
2. To make the Policy Server highly available, install it on each node that should be in the Policy Server cluster.

   Note: Each Policy Server in the cluster uses the same policy store.
3. Verify that you can ping the systems that host the Policy Server from the system where you plan to install the CA Identity Manager Server.

To install the CA Identity Manager Extensions for SiteMinder:

Before you install the CA Identity Manager server, you add the extensions to each Policy Server. Assume that the Policy Server is on the system where you planned to install the CA Identity Manager server. Then, you can install the extensions and the CA Identity Manager server simultaneously. If so, omit this procedure.

1. Stop the CA SiteMinder services.
2. Set your default directory location to the root of the SiteMinder installation area.
3. Issue the following command:

   ```
   ./stop-all
   ```

   All SiteMinder executables shut down.
4. Install the CA Identity Manager Extensions for SiteMinder. Do one of the following tasks:

   - **Windows:** From your installation media, run the following program in the top-level folder:
     `ca-im-release-win32.exe`
   - **UNIX:** From your installation media, run the following program in the top-level folder:
     `ca-im-release-sol.bin`

   release represents the current release of CA Identity Manager.

5. Select Extensions for SiteMinder.
6. Complete the instructions in the installation dialog boxes.
7. Issue the following command:

   ```
   ./stop-all
   ```

   All SiteMinder executables shut down.
8. Issue the following command:

   ```
   ./start-all
   ```

   All SiteMinder executables start.

To install JCE Libraries:

The CA Identity Manager server requires the Java Cryptography Extension (JCE) libraries if you are also using CA SiteMinder.

Before you install the CA Identity Manager Server, perform these steps:

1. Download and install the Java Cryptography Extension Unlimited Strength Jurisdiction Policy Files.
2. Select the one that works with your application server and JDK.

   The download ZIP file includes a readme text file with installation instructions.

Create the Database

CA Identity Manager requires a relational database to store objects and data for auditing, snapshots (reporting), workflow, and task persistence. Install a supported version of Oracle or Microsoft SQL Server and create a database.

When installing CA Identity Manager, all of the database schemas are created automatically when the application server is started. However, after installing CA Identity Manager, you can configure separate databases for auditing, snapshots (reporting), workflow, and task persistence. To create these databases, see the chapter on Separate Database Configuration.

Note: We strongly recommend a separate database for task persistence. Using a separate database provides the best performance.

Install JBoss

CA Identity Manager r12.5 SP16 works with JBoss 5.0 and 5.1. Therefore, install a new version of JBoss if your version is before 5.0. You can install JBoss 5.0 or 5.1 on the same system as the previous version, but in a different file location from the previous version. Also, install the JDK identified in the support matrix as supporting JBoss 5.

Note: For a complete list of supported platforms and versions, see the CA Identity Manager support matrix on CA Support.

When using JBoss as the application server, note the following points:

- The CA Identity Manager Server is a J2EE application that is deployed on a supported application server. The iam_im.ear is deployed in the *jboss_home*/server/default/deploy folder. For a clustered installation, iam_im.ear is under jboss_home/server/all/deploy.

  Important! If any datastore file in the deploy directory is modified, JBoss loses the connection to that datastore and should be restarted.

- Install the required version of the JDK before installing the CA Identity Manager Server. You can download the JDK from the following Oracle URL:

  http://www.oracle.com/technetwork/java/index.html

Solaris Requirements

Provisioning Server Requirements

Verify /etc/system and verify the following minimum IPC kernel parameter values:

- set msgsys:msginfo_msgmni=32
- set semsys:seminfo_semmni=256
- set semsys:seminfo_semmns=512
- set semsys:seminfo_semmnu=256
- set semsys:seminfo_semume=128
- set semsys:seminfo_smmsl=128

- set shmsys:shminfo_shmmni=128
- set shmsys:shminfo_shmmin=4

Solaris 9 or 10 Requirements

Before installing provisioning software on Solaris 9 or 10, download and install the required patches.

1. Download the Sun Studio 10 patches for the Provisioning SDK from the following location:

   `http://developers.sun.com/prodtech/cc/downloads/patches/ss10_patches.html`

2. Download and install patch 117830.

   Note: Sun Studio 11 does not require patching.

3. Download Solaris 9 patches for all Provisioning components from the following location:

   `http://search.sun.com/search/onesearch/index.jsp`

4. Download and install 9_recommended.zip.

IPv6 Support

CA Identity Manager supports IPv6 on the following operating systems:

- Solaris 10
- Windows XP SP2 or higher
- Windows 2003 SP2 or higher
- Windows 2008 or higher

Note: The Java Connector Server does not support IPv6 on Microsoft Windows platforms. No JDK is available to work with IPv6 as of release time for CA Identity Manager r12.5 SP16. If a JDK is released that works with IPv6, the CA Identity Manager support matrix is updated on CA Support.

IPv6 JDK Requirements on JBoss

The following JDKs are required to support IPv6:

Application Server	JDK Requirement
JBoss (standalone)	JDK 1.5
JBoss cluster using an IPv4/IPv6 stack	JDK 1.5

JBoss cluster	JDK 1.5 for Solaris *only*. Note: For Windows, no JDK is available to work with IPv6. If a JDK is released that works with IPv6, the CA Identity Manager support matrix is updated on CA Support.

IPv6 Configuration Notes

Note the following points before configuring a CA Identity Manager Environment that supports IPv6:

- For CA Identity Manager to support IPv6 addresses, all components in the CA Identity Manager implementation (including the operating system, JDK, directory servers, and databases) must also support IPv6 addresses.
- If CA Identity Manager integrates with SiteMinder, the Web Server plug-in for the application server must also support IPv6.
- When you connect to SiteMinder or any database from CA Identity Manager using a JDBC connection, specify the hostname not the IP address.
- The Report Server can be installed on a dual-stack host, which supports IPv4 and IPv6, but the communication to the server must be IPv4.
- When you configure a connection to the Report Server in the Management Console, the server name must be in IPv4 format.
- CA Identity Manager does not support IPv6 link local addresses.
- In an IPv4/6 environment, if you want to configure CA Directory DSAs to listen on multiple addresses, add the additional addresses to your DSA knowledge files. For more information, see the CA Directory documentation.
- On a Windows 2008 system that uses IPv6, ensure that the IPv4 loopback address is enabled. Otherwise, the C++ Connector Server does not start.

Provisioning Directory on Windows 2008 with Pure IPv6 Not Supported

Due to a Sun Java Systems limitation, installing the Provisioning Directory on a Windows 2008 server with the IPv6 networking service uninstalled is not supported.

To work around with this issue, install the IPv6 service on the system and leave it disabled.

Complete the Installation Worksheets

The CA Identity Manager installation program asks you for information about previously installed software and the software that you are installing. Verify that you provide hostnames (and not IP addresses) in the installer screens.

Note: Use the following **Installation Worksheet** to record this information. We recommend that you complete the worksheet before starting the installation.

Provisioning Directory

Record the following Provisioning Directory and Provisioning Server information you need during the CA Identity Manager installation.

Field Name	Description	Your Response
Provisioning Directory Hostname	The hostname of the Provisioning Directory system if it is remote. You need the hostnames for the primary and any alternate Provisioning Directories.	
Shared Secret	The special password for the Provisioning Directory. Use the same password for the primary and any alternate Provisioning Directories.	
Provisioning Server Hostname	The host names of the primary and any alternate Provisioning Servers.	

JBoss Information

Record the following JBoss information that you need during the CA Identity Manager installation:

Field Name	Description	Your Response
JBoss Folder	The location of the application server home directory.	

Field Name	Description	Your Response
Access URL and port	The URL and port number for one of the following cases: ■ For a single node installation, the system that hosts the CA Identity Manager Server (system that hosts the application server). ■ For a cluster installation, the web server that provides load balancing.	
Java Virtual Machine	The path to the java executable for the JDK.	

Database Connection Information

Verify that an Oracle or Microsoft SQL Server Database is already configured and working. Record the following database information that you need during the CA Identity Manager installation:

Field Name	Description	Your Response
Database Type	The type (vendor/version) of a database that is created for task persistence, workflow, audit, reporting, object storage, and task persistence archive.	
Host Name	The hostname of the system where the database is located. **Note:** Verify that you have provided a hostname and *not* an IP address.	
Port Number	The port number of the database.	
Database Name	The database identifier.	
Username	The username for database access. **Note:** This user must have administrative rights to the database unless you plan to import the schema manually.	
Password	The password for the user account with administrative rights.	

Login Information

Record the following passwords which you need during the Provisioning Components installation.

Field Name	Description	Your Response
Username	A username that you create to log in to the provisioning components. Avoid the username siteminder if you have that product installed. This name conflicts with CA SiteMinder.	
Provisioning Server password	A password for this Server.	
C++ Connector Server password	A password is needed for this server. Each C++ Connector Server can have a unique password.	
Provisioning Directory password	A password which Provisioning Server uses to connect to Provisioning Directory. For an alternate Provisioning Server, enter the Provisioning Directory password which is created for the primary Provisioning Server.	

SiteMinder Information

If you plan to use a SiteMinder Policy Server to protect CA Identity Manager, record the following information:

Field Name	Description	Your Response
Policy Server Host Name	The hostname of the SiteMinder Policy Server.	
SiteMinder Administrator Name	The administrator username for the SiteMinder Policy Server.	
SiteMinder Administrator Password	The administrator user password for the SiteMinder Policy Server.	

Field Name	Description	Your Response
SiteMinder Folder (Solaris Only)	The location of SiteMinder on the system with a SiteMinder Policy Server installed.	
SiteMinder Agent Name	The name of the SiteMinder Agent that CA Identity Manager uses to connect to SiteMinder.	
SiteMinder Shared Secret	The shared secret of the given Agent Name.	

Chapter 3: Single Node Installation

This section contains the following topics:

Installation Status

This table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
X	**2. Perform one of these installations:** ■ **Single node installation** ■ **Installation on an application server cluster**
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5.(Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support the failover and load balancing.

CA Identity Manager Components

In a single node installation, you install one copy of each component, but use two or more systems for where you install them.

Note: If you intend to install multiple copies of components for high availability, see the chapters on installation on a cluster and high-availability provisioning installation.

Install one of each of the following components on a system at your site:

- CA Identity Manager Server—Installs the server that provides the core functionality of the product.
- CA Identity Manager Administrative Tools—Installs tools such as the Provisioning Manager, which runs on a Windows system, the SDK for the Java Connector Server, and Connector Xpress.

  Connector Xpress manages dynamic connectors, maps them to endpoints, and establishes routing rules. Dynamic connectors allow provisioning and management of SQL databases and LDAP directories.
- CA Identity Manager Provisioning Server—Enables provisioning in CA Identity Manager. Installation of this server includes the C++ Connector Server, which manages endpoints that use C++ connectors.
- Java Connector Server—Manages endpoints that use java connectors. The Java Connector Server is registered with the Provisioning Server when you install it.

  Note: You can instead install the Java Connector Servers separate from the Provisioning Server. See the *Java Connector Server Implementation Guide*.
- CA Identity Manager Provisioning Directory Initialization—Configures a CA Directory instance to store provisioning data. Use the installation program on each system where CA Directory is installed.
- Extensions for SiteMinder—Extends the SiteMinder Policy Server if you are using it to protect CA Identity Manager. Install these extensions on the same system as the Policy Server before you install the CA Identity Manager Server.

How to Perform a Single Node Installation

Use the following checklist to perform a basic installation of CA Identity Manager:

✓	Step
	1. Install CA Identity Manager on the systems required.
	2. Configure support for IPv6 if necessary.

✓	Step
	3. Verify that the CA Identity Manager Server starts.
	4. Configure Provisioning Manager if installed on a remote system.
	5. Install optional provisioning components.

Install CA Identity Manager Components

For a production environment, use separate systems for data servers. For example, we recommend that the Provisioning Directory and a database (SQL or Oracle) are on a separate system from the CA Identity Manager Server and the Provisioning Server. If you are installing SiteMinder, you can also prefer to have it on a separate system. The Administrative Tools can be installed on any system.

Use the CA Identity Manager installer to perform the installation on the systems required. In the procedures that follow, the step to run the installer refers to this program in the top-level folder of your installation media:

- **Windows:**
  `ca-im-release-win32.exe`
- **UNIX:**
  `ca-im-release-sol.bin`

release represents the current release of CA Identity Manager.

For each installed component, verify that you have the required information for installer screens (see page 30) such as host names and passwords. If any issues occur during installation, verify the installation logs (see page 117).

To install the Extensions for SiteMinder:

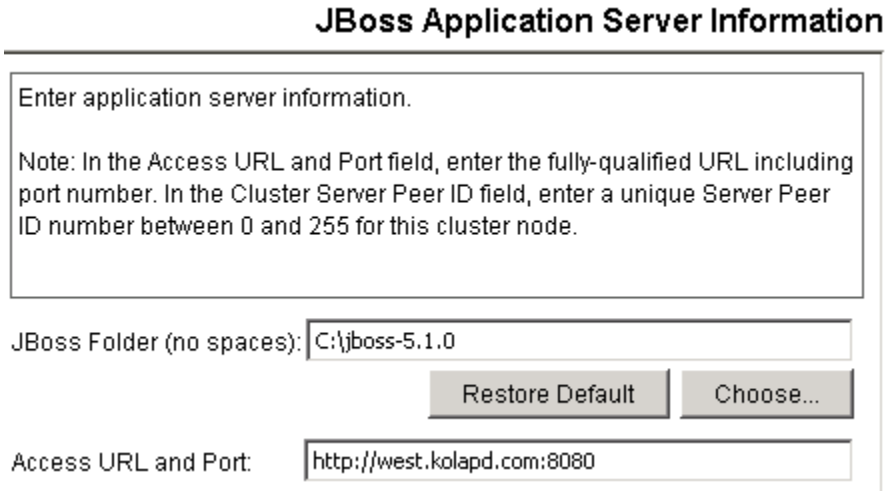
1. Log in to the system where SiteMinder is installed as a Local Administrator (for Windows) or root (for Solaris).
2. Stop the SiteMinder services.
3. Run the installer and select Extensions for SiteMinder.

To install the CA Identity Manager Server:

1. If you have installed SiteMinder on a separate system, install the extensions for SiteMinder on the same system.
2. Log in to the system where the application server is installed as a Local Administrator (for Windows) or root (for Solaris).
3. Stop the application server.

4. Run the installer and select the CA Identity Manager Server.

   Be sure to supply the port number that corresponds to the configuration of JBoss. Ports 1099 and 8080 are used by default. However, conflicts occur when the other applications on the system use these ports. For example, Oracle by default starts XDB service on port 8080. Configure either JBoss or the other application to use a different port.



5. If you have SiteMinder on the local system, select Extensions for SiteMinder. If it is on a remote system, select Connect to Existing SiteMinder Policy Server.

To install the Provisioning Directory:

1. Log in to the system as a Local Administrator (for Windows) or root (for Solaris).
2. Ensure that CA Directory is already installed on the system.
3. Run the installer and select the CA Identity Manager Provisioning Directory Initialization.

   Answer the question about deployment size.
4. Consider the following guidelines, while allowing room for future growth:
   - Compact—up to 10,000 accounts
   - Basic—up to 400,000 accounts
   - Intermediate (64 bit only)—up to 600,000 accounts
   - Large (64 bit only)—more than 600,000 accounts

   Note: If you are installing a Provisioning Directory in an established CA Identity Manager installation, be sure to make the deployment size large enough. Otherwise, an error occurs because the data does not fit when loaded into the data files. Intermediate and Large installations require 64-bit Directory installs (either Solaris or Windows 64 bit).

To install the Provisioning Server:

1. Log into the system as a Local Administrator (for Windows) or root (for Solaris).
2. Ensure that CA Directory is already installed and you have the details of the remote Provisioning Directory.
3. Run the installer and select the CA Identity Manager Provisioning Server.

Configure IPv6 Support

If you are installing on a IPv6 supported JBoss system, some configuration is required.

Follow these steps:

1. Open the run.bat/sh file located in *jboss_installation*\bin.
2. Uncomment *one* of the following properties in the IDM_OPTS entry:
   - For IPv6 only systems, uncomment the following entry:
     `#IDM_OPTS="$IDM_OPTS -Djava.net.preferIPv6Addresses=true"`
   - For IPv6/IPv4 systems, uncomment the following entry:
     `#IDM_OPTS="$IDM_OPTS -Djava.net.preferIPv4Stack=true"`

     Note: These properties are shown for UNIX; however, they are the same as found after a REM in Windows.
3. Save the file.

Verify the CA Identity Manager Server Installation

To start CA Identity Manager on JBoss, you use the run.bat file for Windows, or the run.sh file on UNIX. This file is located in the bin directory where JBoss is installed.

Follow these steps:

1. Start the CA Identity Manager server used databases.
2. Start the CA Identity Manager Server as follows:
   - **Windows:** Go to Start, Programs, CA, Identity Manager, Start Identity Manager Server.
   - **UNIX:** Enter the following command from the *jboss_home*/bin directory:
     `./run.sh`
3. Wait until you see that the server has started. This message appears in a console window:

   ```
   DATE+TIME INFO  [com.sun.jersey.server.impl.application.WebApplicationImpl]
   (main) Initiating Jersey application, version 'Jersey: 1.1.5.1 DATE+TIME'
   ```
4. Access the Management Console and confirm the following points:
   - You can access the following URL from a browser:

     http://*im_server*:*port*/iam/immanage

     For example:

     http://MyServer.MyCompany.com:*port-number/*iam/immanage
   - The Management Console opens.
   - No errors are displayed in the application server log.
   - You do not receive an error message when you click the Directories link.
5. Verify that you can access an upgraded environment using this URL format:

   http://*im_server*:*port*/iam/im/*environment*

Configure a Remote Provisioning Manager

If you installed the Provisioning Manager on a different system from the Provisioning Server, you configure communication to the server.

Note: To install the Provisioning Manager, install the CA Identity Manager Administrative Tools on a Windows system.

Follow these steps:

1. Log in to the Windows system where you installed Provisioning Manager.
2. Go to Start, Programs, CA Identity Manager, Provisioning Manager Setup.
3. Enter the hostname of the Provisioning Server.
4. Click Configure.
5. For an alternate Provisioning Server, select the domain name from the pull-down list.

6. Click Ok.

   You can now start the Provisioning Manager and see the domain name that you configured.

Install Optional Provisioning Components

Optional Provisioning Components for CA Identity Manager are in the im-pc-*release*.zip

release represents the current release of CA Identity Manager.

The ZIP file includes the following:

SPML Manager

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

SPML Service

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

Remote Agents

Run the specific agent installer from the Provisioning Component media (under \RemoteAgent) to install these components. If you want IPv6 support, you must install your agents.

Password Sync Agents

Run the Password Sync Agent installer from the Provisioning Component media (under \Agent) to install this component.

GINA

Run the GINA installer from the Provisioning Component media (under \Agent) to install this component.

Credential Provider

Run the Vista Credential Provider installer from the Provisioning Component media (under \Agent) to install this component.

Bulk Loader Client/PeopleSoft Feed

Run the Bulk Loader Client installer from the Provisioning Component media (under \Clients) to install this component.

JCS SDK

Run the JCS SDK installer from the CA Identity Manager media (under \Provisioning) to install this component.

CCI Standalone

Run the CCI Standalone installer from the Provisioning Component media (under \Infrastructure) to install this component.

The CA Identity Manager installer installs all connectors by default. However, in some cases, install an agent on an endpoint system you are managing before you can use the related connector.

Connectors run on the Provisioning Server and communicate with the systems managed by an endpoint. For example, systems running Active Directory Services (ADS) can be managed only if the ADS Connector is installed on the Provisioning Server.

Note: For more information about each connector, see the *Connectors Guide*

More information exists for these components in the following guides:

- Credential Provider (*Administration Guide)*
- GINA (*Administration Guide)*
- Password Synchronization Agent (*Administration Guide)*
- Connector Xpress (*Connector Xpress Guide)*
- SPML Service (*Provisioning Reference Guide)*
- Agents for use with connectors (*Connectors Guide)*

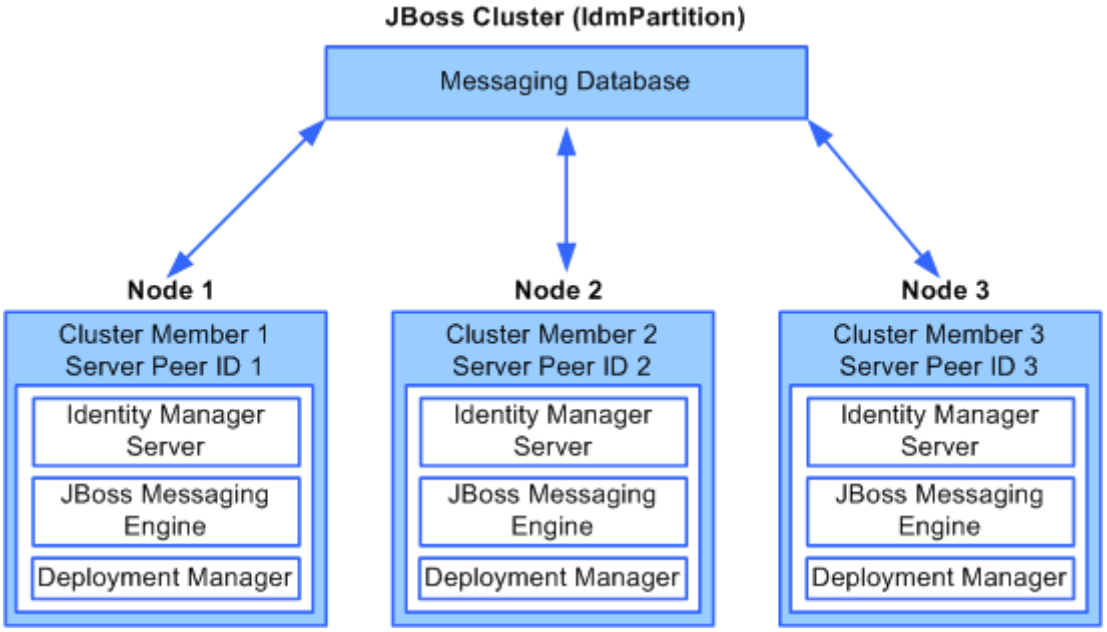
Chapter 4: Installation on a JBoss Cluster

This section contains the following topics:

Example: CA Identity Manager Server on a JBoss Cluster

CA Identity Manager uses the farming method of a JBoss cluster. In this type of cluster, you create a master node and it is usually the node that starts first in the cluster. As other nodes start, they receive deployment files from the master node. If the master node fails, another node becomes the new master node.

The following figure shows the relationship between the nodes and cluster members. Each node contains one cluster member. Each member of the cluster has a unique Server Peer ID. The master node would be cluster member 1, assuming it was created first.



In this figure, the messaging database is a central store for cluster members to share messages and each node contains three components:

CA Identity Manager Server

Provides the core functionality of the product.

JBoss Messaging Engine

Provides messaging functionality for members of the cluster using JMS.

Deployment Manager

Keeps track of the cluster members and the current master node member, which is also responsible for deploying files from the master node to other nodes.

Installation Status

This table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
X	**2. Perform one of these installations:** ■ **Single node installation** ■ **Installation on an application server cluster**
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
	5. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

How to Install CA Identity Manager on a JBoss Cluster

The following procedures describe how to set up multiple JBoss application servers with the same CA Identity Manager application on each server. In this type of cluster, each JBoss application server acts independently of the other application servers, but they share the load through JMS messaging.

✓	Step
	1. Test the default multicast address.

✓	Step
	2. Create the master node.
	3. Add cluster nodes.
	4. Configure the JK Connector

Test the Default Multicast Address

The run script uses a multicast address, either the default address or an alternative address supplied by your network administrator.

Follow these steps:

1. Install JBoss and the JDK on the computer.
2. Run sender on first node as follows:
   a. Navigate to jboss-home-1/server/all/lib.
   b. Run: java -cp jgroups.jar org.jgroups.tests.McastSenderTest -mcast_addr 224.10.10.10 -port 5555
3. Run receivers on other nodes in the cluster as follows:
   a. Navigate to jboss-home-*N*/server/all/lib.

      N represents the next node in the cluster.

   b. Run: java -cp jgroups.jar org.jgroups.tests.McastReceiverTest -mcast_addr 224.10.10.10 -port 5555
4. Send a message from the first node as follows:
   a. On the console of the first node, enter any text and press enter.
   b. Confirm that a reply appears, to acknowledge the text was sent.
   c. Confirm that the message appears on the console of all other nodes in the cluster.
   d. If either the send or receive test fails, ask your network administrator to provide a multicast address that works and repeat this test.

Create the Master Node for JBoss 5

You begin creating the JBoss cluster by creating the master node, the first node in the cluster.

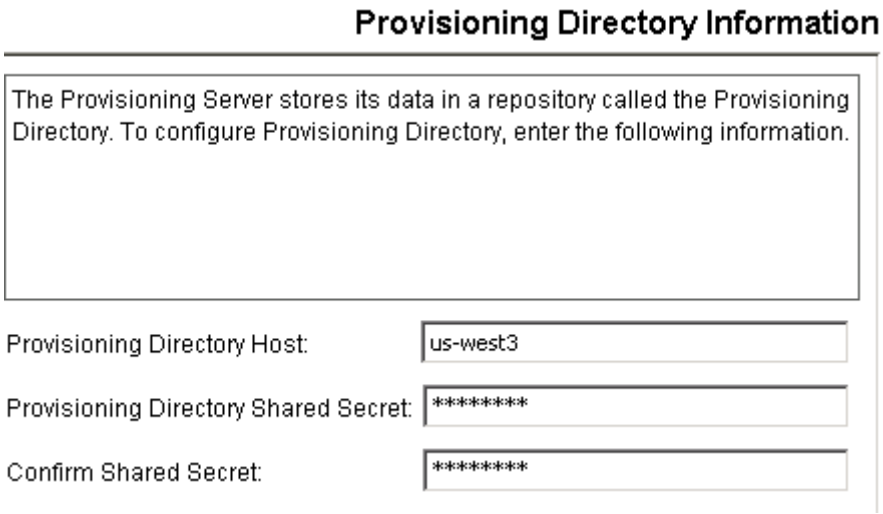
Note: On Windows, IPv6 is not supported for a JBoss cluster with the current release of the JDK 1.6, the JDK that works with JBoss 5. Each node must be an IPv4 system or part of an IPv4/IPv6 stack.

Follow these steps:

1. Install JBoss 5 64-bit and the JDK 1.6 on the computer.
2. Start the CA Identity Manager installation program.
   - Windows: From your installation media, run the following program:
     `ca-im-release-win32.exe`
   - UNIX: From your installation media, run the installation program. For example, for Solaris:
     `ca-im-release-sol.bin`

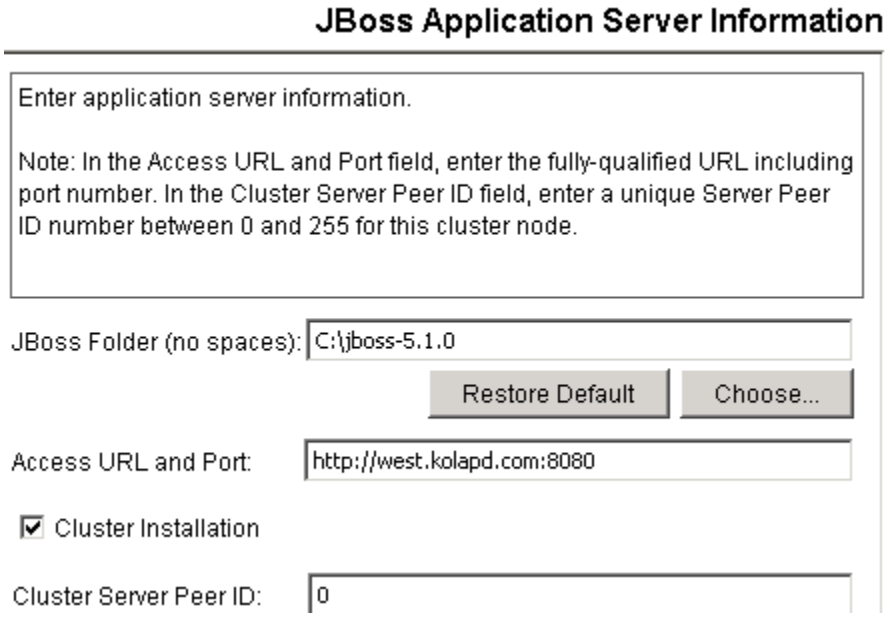
   release represents the current release of CA Identity Manager.

   Important! Make sure that you have the collected the information needed by the installer, such as user names, host names, and ports.
3. Complete the Select Components section by including the CA Identity Manager Server and any other components that you need on this system.
4. Complete the other sections based on your requirements for the installation.
5. When you enter any password or shared secret in the installation, be sure to provide a password that you can recall when needed.



6. Complete the JBoss Application Server Information page as follows:
   a. Enter the Access Server URL and port with the URL and port number of the web server used for load balancing.
   b. Select Cluster Installation.
   c. Enter a Peer ID, a unique number between 0 and 255. Make a record of the Peer ID, so that you use a different number for other nodes.

Figure 1: The user enters JBoss information.



7. If the multicast address test failed, perform one of the next two steps for Windows or Solaris.
8. On a Windows system, edit run.bat in the *jboss_home*\bin directory:
   a. Locate the line that begins as follows:
      `ARGS="${ARGS}`
   b. Add a multicast address preceded by the -u argument as follows:
      `ARGS="${ARGS} -g IdmPartition -Djboss.messaging.ServerPeerID=`*`PeerID`*` -u `*`multicast-address`*`"`
   c. If you are installing on a system that supports IPv6/IPv4, uncomment the IDM_OPTS entry:
      `set IDM_OPTS=%IDM_OPTS% -Djava.net.preferIPv4Stack=true`

9. For a Solaris system, edit run.sh in the *jboss_home*\bin directory:

   a. Locate the line that begins as follows:
   `ARGS="${ARGS}`

   b. Add a multicast address preceded by the -u argument as follows:
   `ARGS="${ARGS} -g IdmPartition -Djboss.messaging.ServerPeerID=PeerID -u multicast-address"`

   c. If you are installing on a system that supports IPv6, modify one of the following properties in the IDM_OPTS entry:

   - For IPv6 only systems, uncomment the following entry:
     `IDM_OPTS="$IDM_OPTS -Djava.net.preferIPv6Addresses=true"`
   - For IPv6/IPv4 systems, uncomment the following entry:
     `IDM_OPTS="$IDM_OPTS -Djava.net.preferIPv4Stack=true"`

If any issues occur during installation, inspect the installation logs (see page 117).

Add Cluster Nodes

We recommend that you install each cluster node on a separate system. However, if you install all nodes on one system, each node needs a separate jboss_home. This precaution is necessary to avoid contention over the workpoint.log in the *jboss_home*/bin directory.

Follow these steps:

1. Install JBoss and the JDK on the computer.
2. Install the CA Identity Manager server on that system.
   - Windows: From your installation media, run the following program:
     `ca-im-release-win32.exe`
   - UNIX: From your installation media, run the following program:
     `ca-im-release-sol.bin`

   release represents the current release of CA Identity Manager.
3. Be sure to supply the same values for FIPS, SiteMinder, database, and shared secret details and all other values entered for the master node.
4. Select Cluster Installation.
5. Enter a Peer ID that is different from the other nodes you have created.
6. If the multicast address test failed, perform one of the next two steps for Windows or Solaris.
7. On a Windows system, edit run.bat in the *jboss_home*\bin directory:
   a. Locate the line that begins as follows:
      `ARGS="${ARGS}`
   b. Add a multicast address preceded by the -u argument as follows:
      `ARGS="${ARGS} -g IdmPartition -Djboss.messaging.ServerPeerID=PeerID -u multicast-address"`
   c. If you are installing on a system that supports IPv6/IPv4, uncomment the IDM_OPTS entry:
      `set IDM_OPTS=%IDM_OPTS% -Djava.net.preferIPv4Stack=true`
8. For a Solaris system, edit run.sh in the *jboss_home*\bin directory:
   a. Locate the line that begins as follows:
      `ARGS="${ARGS}`
   b. Add a multicast address preceded by the -u argument as follows:
      `ARGS="${ARGS} -g IdmPartition -Djboss.messaging.ServerPeerID=PeerID -u multicast-address"`
   c. If you are installing on a system that supports IPv6, modify one of the following properties in the IDM_OPTS entry:
      - For IPv6 only systems, uncomment the following entry:
        `IDM_OPTS="$IDM_OPTS -Djava.net.preferIPv6Addresses=true"`
      - For IPv6/IPv4 systems, uncomment the following entry:
        `IDM_OPTS="$IDM_OPTS -Djava.net.preferIPv4Stack=true"`

If any issues occur during installation, inspect the installation logs (see page 117).

Configure the JK Connector

Follow these steps:

1. Install a JK connector based on these instructions:

   http://community.jboss.org/wiki/usingmodjk12withjboss

2. Note the following when you use this procedure:

   a. When you configure the modjk workers, use the workers.properties file in this location:

      Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\samples\ConnectorConfiguration\windows\IIS_JBoss

      UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/samples/Cluster/JBoss/ConnectorConfiguration

   b. In this file, replace worker.worker*N*.* (the worker name) with your corresponding node's Peer ID.

      If you have more than two nodes, copy a worker.workerN.* set for each additional node and rename the worker name.

   c. Fill in the worker.worker*N*.host field with your corresponding nodes' hostnames.

      For example, consider a cluster where the CA Identity Manager server is installed on three JBoss hosts named myhostA, myhostB, and myhostC, using Peer IDs 1, 2, and 3. The workers.properties file appears as follows:

```
worker.worker1.port=8009
worker.worker1.host=myhostA
.
.
.
worker.worker1.recovery_options=28

worker.worker2.port=8009
worker.worker2.host=myhostB
.
.
.
worker.worker2.recovery_options=28

worker.worker3.port=8009
worker.worker3.host=myhostC
.
.
.
worker.worker3.recovery_options=28
.
.
.
worker.router.balance_workers=worker1,worker2,worker3
```

d. Copy the uriworkermap.properties file in the above location to *APACHE_HOME*/conf.

e. Omit the step about configuring Tomcat for session stickiness. This feature is already configured by the installer and in the workers.properties file.

Start the JBoss Cluster

Once all configuration is complete, start all servers in the following order.

Follow these steps:

1. Start one of the SiteMinder Policy Servers that supports CA Identity Manager.

   Note: If you have a Policy Server cluster, only one Policy Server should be running while you create CA Identity Manager directories, create or modify CA Identity Manager environments, or change WorkPoint settings.

2. From a command line, navigate to:

   jboss_home/bin

3. Enter the following command to start the CA Identity Manager server:
   - For Windows:

     run.bat -c all
   - For UNIX:

     ./run.sh -c all
4. Wait till you see that the server has started. This message appears in a console window:

   ```
   DATE+TIME INFO  [com.sun.jersey.server.impl.application.WebApplicationImpl]
   (main) Initiating Jersey application, version 'Jersey: 1.1.5.1 DATE+TIME'
   ```
5. If you have already installed a SiteMinder Web Agent, start the Web Server where you installed the SiteMinder Web Agent and the application server proxy plug-in.

Verify the Clustered Installation

When you have completed all steps and started the cluster, check that the installation was successful.

Follow these steps:

1. Start the databases used by the CA Identity Manager server.
2. Start any extra Policy Servers and CA Identity Manager nodes that you stopped.
3. Access the Management Console and confirm the following points:
   - You can access the following URL from a browser:

     http://*IdentityMinder_server_node:port*/iam/immanage

     For example:

     http://MyServer.MyCompany.com:*port-number/*iam/immanage
   - The Management Console opens.
   - No errors are displayed in the application server log.
   - You do not receive an error message when you click the Directories link.
4. Verify that you can access an upgraded environment using this URL format:

   http://*web_server_proxy_host*/iam/im/*environment*

Configure a Remote Provisioning Manager

If you installed the Provisioning Manager on a different system from the Provisioning Server, you configure communication to the server.

Note: To install the Provisioning Manager, install the CA Identity Manager Administrative Tools on a Windows system.

Follow these steps:

1. Log in to the Windows system where you installed Provisioning Manager.
2. Go to Start, Programs, CA Identity Manager, Provisioning Manager Setup.
3. Enter the hostname of the Provisioning Server.
4. Click Configure.
5. For an alternate Provisioning Server, select the domain name from the pull-down list.
6. Click Ok.

   You can now start the Provisioning Manager and see the domain name that you configured.

Install Optional Provisioning Components

Optional Provisioning Components for CA Identity Manager are in the im-pc-*release*.zip

release represents the current release of CA Identity Manager.

The ZIP file includes the following:

SPML Manager

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

SPML Service

Run the SPML installer from the Provisioning Component media (under \Clients) to install this component.

Remote Agents

Run the specific agent installer from the Provisioning Component media (under \RemoteAgent) to install these components. If you want IPv6 support, you must install your agents.

Password Sync Agents

Run the Password Sync Agent installer from the Provisioning Component media (under \Agent) to install this component.

GINA

Run the GINA installer from the Provisioning Component media (under \Agent) to install this component.

Credential Provider

Run the Vista Credential Provider installer from the Provisioning Component media (under \Agent) to install this component.

Bulk Loader Client/PeopleSoft Feed

Run the Bulk Loader Client installer from the Provisioning Component media (under \Clients) to install this component.

JCS SDK

Run the JCS SDK installer from the CA Identity Manager media (under \Provisioning) to install this component.

CCI Standalone

Run the CCI Standalone installer from the Provisioning Component media (under \Infrastructure) to install this component.

The CA Identity Manager installer installs all connectors by default. However, in some cases, install an agent on an endpoint system you are managing before you can use the related connector.

Connectors run on the Provisioning Server and communicate with the systems managed by an endpoint. For example, systems running Active Directory Services (ADS) can be managed only if the ADS Connector is installed on the Provisioning Server.

Note: For more information about each connector, see the *Connectors Guide*

More information exists for these components in the following guides:

- Credential Provider (*Administration Guide)*
- GINA (*Administration Guide)*
- Password Synchronization Agent (*Administration Guide)*
- Connector Xpress (*Connector Xpress Guide)*
- SPML Service (*Provisioning Reference Guide)*
- Agents for use with connectors (*Connectors Guide)*

Chapter 5: Separate Database Configuration

This section contains the following topics:

Installation Status

This table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Single node installation ■ Installation on an application server cluster
X	**3. (Optional) Create separate databases.**
	4. (Optional) Install the Report Server.
	5. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

Create Separate Databases

CA Identity Manager requires a relational database to store objects and data for auditing, snapshots (reporting), workflow, and task persistence. When installing CA Identity Manager, all of the database schemas are created automatically when the application server is started. However, for scalability purposes, you may want to create a separate database to replace any one of the existing database schemas initially created by CA Identity Manager during installation.

You can create a database instance for the following:

- Workflow
- Auditing
- Task Persistence
- Object Store
- Snapshots (reporting)
- Archive (task persistence archive)

Important! The Windows default locations for CA Identity Manager database schema files are the following:

- Workflow: See the section, Run the CreateDatabase script.
- Auditing: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db
- Task Persistence: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db
- Object Store: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db
- Snapshots (reporting): C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\imrexport\tools\db
- Archive (task persistence archive): C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db

How to Create Separate Databases

To create separate databases for CA Identity Manager:

✓	Step
	1. Create a Microsoft SQL Server or Oracle database instance for CA Identity Manager.
	2. Edit the data source.
	3. (Optional) Run the SQL scripts.

Create an MS SQL Server Database Instance

Follow these steps:

1. Create a database instance in SQL server.
2. Create a user and grant this user the necessary rights (such as public and db_owner rights) to the database by editing the properties of the user.

   Note: The user must have at least select, insert, update, and delete permissions for all of the tables created by the .sql script for creating the database, and must be able to execute all of the stored procedures (if applicable) defined in these scripts. For example, the user must have these permissions on the tables defined in the following default location:

   C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\taskpersistence\sqlserver\idm_db_sqlserver.sql
3. While editing the user's properties, set the database you just created as the default database for the user.
4. Ensure the Authentication setting has a value of SQL Server on the Security tab of the SQL Server Properties dialog for the server where the database is installed.

Note: For complete information about Microsoft SQL Server, see your Microsoft SQL Server documentation.

Create an Oracle Database Instance

Follow these steps:

1. Create a new tablespace.
2. Create a new user.

3. Grant the user rights to the new database.
   - Create/alter/drop tables
   - Create/alter/drop view
   - Create/alter/drop INDEX
   - Create/replace/drop stored procedures
   - Create/replace/drop functions
   - Create/drop sequence
   - Create/replace/drop triggers
   - Create/replace/drop types
   - Insert/select/delete records
   - CREATE SESSION / connect to database
4. Give DBA rights to the user.

Note: For complete information about Oracle, see your Oracle documentation.

Edit the Data Source

To edit the data source

1. In a text editor, open the appropriate data source descriptor located in the jboss_home/server/default/deploy directory, or jboss_home/server/all/deploy for a clustered installation.

   The following are the JNDI data source descriptors:

   - Task Persistence: iam/im/jdbc/jdbc/idm
   - Workflow: iam/im/jdbc/jdbc/WPDS
   - Auditing: iam/im/jdbc/auditDbDataSource

- Snapshots: iam/im/jdbc/jdbc/reportsnapshot
- Object Store: iam/im/jdbc/jdbc/objectstore
- Archive: iam/im/jdbc/jdbc/archive

2. Change the DatabaseName, User, and Password in the data source descriptor to the appropriate values for the new database.

   Important! For your version of JBoss, the username and password may instead be in jboss_home\server\default\conf\login-config.xml. If so, you can create a JBoss security realm, which is required to support FIPS. This approach also avoids having a username and password in clear text. For more information, see the Configuration Guide.

The database schema (SQL scripts) will be automatically applied when you restart CA Identity Manager.

Run the SQL Scripts

SQL scripts are automatically run against the databases when CA Identity Manager starts, however if you want to run the SQL scripts yourself, perform the following steps before restarting the application server:

These scripts are installed with the CA Identity Manager Administrative Tools.

Follow these steps:

1. Do one of the following:
   - Microsoft SQL Server: Open the Query Analyzer tool and select the script you need.
   - Oracle: Open the SQL prompt for the script you need.
2. Select one of the following scripts (shown with the default Windows locations) depending on what the database was created for:
   - Task Persistence:
     - Microsoft SQL Server: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\taskpersistence\sqlserver\idm_db_sqlserver.sql
     - Oracle on Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\taskpersistence\oracle9i\idm_db_oracle.sql
     - Oracle on UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/db/taskpersistence/oracle9i/idm_db_oracle.sql

- Auditing:
  - Microsoft SQL Server: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\auditing\sqlserver\ims_mssql_audit.sql
  - Oracle on Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\db\auditing\oracle\ims_oracle_audit.sql
  - Oracle on UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/db/auditing/oracle/ims_oracle_audit.sql
- Snapshots:
  - Microsoft SQL Server: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\sqlserver\ims_mssql_report.sql
  - Oracle on Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\oracle\ims_oracle_report.sql
  - Oracle on UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/imrexport/db/oracle/ims_oracle_report.sql
- Workflow: See the Section "Run the SQL Scripts for Workflow."

3. Run the script file.
4. Verify that no errors appeared when you ran the script.

Run the Script for Workflow

CA Identity Manager includes SQL scripts for setting up a new workflow database instance.

To run the CreateDatabase script:

Follow these steps:

1. Add the path to the sqljdbc.jar to the DB_CLASSPATH attribute in the CreateDatabase.bat or .sh script before you run it.
2. From a command prompt, run CreateDatabase.bat or sh. The default location for this script is:

   Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\Workpoint\install.

   UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/Workpoint/install.

   A command prompt window and the WorkPoint application open.
3. Select the database type from the drop-down.

4. Use the following guidelines to fill in fields in the configuration utility:
   - For the JDBC Class parameter, enter:

     Oracle: oracle.jdbc.driver.OracleDriver

     SQL Server: com.microsoft.sqlserver.jdbc.SQLServerDriver
   - For the JDBC URL, enter:

     Oracle: jdbc:oracle:thin:@*wf_db_system*:1521:*wf_oracle_SID*

     SQL Server: jdbc:sqlserver://*wf_db_system*:1433; databaseName=*wf_db_name*
   - For the Database User ID parameter, enter the workflow user you created when creating the workflow database.
   - For the Password parameter, enter the password you created for the workflow user.
   - For the Database ID, enter WPDS
5. Accept the default check box selections.
6. Click the Initialize button.

   When the configuration is complete, a message that resembles the following appears in the Command Prompt window:

   ```
   The create database process finished with 0 errors.
   ```
7. Restart the application server.

Chapter 6: Report Server Installation

This section contains the following topics:

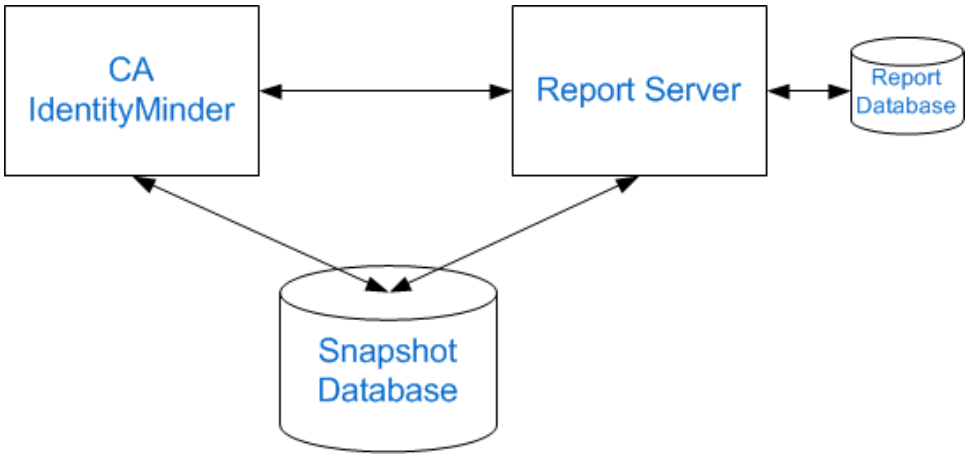
Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Single node installation ■ Installation on an application server cluster
	3. (Optional) Create separate databases.
X	**4. (Optional) Install the Report Server.**
	5. (Optional) Configure the SSL Certificate in the Report Server.
	6. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.

Reporting Architecture

In CA Identity Manager, the reporting setup requires the three major components in the following diagram:



Note: The Snapshot Database in this illustration graphic could also be the Audit Database or Workflow Database.

Report Server

Also known as CA Business Intelligence, this server generates reports, communicating directly with CA Identity Manager and the Snapshot Database.

Report Database

The database where the CA Report Server (Business Objects) stores its own data.

CA Identity Manager

CA Identity Manager allows you to export CA Identity Manager object data to the Report Database.

Snapshot Database

A separate database containing the snapshot data of objects in CA Identity Manager

Important! The Report Server uses Business Objects Enterprise. If you already have a Report Server in your environment and want to use it with CA Identity Manager, the minimum version required by CA Identity Manager is CA Business Intelligence 3.2 SP3.

Reporting Considerations

Consider the following before installing the Report Server:

- Installing the Report Server can take up to two hours.

- If JBoss is installed on the computer where you are installing the Report Server, port conflicts may occur. If Apache Tomcat is the web server, you can locate JBoss port information in the following files:
  - jboss-service.xml

    Default location: *jboss_home*\server*server_configuration*\conf
  - server.xml

    Default location:
    jboss_home\server*server_configuration*\deploy\jboss-web.deployer

  jboss_home

  Specifies the JBoss installation path.

  server_configuration

  Specifies the name of your server configuration.

  Default value: default

  Note: Restart JBoss if you make changes to either of these files.

Hardware Requirements

The hardware requirements for the Report Server are based on the operating system. See the PDF with the filename that matches your operating system in the *installer-media-root-directory*/Docs folder.

Note: For more information about supported OS versions and databases, see the Business Objects website.

How to Install the Report Server

The following checklist describes the steps to install the reporting feature of CA Identity Manager:

✓	Step
	1. Review the report pre-installation checklist.
	2. Gather reporting information.
	3. Open ports required by the Report Server.
	4. Install the Report Server (CA Business Intelligence).
	5. Run the Registry Script.

✓	Step
	6. Copy the JDBC JAR files.
	7. Bypass the proxy server.
	8. Deploy the default reports.
	9. Perform a post-installation step.

Note: For more information about configuring reporting after the installation, see the *Administration Guide*.

Reports Pre-Installation Checklist

Print the following checklist to be sure that you meet the minimum system and database requirements before installing the Report Server:

- Be sure that the Windows or UNIX system on which you are installing the Report Server meets the minimum system requirements.
- If you create a database instance for the Snapshot Database, run the following scripts on the new database:
  - Microsoft SQL: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\sqlserver\ims_mssql_report.sql
  - Oracle: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\imrexport\db\oracle\ims_oracle_report.sql

    To execute these scripts, the database user needs DBA, connect, and resource roles and system privileges to create tables, indexes, sessions and views with global query rewrite permission.
- On UNIX, set the following parameters as global in the local .profile files:
  - ORACLE_BASE: the top-level directory where Oracle is installed.
  - ORACLE_HOME: the path to the Oracle root directory under ORACLE_BASE
  - LD_LIBRARY_PATH: $ORACLE_HOME/lib32:$ORACLE_HOME/lib

    If Oracle is a 64-bit installation, use lib32. Use SQL Plus to connect to the oracle database instance to determine if it is a 64-bit installation.
  - ORACLE_SID: the SID name used in the tnsnames.ora file.
  - JAVA_HOME: the path to the Java root directory. Business Objects installs a JDK in the following location:

    `report_server_home/jre`

    Note: JDK 1.5 is the minimum version supported for reports.

- PATH: $LD_LIBRARY_PATH:$JAVA_HOME:$JAVA_HOME/bin:$ORACLE_HOME/bin:$PATH
- LC_ALL: en_US.UTF-8

  Note: Be sure that the CASHCOMP environment variable is empty.

- On UNIX systems:
  - 3 GB of free space is required under /tmp.
  - You need access to a non-root user account to install the Report Server.

    This user should have a home directory in the local file system. For example, the following command creates a user with a local home directory:

    ```
    useradd -u 505 -g 0 -d /export/home/cabi -m cabi
    ```

    Also, add the non-root user to the install group and any group for which the root user is a member.
  - Enter the database server name in the /etc/hosts file if the database server is not on the same system as the Report Server. (If you have DNS, this step is unnecessary.)
  - If you encounter problems, inspect the SDK.log under these locations:

    /opt/CA/SharedComponents/CommonReporting3/ca-install.log

    /opt/CA/SharedComponents/CommonReporting3/CA_Business_Intelligence_InstallLog.log

Reporting Information

Record the following information you need during the Report Server installation:

Field Name	Description	Your Response
Administrator Password	Defines the password to log in to the Business Objects Infoview console.	
User Name	Identify the username for the Report Database.	
Password	Identify the administrative password credentials for the Report Database.	

Field Name	Description	Your Response
Pre-Installed Tomcat Information	Identify the path and port numbers for any previous installation of Tomcat. If you do not want to use a previous installation of Tomcat, Report Server installer can install Tomcat.	
Tomcat Port Numbers	The Tomcat connection, redirect, and shutdown ports. **Note:** If you install the Report Server on the same system as the CA Identity Manager, be sure that the Tomcat connection port does not conflict with the port number you specified for the application server URL when installing the CA Identity Manager.	

Open Ports for the Report Server

For CA Identity Manager and the Report Server to communicate successfully, the following ports must be opened.

- The Central Management Server (CMS) port: 6400
- The Report Server web application port:
  - JBoss/Tomcat: 8080
  - WebLogic: 7001
  - WebSphere: 9080

  Note the following:
  - This port is not the application server port for the CA Identity Manager Server.
  - The web server ports are provided during the Report Server installation. If you use different ports during the installation, those ports must be opened through the firewall when the Report Server is deployed in production.
  - The Report Server does not connect to the application server used by CA Identity Manager.
- All database ports that CA Identity Manager has configured for the reporting and auditing databases. The CA Identity Manager Server must send database information to the Report Server, so these ports must be opened. For example, if the Snapshot Database is an Oracle database, the Report Server needs the Oracle port open outbound.

Install the CA Report Server

You can install the Report Server on a supported Windows or UNIX system. The following sections detail how to install the Report Server using a Windows and UNIX installation wizard.

Important! For a production environment, install the Report Server on a separate system from the system with the CA Identity Manager Server. If you want to install the Report Server on the same system as the CA Identity Manager Server for demonstration purposes, do not choose the default tomcat ports 8080 and 1099 if JBoss is using those ports.

Note: CA Identity Manager supports CA Business Intelligence 3.2 (which is Business Objects XI 3.0 SP3).

Run the Windows Installer

Install the Report Server using the Windows installation wizard (Disk1\InstData\VM\Install.exe) found on the Report Server media.

Note: The Report Server is available for download on the CA Support site, under CA Identity Manager product downloads.

Follow these steps:

1. Exit all applications.
2. Download the Report Server and unzip it.
3. Navigate to Disk1\InstData\VM and double-click the installation executable.

   The installation wizard starts.
4. Use the gathered reporting information to install the Report Server.

   Note the following:

   - Select a New install during installation. Select SQL Anywhere, Oracle or SQL Server as the Report Database. If you must set non-default ports to avoid port conflicts, select a Custom install, but select SQL Anywhere, Oracle or SQL Server for the Report Database.
   - Select Tomcat as the web server, deselecting IIS.
   - If you are installing the Report Server on the same system as CA Identity Manager, select the Tomcat connection port carefully. Verify that it does not conflict with the port number you specified for the application server URL when installing CA Identity Manager. However, we recommend installing the Report Server on a different system than the CA Identity Manager Server in a production environment.

5. Review the installation settings and click Install.

   The Report Server is installed.

Run the UNIX Installer

Add execute permissions to the install file by running the following command:
`chmod+x /cabi-solaris-3_3_10/cabiinstall.sh`

Important! The installer may crash if executed across different subnets. To avoid this problem, install the Report Server directly on the host system.

Follow these steps:

1. Log in as the non-root user you created to install the Report Server.
2. Exit all applications.
3. Download the Report Server and untar it.

   Note: The Report Server is available for download on the CA Support site, under CA Identity Manager product downloads.
4. Open a command window and navigate to where the install program is located.
5. Enter the following command:

   `/cabi-solaris-3_3_10/cabiinstall.sh`
6. Use the gathered reporting information to install the Report Server. Note the following:

   Select a New install during installation. Select SQL Anywhere, Oracle or SQL Server as the Report Database. If you must set non-default ports to avoid port conflicts, select a Custom install, but select SQL Anywhere, Oracle or SQL Server as the Report Database.

   - Select Tomcat as the web server.
   - The installer installs the Report Server to /opt/CA/SharedComponents/CommonReporting3. Specifying another location does not change the installation location. So the /opt/CA directory must have non-root user permissions or the installation fails.
7. Review the installation settings and click Install.

   The Report Server is installed.

Run the Linux Installer

Follow these steps:

1. Install and start up an X-server on your client operation system.

   You can download X-Win32 from this location:

   `http://www.starnet.com/products/xwin32/download.php`

2. Log on to Linux by using the Business Objects installation account and run the following commands:

   ```
   bash$ export DISPLAY=$YOURXWin32ClientMACHINENAME:0.0
   bash$ echo &DISPLAY
   bash$ cd $INSTALLDIR/bobje/setup/
   bash$ source env.sh
   bash$ regedit
   ```

   where $INSTALLDIR is where the report server is installed.

3. Switch to the X-win32 client system.

   A Registry Editor message appears indicating that the configuration succeeded.

4. Create a registry category under the following HKEY_LOCAL_MACHINE location:

   HKEY_LOCAL_MACHINE\Software\Business Objects\Suite 12.0\Crystal Reports\DatabaseOptions

5. Add a key named MergeConnectionProperties under the DatabaseOptions category and set the value to Yes.

6. Add a key named MergeConnectionProperties under the following HKEY_CURRENT_USER location:

   HKEY_CURRENT_USER\Software\Business Objects\Suite 12.0\Crystal Reports\DatabaseOptions

7. Set the value for MergeConnectionProperties to Yes.

8. Refresh or schedule a report in Infoview to confirm the installation succeeded.

Run the Registry Script

For CA Identity Manager to change data sources for reports in the Report Server, run the mergeConnection script.

Note: On a 64-bit system, omit this procedure. The Report Server is a 32-bit application, so you use the 32-bit side of the registry. On a 64-bit system, open REGEDT32 directly from System32, and create the MergeConnectionProperties key with the Type REG_SZ and value Yes. Create the key in this location:

`@HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Business Objects\Suite 12.0\Crystal Reports`

On Windows, perform the following steps:

1. Copy the mergeConnection script from the system with the CA Identity Manager Admin toolkit to the Report Server. On the system with the toolkit, the default location for this script is as follows:

   C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\ReportServerTools

2. Run the mergeconnections_3.0.reg script and respond to the prompts that appear.
3. Click Start, Program Files, CA, Report Server, Central Configuration Manager.
4. Start all services, including Tomcat and the BO Server service.

On UNIX and Linux, perform the following steps:

1. Check for Windows control characters in the mergeconnections script.

   If you downloaded the software using FTP in binary mode, these characters do not appear in this script. If you used another download method, use the dos2unix command to remove these characters.

2. Copy the mergeconnections_3.0.cf script from the system with the CA Identity Manager Admin toolkit to the Report Server. On the system with the toolkit, the default location for this script is as follows:

   /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/ReportServerTools

   On the Report Server system, place the script in this location:

   installation-directory/bobje/enterprise120/generic

3. Source in the environment variables for BusinessObjects Enterprise, as follows:

   source *installation-directory*//bobje/setup/env.sh

4. Run the following script, as follows:

   ./configpatch.sh mergeconnections_3.0.cf

   Select 1 as the option when prompted.

   Note: On Linux systems, set the environment variable as follows before you run the script:

   export _POSIX2_VERSION=199209

5. Restart crystal processing servers as follows:

   a. Log in as the non root user you used to install the Report Server.

   b. Issue these commands:

   ```
   cd /opt/CA/SharedComponents/CommonReporting3/bobje
   ./stopservers
   ./startservers
   ```

Copy the JDBC JAR Files

Follow these steps:

1. Navigate to the jdbcdrivers folder where the CA Identity Manager Admin toolkit is installed. The default location is as follows:
   - Windows: C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\lib\jdbcdrivers
   - UNIX: /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/lib/jdbcdrivers
2. Copy ojdbc6.jar (for Oracle) or sqljdbc.jar (for SQL Server) to the following location:
   - Windows: CA\SC\CommonReporting3\common\4.0\java\lib
   - UNIX: /opt/CA/SharedComponents/CommonReporting3/bobje/java/lib

   Note: Copy sqljdbc.jar from Tools\lib\jdbcdrivers\1.2 to use the 1.2 driver that is compatible with the Report Server.
3. Open the CRConfig.xml file, found in the following location:
   - Windows: CA\SC\CommonReporting3\common\4.0\java
   - UNIX: /opt/CA/SharedComponents/CommonReporting3/bobje/java
4. Add the location of the JDBC JAR files to the Classpath. For example:
   - Windows: <Classpath>report_server_home\common\4.0\java\lib\sqljdbc.jar; report_server_home\ common\4.0\java\lib\ojdbc14.jar …</Classpath>
   - UNIX: <Classpath>${BOBJEDIR}/java/lib/sqljdbc.jar:${BOBJEDIR}/java/lib/ojdbc14.jar: …</Classpath>
5. Save the file.
6. Restart the Report Server as follows:
   - For Windows, do the following:
     a. Go to Start, Program Files, BusinessObjects XI *version*, BusinessObjects Enterprise, Central Configuration Manager.

        The Central Configuration Manager opens.

     b. Select all services and click Restart.
   - For UNIX, do the following:

     ```
     cd /opt/CA/SharedComponents/CommonReporting3/bobje
     ./stopservers
     ./startservers
     ```

Bypass the Proxy Server

If you are using a proxy server to channel outbound requests on the system where CA Identity Manager is installed, you need to bypass the proxy server. For details, see: Java Networking and Proxies.

Deploy Default Reports

CA Identity Manager comes with default reports you can use for reporting. BIConfig is a utility that uses a specific XML format to install these default reports for CA Identity Manager.

If you are upgrading from a previous version of the Report Server, first remove the CA Identity Manager Reports folder using the Central Management Console. The existing reports do not work. You can then deploy default reports for the new Report Server.

Important! This process updates all default reports. If you customized any default reports, be sure to back them up before performing the update.

Follow these steps:

1. Gather the following information about the Report Server:
   - Hostname
   - Administrator name
   - Administrator password
   - Snapshot database type
2. Copy all content from the Reports installer-root-directory/disk1/cabi/biconfig folder to the *im_admin_tools_dir*/ReportServerTools folder.
3. Set the JAVA_HOME variable to the 32-bit version of the JDK1.5 you installed.

4. Run one of the following commands:

   - For a Microsoft SQL Snapshot Database:

     ```
     biconfig -h "hostname" -u "administrator_name" -p "administrator_password"
     -f "ms-sql-biar.xml"
     ```

   - For an Oracle Snapshot Database:

     ```
     biconfig -h "hostname" -u "administrator_name" -p "administrator_password"
     -f "oracle-biar.xml"
     ```

   Note: In a UNIX operating environment, be sure that biconfig.sh has execute permissions.

5. View the biconfig.log file found in the location where you ran the command in Step 4.

6. Verify that the default reports installed successfully. Inspect the end of the log file for status; a successful install appears as follows:

   ```
   ReportingDeployUtility - Reporting utility program terminated and return code =
   0
   ```

BusinessObjects XI 3.x Post-Installation Step

If you run report tasks and receive a "Server Input% not found or server may be down" error message, perform this procedure.

Follow these steps:

1. Log in to the Central Management Console using the username and password you entered during the Report Server installation.
2. Under the main dashboard, select Servers.
3. Under the Server Name column, search for Input File Repository server and double-click the name.
4. In the Server Name text box, enter the following:

   Input.*report_server_hostname*.InputFileRepository

5. Click Save.
6. Under the Server Name column, search for Output File Repository server and double-click the name.
7. In the Server Name text box, enter the following:

   Output.*report_server_hostname*.OutputFileRepository

8. Click Save.
9. Restart *all* the servers by selecting the servers in the Server List.

How to Secure the CA Identity Manager and Report Server Connection for JBoss/WebLogic

CA Identity Manager and Report Server communicate over a non-secure connection.. Secure Sockets Layer (SSL) connection can be used to secure the connection between Report Server and CA Identity Manager.

An SSL connection ensures that the communication is encrypted when data is accessed from the Report Server. Before configuring the SSL, verify that the BO (Business Objects) Server has HTTPS enabled. To secure the connection with SSL, self-signed certificate or the certificate from the Certified Authority (CA) can be used.

To configure an SSL certificate using self-signed certificate, perform the following steps:

1. Export the certificate from the keystore used in the BO Server, using any tool which generates a certificate.
2. Copy the certificate to a directory where the CA Identity Manager is installed.
3. Import the Certificate in to the Java trust store (cacerts). Also, verify that the certificate is imported in to the java version which is currently used by CA Identity Manager server.
4. Restart the Application Server for the changes to take effect.
5. In CA Identity Manager, go to System, Reporting, Report Server Connection. Select the Secure Connection option.
6. Click Test Connection to verify the connectivity.

The following procedure is an example on how to export and import a certificate using the Keytool utility.

Follow these steps:

1. In the BOServer, open the command prompt and enter the following command to export the certificate from the keystore:

   Windows:

   ```
   ..\jvm\bin\keytool -export -alias testcert -file certificate.cer -keystore
   c:\cert\.keystore  –storepass <keystore password>
   ```

   Linux or Solaris:

   ```
   ../jvm/bin/keytool -export -alias testcert -file certificate.cer -keystore
   /root/.keystore  –storepass <keystore password>
   ```

2. Copy the certificate to a directory where the CA Identity Manager is installed.
3. In the CA Identity Manager server, open the command prompt and enter the following command to import the certificate into the keystore:

   Windows:

```
..\jvm\bin\Keytool -import - trustcacerts –file c:\cert\certificater.cer –alias
testcert -keystore JAVA_HOME\jre\lib\security\cacerts -storepass password
```

Linux or Solaris:

```
../jvm/bin/Keytool -import - trustcacerts –file /root/certificater.cer –alias
testcert -keystore JAVA_HOME/jre/lib/security/cacerts -storepass password
```

The certificate is successfully installed.

Note: We recommend that you refer the vendor-specific documentation to configure SSL on the Report Server. The Report Server supports Tomcat and IIS servers.

Verify the Reporting Installation

To verify that reporting has been installed correctly, do the following:

- In the Central Management Console, be sure that all services are running.
- Be sure that your Report Database is running.

Note: For more information on configuring reporting after the installation, see the *Administration Guide*.

Silent Installation

For more information about silent installation of the Report Server, see the *CA Business Intelligence Installation Guide.* The Report Server documentation is available in one of the following locations when you extract the Report Server installer files:

- **Windows**: *install_root_directory*\Docs\CABI_Impl_ENU.pdf
- **UNIX:** *install_root_directory/*Docs/ENU/CABI_Impl_ENU.pdf

How to Uninstall Reporting

You uninstall the Report Server when it is no longer required on the system. For more information, see the CA Business Intelligence documentation.

After uninstalling the Report Server, Remove Leftover Items (see page 77).

Remove Leftover Items

The following sections detail the items you must manually remove after uninstalling the Report Server to keep the system as clean as possible and to prevent a reinstallation of the Report Server to the same system from failing.

Remove Windows Items

Follow these steps:

1. Navigate to *report_server_home*.

   report_server_home specifies the Report Server installation path.
2. Open the BusinessObjects Enterprise 12 folder, and delete the following folders:
   - Data
   - java
   - Logging
   - Samples
   - Web Content
   - Web Services
   - win32x86
3. Return to the Report Server folder.
4. Open the common folder.
5. Open the 4.0 folder, and delete the following folders:
   - crystalreportviewers115
   - java

You have completed removing leftover items.

Remove UNIX Items

Following are the steps to remove leftover Report Server items on UNIX:

Follow these steps:

1. Navigate to the following location from a command prompt:

   /opt/CA/SharedComponents
2. Delete the CommonReporting3 folder.

You have completed removing leftover items.

Chapter 7: High Availability Provisioning Installation

Based on the guidelines in this chapter, you implement high availability for provisioning components by installing alternate Provisioning Servers and Provisioning Directories, and connector servers for C++ and Java connectors.

This section contains the following topics:

Installation Status

The following table shows you where you are in the installation process:

You Are Here	Step in Installation Process
	1. Install prerequisite hardware and software and configure your system as required.
	2. Perform one of these installations: ■ Single node installation ■ Installation on an application server cluster
	3. (Optional) Create separate databases.
	4. (Optional) Install the Report Server.
X	**5. (Optional) Install alternate Provisioning Directories, alternate Provisioning Servers, and connector servers to support failover and load balancing.**

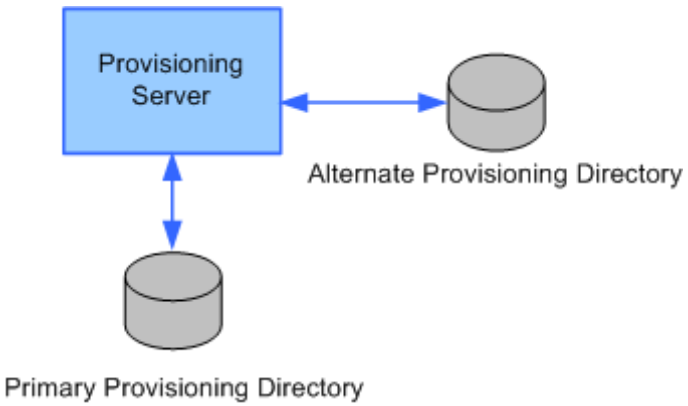
How to Install High Availability Provisioning Components

The following table describes the steps involved in installing provisioning components for high availability:

✓	Step
	1. Install primary and alternate Provisioning Servers and provisioning directories for load balancing and failover.
	2. Install several connector servers for load balancing and failover.
	3. Enable clients of the provisioning server to fail over.

Redundant Provisioning Directories

To support failover, you can install primary and alternate Provisioning Directories. For example, you may have two systems, one with the Provisioning Server and the primary Provisioning Directory on it. The second system has the alternate Provisioning Directory. If the primary Provisioning Directory fails, the alternate Provisioning Directory is assigned automatically.



Follow these steps:

1. Install the primary Provisioning Directory using the Provisioning Directory installer from where you unpacked the install package.
   - **Windows:**
     Unpacked-Install-Package\Provisioning\Provisioning Directory\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ProvisioningDirectory/setup
2. Install one or more alternate Provisioning Directories. See the next section.

Install Alternate Provisioning Directories

Once you have performed the prerequisite configuration required, you can install alternate Provisioning Directories.

Follow these steps:

1. Log in as a Local Administrator (for Windows) or root (for Solaris) into the system where you plan to install the alternate Provisioning Directory.
2. Be sure that CA Directory is installed on this system.
3. Copy custom schema files to the %DXHOME%/config/schema directory if any of the following is true for the primary Provisioning Directory:
   - COSX (etrust_cosx.dxc) has been modified
   - LDA connector (etrust_lda.dxc) is installed
   - A custom C++ connector schema has been created

   The Provisioning Directory installation checks the %DXHOME%/config/schema directory for extra schema files named etrust_*.dxc, and adds them to the group schema file, impd.dxg. If the custom schema files are not copied locally, data replication between the Provisioning Directories fails.
4. Run the Provisioning Directory installer from where you unpacked the install package.
   - **Windows:**
     Unpacked-Install-Package\Provisioning\Provisioning Directory\setup.exe
   - **UNIX:**
     Unpacked-Install-Package/Provisioning/ProvisioningDirectory/setup
5. Select High Availability, and respond to the questions about the hostnames for systems where other Provisioning Directories are installed and which system is the primary Provisioning Directory.

6. Respond to other questions using the same answers given during the primary Provisioning Directory installation for:
   - Deployment Size
   - Shared Secret
   - FIPS key
7. Respond to this question based on how and when you want to replicate data from the Primary Provisioning Directory:

   `Do you want to start replication to the Provisioning Directory.`

   If you are upgrading from a previous release, you may have a significant amount of data to replicate. You should deselect the checkbox if you do not want replication to start at this time. After the installation, you would then need to copy an LDIF data dump or online backup files from an existing Provisioning Directory and load the data or start the DSAs manually, which will start automatic replication.

Important! If alternate Provisioning Directory installation failed, data replication may have occurred before the failure. If so, the master and alternate Provisioning Directories have a record that replication occurred. If you now reinstall the alternate Provisioning Directory, that data is not replicated again. Instead, use the High Availability Configuration command on the primary and alternate Provisioning Directories to remove and add back the alternate Provisioning Directory before you reinstall it.

Reconfiguring Systems with Provisioning Directories

If needed, you can change the configuration of which systems have a Provisioning Directory.

Follow these steps:

1. Log into the system where the primary Provisioning Directory is installed.
2. On a command line prompt, navigate to the highavailability sub-directory where you installed the Provisioning Directory. For example:

   ```
   cd C:\\Program Files\CA\Identity Manager\Provisioning Directory\highavailability
   ```

3. Enter this command:

   ```
   highavailability.bat
   ```

   The command displays a summary of the current configuration: the domain name, the hostname of each Provisioning Server and Provisioning Directory, and which one is the Primary Provisioning Directory.

4. Respond to prompts for the hostnames for each alternate Provisioning Directory that you plan to add.

   If you plan to install alternate Provisioning Servers, you can add their hostnames now by responding to the prompts.

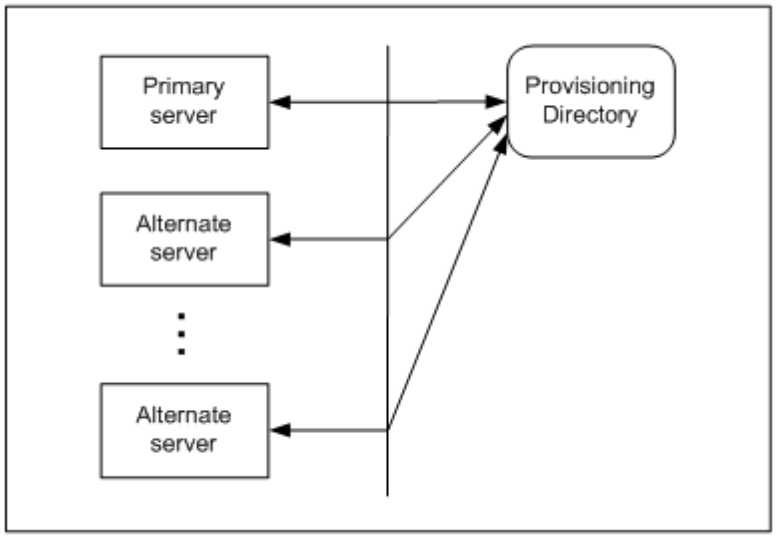
5. Log in to all other Provisioning Directory and Provisioning Servers and repeat steps 2 through 4.

   The configuration on each system needs to match.

Redundant Provisioning Servers

Multiple Provisioning Servers share the workload of a provisioning domain, providing performance, scalability, and high availability. The first Provisioning Server installed is called the primary Provisioning Server. Additional servers are called alternate Provisioning Servers.

As shown in this illustration, you can configure multiple alternate Provisioning Servers for one primary Provisioning Server.



In this illustration, three Provisioning Servers are configured to serve the provisioning domain. All servers are configured to use the Provisioning Directory of the primary Provisioning Server installation.

Router DSA for the Provisioning Server

The Provisioning Server communicates through a CA Directory router DSA, and not directly to the Provisioning Directory. The router DSA, imps-router, is installed with the Provisioning Server installer. This DSA accepts requests from the Provisioning Server and routes them to the appropriate Provisioning Directory DSA (impd-co, impd-main, impd-inc, or impd-notify) depending on the prefix.

In a high-availability installation, the imps-router DSA has connection information for Provisioning Directory DSA on at least one alternate Provisioning Directory system. If a primary Provisioning Directory DSA becomes unavailable, the router DSA attempts to use an alternate DSA.

The imps-router DSA has been assigned ports 20391, 20391, 20393 (for address, SNMP, and console respectively).

Note: In previous releases of this software, the etrustadmin DSA used port 20391. Any connections to 20391 on the Provisioning Directory system fail unless the Provisioning Directory and Provisioning Server are on the same system. Therefore, reroute these connections to port 20391 on the Provisioning Server system.

For CA Directory DSAs running on one system to communicate with DSAs on another system, they must have connection information for each other. So during Provisioning Directory installation, you identify each Provisioning Server that can connect to it.

Install Provisioning Servers

To support failover, you can install primary and alternate Provisioning Servers.

Follow these steps:

1. Install the primary Provisioning Server using the Provisioning Server installer from where you unpacked the install package.
   - **Windows**:
     Unpacked-Install-Package\Provisioning\Provisioning Server\setup.exe
   - **UNIX or Linux**:
     Unpacked-Install-Package/Provisioning/ProvisioningServer/setup
2. Install one or more alternate Provisioning Servers. See the next section.
3. Enter the alternate Provisioning Server host and port number when you enable provisioning in the CA Identity Manager Management Console. For details, see the *Configuration Guide*.

Install Alternate Provisioning Servers

Once you have performed the prerequisite configuration involving the highavailability command, you can install one or more Provisioning Servers.

Follow these steps:

1. Log in as a Local Administrator (for Windows) or root (for Solaris) on each system that will host an alternate Provisioning Server.
2. Make sure that CA Directory is installed on this system.
3. Copy custom schema files to the %DXHOME%/config/schema directory if any of the following is true for the primary Provisioning Directory:
   - COSX (etrust_cosx.dxc) has been modified
   - LDA connector (etrust_lda.dxc) is installed
   - A custom C++ connector schema has been created

   The Provisioning Directory installation checks the %DXHOME%/config/schema directory for extra schema files named etrust_*.dxc, and adds them to the group schema file, impd.dxg. If the custom schema files are not copied locally, the Provisioning Server will not route any custom schema.
4. Run the Provisioning Server installer from where you unpacked the install package.
   - **Windows**:
     Unpacked-Install-Package\Provisioning\Provisioning Server\setup.exe
   - **UNIX**:
     Unpacked-Install-Package/Provisioning/ProvisioningServer/setup
5. Complete the instructions in the installer dialog boxes.

   You can select a check box during installation to configure Provisioning Directory high availability. If you choose this option, you must supply the hostnames of any alternate Provisioning Directories and specify the primary Provisioning Directory.

Reconfiguring Systems with Provisioning Servers

If needed, you can change the configuration of which systems have a Provisioning Server.

Follow these steps:

1. Log into the system where the primary Provisioning Directory is installed.
2. On a command line prompt, navigate to the location where you installed the Provisioning Directory or Provisioning Server. You find the highavailability sub-directory there. For example:

   ```
   cd C:\\Program Files\CA\Identity Manager\Provisioning
   Directory\highavailability
   ```

3. Enter this command:

   `highavailability.bat`

   The command displays a summary of the current configuration: the domain name, and the hostname of each Provisioning Server and Provisioning Directory.

4. Respond to the prompts to provide the hostnames required for each Provisioning Server that you want to add.

   If you plan to also install alternate Provisioning Directories, you can add their hostnames now by responding to the command prompts.

5. Log in to each system that will host a Provisioning Directory and repeat Steps 2 through 4.

   The configuration on each system needs to match.

Configure Provisioning Server Failover

For CA Identity Manager to distinguish the primary from the alternate Provisioning Server, you create server definitions in JIAM in the Management Console. You create these definitions in the directory object associated with the CA Identity Manager directory for your environment. During initialization, CA Identity Manager reads any failover server definitions defined in that object, adding them to the JIAM failover server definitions.

Note: For details on setting up server definitions, see the *Configuration Guide*.

Redundant Connector Servers

With the Connector Server Framework (CSF), you can run multiple Connector Servers and configure the Provisioning Servers to communicate with Connector Servers in specific contexts.

As a result, the Provisioning Server can:

- Support Connector Servers on different platforms to manage endpoint types that are unavailable on the platform where the Provisioning Server is installed.
- Communicate with multiple Connector Servers, which each manage a different set of endpoint types or endpoints. Therefore, endpoint types or endpoints can be managed on a parallel basis to achieve load balancing.

Connector Server Framework

The use of several Connector Servers is called the Connector Server Framework. The Connector Server Framework has two important characteristics:

- Scalability - multiple connector servers may share the load of working on a set of endpoints.

  For example, a lengthy exploration of an endpoint on one connector server does not influence the ability to operate on an endpoint that is being controlled by another Connector Server

- Communication channel security - communication between Provisioning Server and connector server is encrypted using TLS.

  If an endpoint type uses a proprietary protocol to communicate between the connector server and endpoints of that protocol, the extent of use of the proprietary protocol may be limited to a local network, or even to just local communication inside one server.

When deciding on an implementation strategy, consider these factors so that you protect the Connector Servers in your organization against unauthorized access:

- The Connector Server may be configured to disclose passwords in clear text.

  Any person with access to the system running the Connector Server and with sufficient privileges to modify the configuration of the Connector Server and to restart the Connector Server can make the Connector Server log passwords appear in clear text.

  The Connector Server is based on the open source slapd process. The instructions to make a slapd process log incoming passwords in clear text are in the public domain, for example, by looking at the manual pages at http://www.openldap.org

- The Connector Server is only protected by a bind password.

  The Connector Server trusts any client who connects to it and is able to provide the proper credentials, such as Bind DN and Bind Password. The Connector Server does not know if the connection comes from a Provisioning Server or not. Any user with internal access may disclose the bind password, then connect to the Connector Server from another server, and so have administrator privileges over the endpoints controlled by the Connector Server.

- The Connector Server is not protected against brute force attacks on the bind password

  Unlike the Provisioning Server, the Connector Server is not protected against repeated attempts at binding with different passwords. An attacker may therefore try to guess the password by brute force attack. Should an attacker succeed in guessing the bind password, then the road is open for the attacker to control the endpoints under control of the Connector Server.

For these reasons you are advised to design your implementation such that

- The same organizational unit is responsible for administrative access to all Provisioning Servers and connector servers.
- Your connector servers are suitably protected by firewalls or similar such that the ports may not be reached by unauthorized means.
- The ability to connect to Provisioning Servers and connector servers on non-TLS ports should be disabled in your production environments.

If you install multiple connector servers on one computer, make sure that each instance uses a unique set of port numbers.

Load-Balancing and Failover

Failover and load-balancing of connector requests is achieved by each provisioning server based on the CSF configuration defined using csfconfig or Connector Xpress.

Each provisioning server consults the CSF configuration that applies to it and determines which Connector Servers it should use to access each endpoint or endpoint type. Failover and load-balancing occur when there are multiple connectors servers configured to serve the same endpoint or endpoint type.

Failover and load-balancing are unified and cannot be controlled separately. One cannot indicate that a particular connector server is to remain idle except when needed for failover. Instead, a provisioning server that is configured to use two or more connector servers interchangeably will distribute work between these connector servers (load balancing) during normal operation. Should one or more of the Connector Server become unavailable, the remaining connector servers will provide failover support for the unavailable connector servers.

Reliability and Scalability

With the Connector Server Framework (CSF), the Connector Server high availability features increase reliability and scalability.

Reliability is enhanced by having multiple Connector Servers serve a Provisioning Server, so it can continue to function if one or more Connector Servers become unavailable.

For example, if one Connector Server manages the UNIX endpoint type and another manages the Active Directory endpoint type; and the Active Directory Connector Server becomes unavailable, the Provisioning Server can still manage the UNIX endpoint types.

Scalability is achieved by having a mechanism to add more Connector Servers to manage an increasing amount of endpoint types or endpoints. For example, if the number of endpoint types increases to 100, the Provisioning Server can be configured to have 20 Connector Servers, with each Connector Server managing five endpoint types. Or configure 20 Connector Servers with each Connector Server managing overlapping sets of 10 endpoint types to allow for failover and load balancing behaviors as well.

Multi-Platform Installations

The Connector Server Framework is the configuration of Connector Servers that exist on multiple systems, which could be Windows or Solaris systems.

The following use cases are supported:

- Use Case 1
  - Provisioning Server and connector server installed on a Solaris system.
  - A second Connector Server installed on a Windows system, serving the non-multi-platform connectors.
- Use Case 2
  - Provisioning Server and connector server installed on a Windows system.
  - A second Connector Server installed on Solaris system, serving the multi-platform connectors.
  - A third Connector Server installed on a remote Windows system, serving the other connectors.

- Use Case 3
  - Provisioning Server installed on a Windows or Solaris system and a Connector Server installed on the same system.
  - Multiple additional Connector Servers installed on Windows or Solaris systems, serving as endpoint agents. This scenario is important for cases where the connector is using a proprietary or un-secured communication channel. Using this topology, the important segment of network traffic is secured by the standard Provisioning Server to Connector Server communication protocol and not by the proprietary protocol.

Install the C++ Connector Server

When you install Java CS, you can install the C++ Connector Server (CCS). The procedure in this topic applies for a single connector server. If you plan to install more than one CCS, see the chapter on High Availability Provisioning Installation.

Follow these steps:

1. Run the following program where you unpacked the install package.
   - **Windows:**
     `Provisioning\Provisioning Server\setup.exe`
   - **UNIX:**
     `Provisioning/ProvisioningServer\setup.bin`
2. Complete the instructions in the installer dialog boxes.

   This installation program also gives you the option to install alternate Provisioning Servers. However, for that component, a different procedure applies.

Configure Connector Servers

You configure the Connector Server Framework by using the csfconfig command or by using Connector Xpress. The csfconfig command uses the data in the Windows Registry (or UNIX counterpart created for Provisioning Server) to connect to a Provisioning Server. The csfconfig command must run on the system where one of the Provisioning Server runs.

Using the command, you can:

- Add or modify a Connector Server connection object with information such as the connector server, host, and port.
- Define for which endpoints or endpoint types the connector server is used; possibly varying this definition for alternate provisioning servers.
- Delete the Connector Server connection information object.

- List all connector server connection objects in a domain.
- Show one or all connector server connection objects for one or all connector servers

The csfconfig command uses the authorizations provided by a global user credential, so that global user must have the necessary administrative privileges to manipulate the appropriate ConfigParam and ConfigParamContainer objects.

csfconfig Command

To use the csfconfig command, the command line syntax is:

```
csfconfig [--help[=op]] [operation] [argument]
```

You can use these arguments in any order. The operation argument is required unless you are using the --help argument.

The --help[=op] option provides minimal on-line help. The "=op" argument may be used to list the arguments that are required or optional for the operation. For example, "--help=add" will provide a description of the add operation, while "--help" will provide general information.

If help is requested, other arguments are ignored and no request is sent to the server.

Note: The domain parameter can be omitted as it is always the domain used in the whole installation.

The following operations are available.

add

Add a new CS connection object. A name will be generated by this operation if one is not specified by the user. Required arguments: auth, host, pass. Optional arguments: authpwd, br-add, desc, domain, name, port, usetls, debug.

addspec

Adds a branches specialization for one provisioning server.

When you have installed alternative provisioning servers, sometimes a connector server is not to be used by all of these Provisioning Servers. Or sometimes different provisioning servers will want to use the same connector servers for different branches (endpoint types or endpoints). A branches specialization is a list of branches that is specific to one provisioning server. Only provisioning servers without a specialization will use the branches specified in the main CS connection object. Required arguments: auth, name, server. Optional arguments: authpwd, br-add, domain, debug.

list

List all CS connection objects. Required arguments: auth. Optional arguments: authpwd, domain, debug.

modify

Modify a CS connection object. Required arguments: auth, name. Optional arguments: authpwd, br-add, br-rem, desc, domain, host, pass, port, usetls, debug.

modspec

Edits a specialization created by addspec. Required arguments: auth, name, server. Optional arguments: authpwd, br-add, br-rem, domain, debug.

remove

Remove an existing CS connection object. Required arguments: auth, name. Optional argument: authpwd, debug.

remspec

Removes a specialization created by addspec. Required arguments: auth, name, server. Optional arguments: authpwd, domain, debug.

modify

Modify a CS connection object. Required arguments: auth, name. Optional arguments: authpwd, br-add, br-rem, desc, domain, host, pass, port, server, tls, usetls.

show

Show a specific CS connection object or show all CS connection objects. The output shows the host and port of the connector server if it is available. Required arguments: auth. Optional arguments: authpwd, name, domain, debug.

Each operation takes several arguments in the form “name=value”. Spaces are not allowed before or after the “=” symbol, and if the value contains any spaces, the argument must be quoted appropriately for the platform (Windows or UNIX). Except as noted, the value must be provided, and must be non-empty.

The following arguments are used for the operations as noted above:

auth=<value>

Identify the global user for authentication.

Value format: “name” where name is the global user's name.

authpwd=<value>

Identify a file containing just the global user's password on the first line. If this argument is not specified, the user will be prompted for a password.

Value format: any appropriate operating system file path.

br-add=<value>

Add a new branch. This argument may be specified multiple times to add multiple branches.

Value format: “[[endpoint,]endpoint type][@[domain]]”. Use a branch of “@” by itself to represent all branches. Add “endpoint type” or “endpoint,endpoint type” to identify a specific endpoint type or endpoint.

br-rem=<value>

Remove an existing branch. This argument may be specified multiple times to remove multiple branches.

Value format: same format as specified for br-add.

debug=<value>

Turns on trace logging for the command. Tracing messages are written to the file PSHOME\logs\etaclientYYYYMMDD.log file.

Value format: The value “yes" enables logging.

desc=<value>

Provide an arbitrary description for the object. If not specified in an add operation, it will default to the value of the host argument.

Value format: an arbitrary string.

domain=<value>

Define the default domain. If not specified, the domain specified in the auth argument is used as the default.

As the value can only be the default, this parameter can always be omitted

host=<value>

Define the name of the host on which the Connector Server runs.

Value format: any legal host name or IP address.

name=<value>

The name of the Connector Server object. If not specified during Add, csfconfig will assign a name and display what name was created.

Value format: A case-insensitive string of one or more characters consisting of upper-case English letters (A-Z), lower-case English letters (a-z), digits (0-9), hyphen(-) or underscore(_).

pass[=<value>]

Define the file containing the password for the Connector Server connection object. If the value is not specified, the user will be prompted.

Value format: any appropriate OS file path.

Important! The password you must specify is the password you entered when you installed that Connector Server or you changed subsequent to install by running the pwdmgr utility on that Connector Server system.

port=<value>

Define the port number for the object. This must be a valid number for a port where the Connector Server listens for connections.

Value format: an integer.

server[=<value>]

In addspec, modspec and remspec commands, define the name of the Provisioning Server that is served by the Connector Server . The branches defined in a specialization override, for a particular provisioning server, the branches defined in the CS configuration object by add and modify commands.

Value format: the name of the host where the Provisioning Server is running as returned by the system's hostname command.

Note: The Connector Server configuration objects are stored with the other domain configuration parameters in the provisioning directory. While the Connector Server configuration parameters cannot be viewed or changed with the provisioning manager directly, one can use the provisioning manager (System task, Domain Configuration button) to get a list of known provisioning servers. To do this, open the “Servers” parameter folder and the known provisioning servers will be listed.

usetls[=<value>]

Indicate if TLS should be used to communicate with the Connector Server. The value is optional for the add operation only, in which case it defaults to “yes.” .

Value format: a string “yes” or “no”.

Upon successful completion of the add operation, the name of the newly created Connector Server connection object will be listed. If the name parameter is missing, a name is generated. For example:

```
Created CS object with name = SA000
```

For most operations, successful or not, the status and a message (if any) will be shown. For example:

```
The host name, port number, or TLS flag was successfully changed. The branch settings
were successfully changed.
```

For some errors, such as invalid command line parameters, no status code or server error message is displayed. In these cases, a simple statement of the error will be shown. For example:

```
$ csfconfig add
No authentication information supplied.
For on-line help, use "--help [=<op>]"
```

csfconfig Command Examples

To specify that the UNIX and CA Access Control endpoint types should be served by the Connector Server running on host "sunserver01" and the remaining endpoint types served by a Connector Server running on host "windows02", issue the following commands.

Each command execution prompts you for the etaadmin password.

```
csfconfig add \
auth=”etaadmin” \
br-add=”UNIX – etc” \
br-add=”UNIX – NIS-NIS plus Domains” \
br-add=”Access Control” \
host=”sunserver01” \
usetls=”yes”

csfconfig add \
auth=”etaadmin” \
br-add=”@” \
host=”windows02” \
usetls=”yes”
```

C++ Connector Server on Solaris

The C++ Connector Server installed on Solaris can manage only Solaris UNIX ETC and ACC endpoints. For all other Connectors, install the C++ Connector Server on a Windows system and register it with the Provisioning Server installed on Solaris. During installation, specify that this Connector Server is your default C++ Connector Server.

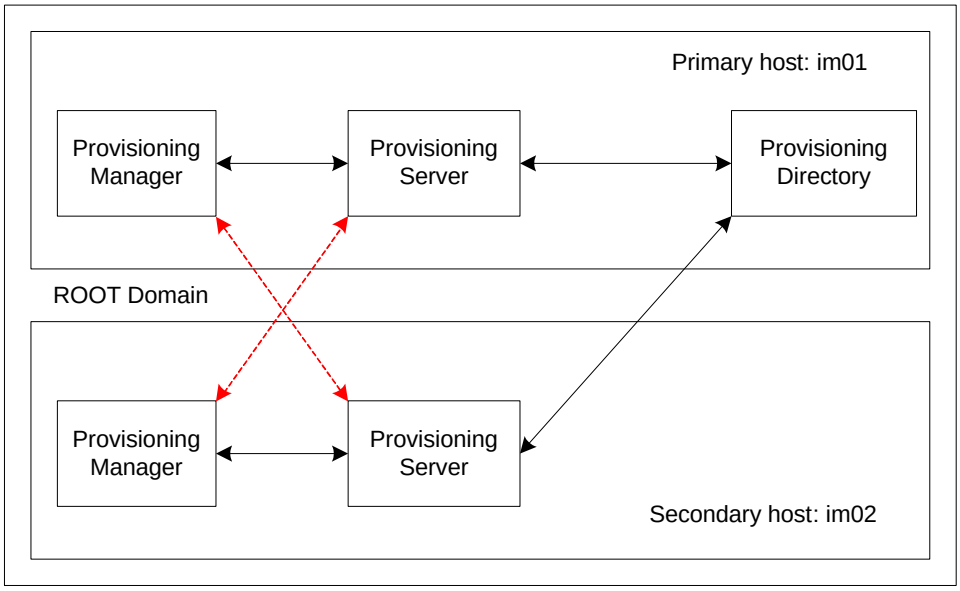
Failover for Provisioning Clients

Client-tier configuration includes the following tasks:

- Configure the Windows client-tier failover
- Configure the Provisioning Manager to communicate with their local Provisioning Servers, and fail over to the remote Provisioning Server

You use the same Provisioning Manager dialog to accomplish both of these tasks, on each server in turn.

The configuration shown in the following illustration lets Provisioning Manager submit identity provisioning requests to one Provisioning Server and fail over to another server:



The Provisioning Manager sends requests to the default Provisioning Server and fails over to another server.

Enable User Console Failover

If the application server for the CA Identity Manager Server fails, it does not receive Provisioning Server updates. As a result, the CA Identity Manager User Console does not show provisioning changes. Therefore, you should configure an alternate URL for the CA Identity Manager Server.

Follow these steps:

1. Launch the Provisioning Manager.
2. Click System, CA Identity Manager Setup.
3. Fill in the host name and port for another system in the cluster.
4. Fill in the environment.

   It must be the same one that is on the primary URL.
5. Click Add.

Enable Provisioning Manager Failover

You can enable Provisioning Manager failover on both the primary and secondary host servers. When this procedure is complete, you will have configured each server for failover to the other.

Follow these steps:

1. Launch the Provisioning Manager.
2. Select File, Preferences, and select the Failover tab.
3. Mark the Enable Failover check box. By default, the local Provisioning Server is already defined.
4. Click Add.
5. Enter the host name of the remote Provisioning Server.

   For example, on im01, enter the server host for im02. On im02, enter the server host for im01.
6. Enter 20389 for the LDAP port value and 20390 for the LDAP/TLS port value, respectively.
7. Adjust the preference order by moving the entries up and down in the list.
8. Click OK.
9. Restart the Provisioning Manager to enable your changes.

Test the Provisioning Manager Failover

You can test your client failover configuration by performing the following procedure:

Follow these steps:

1. Stop the CA Identity Manager - Provisioning Server service on one domain server.
2. Issue one or more operations using Provisioning Manager for this server installation.

   Since you stopped the CA Identity Manager - Provisioning Server service locally, the traffic flows to the failover domain server. If it does not, check your configuration and try the test again.

Appendix A: Uninstallation and Reinstallation

This section contains the following topics:

How to Uninstall CA Identity Manager

To fully uninstall CA Identity Manager, remove CA Identity Manager software components and clean up the CA Identity Manager-specific configuration in your application server. The following checklist describes the steps to uninstall CA Identity Manager:

✓	Step
	1. Delete CA Identity Manager objects with the Management Console.
	2. (Optional) If you used SiteMinder, remove the CA Identity Manager schema from the policy store or remove the Policy Server. For more information, see the *CA SiteMinder Web Access Manager Policy Server Installation Guide*.
	3. Use the highavailability command to uninstall Provisioning Directories and Provisioning Servers from this location: *Unpacked-Install-Package*\Provisioning\Provisioning Directory\highavailability
	4. Uninstall the CA Identity Manager components.
	5. Remove CA Identity Manager configuration information from the application server.

Remove CA Identity Manager Objects with the Management Console

In order to remove objects created automatically by CA Identity Manager when you configure environments and directories, use the Management Console.

1. Open the Management Console:

   http://*im_server*:*port*/iam/immanage
2. Click Environments.
3. Select all of the check boxes for the existing Environments.
4. Click Delete.
5. Click Directories.
6. Select all of the check boxes for the existing Directories.
7. Click Delete.

Remove the CA Identity Manager Schema from the Policy Store

If you were using a SiteMinder Policy Server, remove the CA Identity Manager schema from the policy store.

Remove the CA Identity Manager schema from a SQL Policy Store

On systems where you installed the CA Identity Manager Extensions for SiteMinder, remove the CA Identity Manager schema. The default location for the command to remove the schema follows:

- SQL Server:

  `C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas\MicrosoftSQLServer`
- Oracle:

  UNIX:

  `/opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/policystore-schemas/OracleRDBMS`

  Windows: `C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager/tools/policystore-schemas/OracleRDBMS`

Remove the CA Identity Manager schema from an LDAP Policy Store

Note: If you are using Microsoft Active Directory or Microsoft ADAM as a policy store, you do not need to complete this procedure. You cannot remove schema objects from these policy stores. However, you can disable them. For more information, see the documentation for your directory.

Follow these steps:

1. Complete one of the following:
   - If you are using IBM Directory Server as a policy store, in the IBM Directory Server Web Administration user interface, remove the schema file V3.imsschema60 from the Files section of the schema configuration. Then, restart the directory server.

     Note: There are no other steps required to remove the schema from an IBM Directory Server. Continue with Uninstall CA Identity Manager Software Components.
   - If you are using CA Directory as a policy store, remove the etrust_ims.dxc file from *dxserver_home*\config\schema.

     where *dxserver_home* is the install location of CA Directory.

     Note: There are no other steps required to remove the schema from a CA Directory Server. Continue with Uninstall CA Identity Manager Software Components.
   - If you are using another LDAP directory as a policy store, skip to Step 2.
2. Navigate to the policystore-schemas folder. These are the default locations:
   - **Windows:** C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\tools\policystore-schemas
   - **UNIX:** /opt/CA/IdentityManager/IAM_Suite/Identity_Manager/tools/policystore-schemas
3. Use the appropriate LDIF schema file from the following table to remove the schema from the directory.

   Note: For more information on removing schema files, see the documentation for your directory.

Directory Type	LDIF File
Novell eDirectory	novell\novell-delete-ims8.ldif
Oracle Internet Directory (OID)	oracle-internet-directory\oracle-internet-directory-delete-ims8.ldif

Directory Type	LDIF File
Sun Java Systems (Sun One, iPlanet)	sunone\sunone-delete-ims8.ldif

Uninstall CA Identity Manager Software Components

Use the instructions in this section to uninstall CA Identity Manager components from each system on which you installed a component. For example, if you installed the CA Identity Manager Server and the CA Identity Manager Administrative Tools on separate systems, uninstall components from both systems.

For Windows:

1. Go to Start, Control Panel, Add/Remove Programs and select CA Identity Manager.
2. Select CA Identity Manager.
3. Click Change/Remove.

   All non-provisioning components will be uninstalled.
4. For any provisioning components, use the individual component installer to uninstall the component.

For UNIX:

1. Navigate to the following location:

   IM_HOME/install_config_info/im-uninstall
2. Run the following script:
   `sh uninstall.sh`

   Follow the on-screen instructions.
3. For any provisioning components, use the individual component installer to uninstall the component.

Remove CA Identity Manager from JBoss

After you uninstall CA Identity Manager, there are no additional steps required in the JBoss application server.

To remove the JBoss application server, delete the directory where you installed JBoss.

Reinstall CA Identity Manager

You can reinstall any of the CA Identity Manager software components by rerunning the installer. When you run the installer, it detects any CA Identity Manager components installed on the system. You may reinstall the same components that you originally installed on the system or other components that were not originally on the system.

Note: Reinstalling the CA Identity Manager Administrative Tools replaces all of the files in the Administrative Tools directory. To prevent overwriting custom files, back up the directory where the Administrative Tools are installed.

Appendix B: UNIX, Linux, and Non-Provisioning Installations

For UNIX and LINUX systems and scenarios where no provisioning software is needed, some additional instructions apply.

This section contains the following topics:

UNIX and Console Mode Installation

The examples in this guide provide the Solaris executable name for the installation program. However, you may be installing on AIX or Linux.

- For AIX, use: ca-im-*release*-aix.bin
- For LINUX, use: ca-*release*-linux.bin

release represents the current release of CA Identity Manager

If you are performing an installation in console mode, such as on a UNIX workstation, you add another option to the command line.

- For the main installation, add -i console. For example:
  `./ca-im-release-sol.bin -i console`
- For installation of provisioning components, add -console to the setup command.

Red Hat Linux 64-bit Installation

If you plan to install CA Identity Manager on a Red Hat Linux 64-bit system, you need to prepare the system for the installation.

Follow these steps:

Install four 32-bit packages using the following commands:

```
yum install glibc.i686
yum install libXext.i686
yum install libXtst.i686
yum install ncurses-devel.i686
```

Note: The i686 suffix specifies that the library is 32-bit, for the x86 processor.

Alternatively, the dependencies may be resolved using Add/Remove Software, and unchecking the Only Native Packages filter option. Using this approach, you select and install the required i686 architecture dependencies.

The native ksh shell package also needs to be installed. Use the following command:

```
yum install ksh
```

Another alternative is to resolve the package dependency by using Add/Remove Software. Using this approach, you select and install the required i686 architecture dependencies ksh package.

Non-Provisioning Installation

This guide refers to the Windows and Solaris program names for the installers that provide options to install provisioning software. If you prefer to see no provisioning options, you can use these installers:

- For Windows, use IMWithoutProvisioning\ca-im-web-*release*-win.bat
- For Solaris, use IMWithoutProvisioning/ca-im-web-*release*-sol.sh

release represents the current release of CA Identity Manager.

Appendix C: Unattended Installation

This section contains the following topics:

How to Run an Unattended Installation

Follow these steps:

1. Modify the im-installer.properties file.
2. Run the following command:
   - **Windows:**
     `ca-im-release-win32.exe -f im-installer.properties -i silent`
   - **UNIX:**
     `./ca-im-release-sol.bin -f im-installer.properties -i silent`

Modify the Configuration File

To enable an unattended CA Identity Manager installation, modify the settings in the im-installer.properties configuration file using a text editor. The default parameters in the file reflect the information entered during the initial CA Identity Manager installation. Change the default values as needed.

Note the following when modifying the configuration file:

- Make a back-up copy of the installer properties file before modifying the original, since the file holds all of the values you entered during the initial installation or configuration.
- Do not add extra spaces between the parameter name, the equals sign (=), and the attribute value.
- All directory names on Windows must contain either double back slashes or forward slashes, not single back slashes.

Initial Choices

For basic installation choices, enter values for the following parameters:

Parameter	Instructions
DEFAULT_NEW_INSTANCE_DISPLAY_NAME	Enter 'New Installation' if this is a fresh install. For upgrades, this will be blank.
DEFAULT_COMPONENTS	Enter one or more components: ■ Server - CA Identity Manager Server ■ Exten - Extensions to the Policy Server ■ Admin - CA Identity Manager Administrative Tools ■ Provision - Provisioning Server ■ Directory - Provisioning Directory To install more than one component, separate components by a comma.
DEFAULT_INSTALL_FOLDER	Enter the directory in which to install the CA Identity Manager Server.
DEFAULT_GENERIC_USERNAME	Generic login information for CA Identity Manager components that are installed.
DEFAULT_GENERIC_PASSWORD	Generic password information for CA Identity Manager components that are installed.
DEFAULT_FIPS_MODE	Select if you want to enable FIPS 140-2 compliance.
DEFAULT_FIPS_KEY_LOC	Enter the path to the FIPS key location.

The installation program ignores any parameters that do not apply to the component you are installing. For example, if you set DEFAULT_COMPONENTS to Exten, only the DEFAULT_PS_ROOT and DEFAULT_USE_SITEMINDER parameters are used.

CA Identity Manager Server

If you plan to install the CA Identity Manager Server, enter values for the following:

Parameter	Instructions
DEFAULT_APP_SERVER	Enter, Weblogic, WebSphere, or JBoss

Parameter	Instructions
DEFAULT_APP_SERVER_URL	Enter full URL of the application server hosting CA Identity Manager, including the port.
DEFAULT_JAVA_HOME	Path to JRE or JDK for CA Identity Manager.
Additional Database Parameters	
DEFAULT_DB_HOST	Enter the hostname of the system hosting the CA Identity Manager database.
DEFAULT_DB_PORT	Enter the port of the system hosting the CA Identity Manager database.
DEFAULT_DB_NAME	Enter the name of the CA Identity Manager database.
DEFAULT_DB_USER	Enter the administrative username for the CA Identity Manager database.
DEFAULT_DB_PASSWORD	Enter the password for the administrative user of the CA Identity Manager database.
DEFAULT_DB_TYPE	Enter the type of database used for the CA Identity Manager database.
Additional JBoss Parameter	
DEFAULT_JBOSS_FOLDER	Enter the full pathname of the directory where you installed the JBoss application server. For example, C:\jboss-5.1
Additional WebLogic Parameters	
DEFAULT_BINARY_FOLDER	Enter the full directory path of the directory where you installed WebLogic. For example: C:\Oracle\Middleware\weblogic\
DEFAULT_DOMAIN_FOLDER	Enter the full path and directory name for the WebLogic domain you created for CA Identity Manager.
DEFAULT_SERVER_NAME	Enter the name of the WebLogic server instance you created for use with CA Identity Manager.

Parameter	Instructions
DEFAULT_BEA_CLUSTER	Enter the cluster name for the WebLogic cluster.
Additional WebSphere Parameters	
DEFAULT_WEBSPHERE_FOLDER	Enter the full pathname of the directory where you installed CA Identity Manager Tools for WebSphere.
DEFAULT_WAS_NODE	Enter the name of the node in which the application server is located.
DEFAULT_WAS_SERVER	Enter the name of the system on which the application server is running.
DEFAULT_WAS_CELL	Enter the name of the cell in which the application server is located.
WAS_PROFILE	Enter the location of the WebSphere profile files.
DEFAULT_WAS_CLUSTER	Enter the cluster name for the WebSphere cluster.

If you are using a SiteMinder Policy Server, enter the following:

Parameter	Instruction
DEFAULT_PS_HOST	Enter the fully-qualified domain name of the Policy Server.
DEFAULT_PS_USER	Enter the user name of the Policy Server administrator.
DEFAULT_PS_PW	Enter the password of the Policy Server administrator.

Provisioning Components

If you install Provisioning, enter the following:

Parameter	Instruction
DEFAULT_CONFIG_REMOTE PROVISIONING	Enter true if you are connecting to a remote Provisioning Directory.
DEFAULT_DEPLOYMENT_SIZE	Enter the size of your Provisioning Directory deployment.
DEFAULT_DIRECTORY_IMPS_HOSTN AMES	Enter the hostnames of all Provisioning Servers that will connect to the Directory.
DEFAULT_DOMAIN_NAME	Enter 'im' unless you have an existing Provisioning domain.
DEFAULT_DIRECTORY_HOST	Enter the hostname of the system with Provisioning Directory installed.
DEFAULT_DIRECTORY_PORT	Enter the port number of the system with the Provisioning Directory installed.
DEFAULT_DIRECTORY_PASSWORD	Enter the password for the Provisioning Directory.

Extensions for SiteMinder

To install the extensions for a SiteMinder Policy Server, enter the following:

Parameter	Instruction
DEFAULT_PS_ROOT	(Solaris Only) Enter the directory where the Policy Server is installed.
DEFAULT_USE_SITEMINDER	Enter true if you are using a SiteMinder Policy Server in your implementation.

Configuration File Format

The im-installer.properties file is located in the CA Identity Manager installation directory. For example:

- **Windows**: C:\Program Files\CA\CA Identity Manager\install_config_info
- **UNIX**: /opt/CA/IdentityManager/install_config_info/im-installer.properties

The following is an example of the im-installer.properties file created during a CA Identity Manager installation:

```
############################################################
### Silent input properties file for the IM R12.5SP7 installer ###
############################################################

# Component list
#   Valid values (comma-separated, one or more):
Server,Exten,Admin,Provision,Directory
DEFAULT_COMPONENTS=

# Install folder
#   All products are installed in subfolders under this folder
#   This is parent product root selected by the user
#   For e.g. C:\\Program Files\\CA\Identity Manager
DEFAULT_INSTALL_FOLDER=

#Generic login information
DEFAULT_GENERIC_USERNAME=
#DEFAULT_GENERIC_PASSWORD=<For silent install, insert generic user password here and
uncomment line.>

#Optionally enable management console security - a default user will be created with
the generic login credentials above.
DEFAULT_SECURE_MANAGEMENT_CONSOLE=

# Provisioning Server and Provisioning Directory Information.
#  Configure the Provisioning Server to a remotely installed Provisioning
Directory(true/false)
DEFAULT_CONFIG_REMOTE_PROVISIONING=

 #Select the deployment type that suits your needs (1,2,3 or 4): 1. Compact 2. Basic
3. Intermediate (64 Bit only) 4. Large (64 Bit only)
DEFAULT_DEPLOYMENT_SIZE=
DEFAULT_DIRECTORY_IMPS_HOSTNAMES=
DEFAULT_DOMAIN_NAME=
DEFAULT_DIRECTORY_HOST=
DEFAULT_DIRECTORY_PORT
#DEFAULT_DIRECTORY_PASSWORD=<For silent install, insert password to be used with
Provisioning Components here and uncomment line.>
```

```
#FIPS 140-2 Compliance mode (true/false) for Identity Manager, Admin Tools,
Provisioning Manager and Provisioning Server
DEFAULT_FIPS_MODE=
#Complete path of the FIPS key file. For e.g. C:\\Program Files\\FIPSkey.dat
DEFAULT_FIPS_KEY_LOC=

#Use custom encryption properties for encrypting sensitive data
DEFAULT_KEY_PARAMS_ENABLED=
#Abs path of the encrpytion properties file. E.g. C:\\Program
Files\\keyParams.properties
DEFAULT_KEY_PARAMS_LOC=


#Identity Manager Application Server information
#   App Server
#   Valid values:  JBoss, WebLogic, WebSphere
DEFAULT_APP_SERVER=
DEFAULT_APP_SERVER_URL=

#Path to JDK for the JBOSS Application Server. No input required for other Application
Servers
DEFAULT_JAVA_HOME=

#JBoss info
DEFAULT_JBOSS_FOLDER=
DEFAULT_JBOSS_PROFILE=
DEFAULT_JBOSS_SERVER_ID=

#Weblogic info
DEFAULT_BINARY_FOLDER=
DEFAULT_DOMAIN_FOLDER=
DEFAULT_SERVER_NAME=
DEFAULT_BEA_CLUSTER=

#WebSphere info
DEFAULT_WEBSPHERE_FOLDER=

#WAS_NODE Value: $WAS_HOME$\installedApps\$WAS_NODE$ or
$WAS_HOME$\config\cells\$WAS_CCELL$\nodes\$WAS_NODE$. These should be same.
DEFAULT_WAS_NODE=

#WAS_SERVER Value:
$WAS_HOME$\config\cells\$WAS_CELL$\nodes\$WAS_NODE$\servers\$WAS_SERVER$
DEFAULT_WAS_SERVER=

#WAS_CELL Value: $WAS_HOME$\config\cells\$WAS_CELL$
DEFAULT_WAS_CELL=
```

```
#WAS_PROFILE Value: $WEBPHERE_HOME$\profiles\$WAS_PROFILE$
WAS_PROFILE=

#WAS_CLUSTER Value: $WAS_HOME$\config\cells\$WAS_CELL$\clusters\$WAS_CLUSTER$
DEFAULT_WAS_CLUSTER=

DEFAULT_WAS_NO_AUTO_DEPLOY=$WAS_NO_AUTO_DEPLOY$

#Policy Server info
DEFAULT_PS_HOST=
DEFAULT_PS_USER=
#DEFAULT_PS_PW=<For silent install, insert PS Admin user password here and uncomment
line.>

#8.1 Migration
# SiteMinder Agent Name
DEFAULT_AGENT_NAME=
# SiteMinder Shared Secret
#DEFAULT_AGENT_PW=<For silent install, insert PS Shared Secret here and uncomment
line.>
# Automatically migrate. Valid values (true/false)
DEFAULT_MIGRATE_DIR_ENV=
# Directory to export to
DEFAULT_DIR_ENV_EXPORT=

#Policy Server Extensions info
#   Location of CsSmPs-<Instance name> folder
DEFAULT_PS_ROOT=
#You can use the SiteMinder Policy Server and a SiteMinder Web Agent to provide advanced
security
# for CA Identity Manager environments. Valid values (true/false)
DEFAULT_USE_SITEMINDER=

#Database Info
DEFAULT_DB_HOST=
DEFAULT_DB_PORT=
DEFAULT_DB_NAME=
DEFAULT_DB_USER=
#DEFAULT_DB_PASSWORD=<For silent install, insert database password here and uncomment
line.>

#Following are permissible values: mssql2005 or oracle10
DEFAULT_DB_TYPE=

#WAS Message Engine Database Info
DEFAULT_ME_HOST=
DEFAULT_ME_PORT=
DEFAULT_ME_NAME=
```

```
DEFAULT_ME_USER=
#DEFAULT_ME_PASSWORD=<For silent install, insert database password here and uncomment
line.>
DEFAULT_ME_SCHEMA=

#Upgrading from IM8.1sp2
#       If you have data stores located on separate servers or you wish to install
them on separate
#       servers, you can specify them below. Otherwise if you wish to have all the
data stores on
#       the same server, change the DEFAULT_DB_* properties from above.

#Object Store Datastore Info
#DEFAULT_OS_DB_HOST=
#DEFAULT_OS_DB_PORT=
#DEFAULT_OS_DB_NAME=
#DEFAULT_OS_DB_USER=
#DEFAULT_OS_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Task Persistence Datastore Info
#DEFAULT_TP_DB_HOST=
#DEFAULT_TP_DB_PORT=
#DEFAULT_TP_DB_NAME=
#DEFAULT_TP_DB_USER=
#DEFAULT_TP_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Audit Datastore Info
#DEFAULT_AUDIT_DB_HOST=
#DEFAULT_AUDIT_DB_PORT=
#DEFAULT_AUDIT_DB_NAME=$AUDIT_DB_USER$
#DEFAULT_AUDIT_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Reporting Snapshot Datastore Info
#DEFAULT_RS_DB_HOST=
#DEFAULT_RS_DB_PORT=
#DEFAULT_RS_DB_NAME=
#DEFAULT_RS_DB_USER=
#DEFAULT_RS_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

#Workflow Datastore Info
#DEFAULT_WF_DB_HOST=
#DEFAULT_WF_DB_PORT=
#DEFAULT_WF_DB_NAME=
#DEFAULT_WF_DB_USER=
```

```
#DEFAULT_WF_DB_PASSWORD=<For silent install, insert database password here and
uncomment line.>

# Automatically Upgrade Workflow DB
DEFAULT_UPGRADE_WF_DB=

# Automatically Migrate Task Persistance
DEFAULT_MIGRATE_TP=$

# HTTP Proxy settings
DEFAULT_HTTP_PROXY_ENABLED=
DEFAULT_HTTP_PROXY_HOST=
DEFAULT_HTTP_PROXY_PORT=
DEFAULT_HTTP_PROXY_DOMAIN=
DEFAULT_HTTP_PROXY_USERNAME=
DEFAULT_HTTP_PROXY_PASSWORD=
```

Appendix D: Installation Log Files

The log files are stored based on where you unpack the installation package. The following examples may have different top-level directories than these default locations.

This section contains the following topics:

Log Files on Windows

If you encounter issues during CA Identity Manager installation, see this log file:

C:\Program Files\CA\Identity Manager\IAM Suite\Identity Manager\caiamsuite.log

The CA Identity Manager Server installer logs are written to the following default locations:

- C:\Program Files\CA\Identity Manager\install_config_info (32-bit system)
- C:\Program Files (x86)\CA\Identity Manager\install_config_info (64-bit system)

The Provisioning installer logs are written to the user's Temp directory and copied to the *Install-Directory*_uninst directory.

Example:

C:\Documents and Settings*user*\Local Settings\Temp\imps_server_install.log

Log files on UNIX

If you encounter any issues while performing a CA Identity Manager installation, see the caiamsuite.log file in this location:

/opt/CA/IdentityManager/

The CA Identity Manager Server installer logs are written to the following default location:

/opt/CA/IdentityManager/install_config_info

The Provisioning installer logs are written to the user's Temp directory.

Appendix E: CA Identity Manager as a Windows Service

This section contains the following topics:

Windows Service on JBoss 5

Use this procedure to configure CA Identity Manager as a service on a JBoss 5 system.

Follow these steps:

1. Edit the service.bat file found in the JBoss bin folder.
2. Locate the run.bat lines, which appear as follows:

   ```
   call run.bat < .r.lock >> run.log 2>&1
   ```

3. If you have a JBoss cluster, change these lines, so that they appear as follows:

   ```
   call run.bat -c all < .r.lock >> run.log 2>&1
   ```

   The file contains two occurrences of this line.

4. Locate the service identity lines, which appear as follows:

   ```
   set SVCNAME=JBAS50SVC
   set SVCDISP=JBoss Application Server 5.1
   ```

5. Change these lines, so that they appear as follows:

   ```
   set SVCNAME=CAIMSVC
   set SVCDISP=CA Identity Manager
   ```

6. Save the file.
7. From a command prompt, execute the service.bat script to install the Windows Service:

   ```
   Service.bat install
   ```

8. Using the Services tool, change the service Startup Type from Manual to Automatic.
9. Start the CA Identity Manager Service.
10. View the JBoss log to verify a successful launch.

Windows Service on JBoss 4

Use this procedure to configure CA Identity Manager as a service on a JBoss 4 system.

CA Identity Manager as a Windows Service (JBoss 5)

Use this procedure to configure CA Identity Manager as a service on a JBoss 5 system.

Follow these steps:

1. Edit the service.bat file found in the JBoss bin folder.
2. Locate the run.bat lines, which appear as follows:

   ```
   call run.bat < .r.lock >> run.log 2>&1
   ```

3. If you have a JBoss cluster, change these lines, so that they appear as follows:

   ```
   call run.bat -c all < .r.lock >> run.log 2>&1
   ```

   The file contains two occurrences of this line.

4. Locate the service identity lines, which appear as follows:

   ```
   set SVCNAME=JBAS50SVC
   set SVCDISP=JBoss Application Server 5.1
   ```

5. Change these lines, so that they appear as follows:

   ```
   set SVCNAME=CAIMSVC
   set SVCDISP=CA IdentityMinder
   ```

6. Save the file.
7. From a command prompt, execute the service.bat script to install the Windows Service:

   ```
   service.bat install
   ```

8. Using the Services tool, change the service Startup Type from Manual to Automatic.
9. Start the "CA IdentityMinder" Service.
10. View the JBoss log to verify a successful launch.

Install the Java Service Wrapper Files

Four files are required to use the Java Service Wrapper and three additional files are provided to launch JBoss manually and install or uninstall the Windows Service.

1. Download the Java Service Wrapper.
2. Copy the following files into the JBoss **bin** directory:
   - *wrapper_home*\bin\wrapper.exe—the Java Service Wrapper executable
   - *wrapper_home*\src\bin\App.bat.in—the batch file to run JBoss in a console
   - *wrapper_home*\src\bin\InstallApp-NT.bat.in—the batch file to install the Windows Service
   - *wrapper_home*\src\bin\UninstallApp-NT.bat.in—the batch file to uninstall the Windows Service

   wrapper_home is the location where you installed the Java Service Wrapper.
3. Rename the three batch files from Step 1 as follows:
   - *jboss_home*\bin\CAIdentityManager.bat
   - *jboss_home*\bin\InstallCAIdentityManagerService.bat
   - *jboss_home*\bin\UninstallCAIdentityManagerService.bat
4. Copy the following files into the JBoss **lib** directory:
   - *wrapper_home*\lib\wrapper.dll—native library required by the Java Service Wrapper
   - *wrapper_home*\lib\wrapper.jar—Java Service Wrapper classes
5. Create the following directory:

   jboss_home\conf
6. Copy the following files into this **conf** directory:

   wrapper_home\src\conf\wrapper.conf.in—the Java Service Wrapper configuration
7. Rename the file as follows:

   jboss_home\conf\wrapper.conf

Configure the Java Service Wrapper

The libraries, classes, and parameters for CA Identity Manager must be configured in the Java Service Wrapper configuration file:

jboss_home\conf\wrapper.conf

Note: Property values that are paths to directories or files should *not* be enclosed in quotation marks. Forward (/) or back-slashes (\) can be used as a path separator.

Local Environment Variables

Several local environment variables will be created in order to simplify later configuration and to prevent the default ability of the Java Service Wrapper to use the %PATH% from the environment. Careful inspection of the run-idm.bat file will reveal that the %PATH% is carefully constructed in order to eliminate any SiteMinder library version conflicts. These variables are not system-wide and will only be available for the JVM created by the Java Service Wrapper. Some system-wide environment variables are used in the creation of these local variables.

Add the following properties to the beginning of wrapper.conf before any other properties:

- set.JAVA_HOME=[JAVA_HOME from run-idm.bat]

  Example: set.JAVA_HOME=C:\CA\j2sdk1.4.2_14

- set.NETE_SPS_PATH=[resolved NETE_SPS_PATH from run-idm.bat]

  Example: set.NETE_SPS_PATH=C:\CA\eTrust SiteMinder\agentframework\bin

- set.IM_EAR=../server/default/deploy/iam_im.ear
- set.SYSTEM_PATH=%SystemRoot%\SYSTEM32;%SystemRoot%;%SystemRoot%\SYSTEM32\WBEM
- set.PATH=%IGW_LOC%;%NETE_SPS_PATH%;%NETE_PS_PATH%;%IM_EAR%/library;%SYSTEM_PATH%

Java Executable

During the CA Identity Manager installation, a java SDK was selected and was used to set the JAVA_HOME variable at the top of *jboss_home*\bin\run-idm.bat. The local environment variable for this location is used in the following property:

wrapper.java.command=%JAVA_HOME%\bin\java

Java Classpath

Create all java classpath entries populated in *jboss_home*\bin\run.bat. Also, include the Java Service Wrapper classes in the classpath that are used within the JVM. That list is as follows, taking advantage of local environment variables created previously:

- wrapper.java.classpath.1=../lib/wrapper.jar
- wrapper.java.classpath.2=%JAVA_HOME%\lib\tools.jar
- wrapper.java.classpath.3=./run.jar

Java Library Path

Add the required libraries for JBoss to the library path and the environment path that were created previously:

- wrapper.java.library.path.1=../lib
- wrapper.java.library.path.2=../server/default/lib
- wrapper.java.library.path.3=%PATH%

Java Arguments

Create the Java arguments that are populated in *jboss_home*\bin\run-idm.bat and *jboss_home*\bin\run.bat. That list is as follows, taking advantage of local environment variables created previously and excluding memory settings that will be configured separately:

- wrapper.java.additional.1=-server
- wrapper.java.additional.2=-Dprogram.name=run.bat
- wrapper.java.additional.4=-Djava.security.policy=.\workpoint_client.policy
- wrapper.java.additional.5=-XX:MaxPermSize=128m

Java Memory Sizes

Set the JVM memory settings as follows:

- wrapper.java.initmemory=256
- wrapper.java.maxmemory=512

Main Class

Specify the main class that the Java Service Wrapper should be called as follows:

wrapper.app.parameter.1=org.jboss.Main

Java Service Wrapper Logging

The location of the log file for the Java Service Wrapper will default to *jboss_home*/logs/wrapper.log. Settings can be changed as described in the wrapper.conf file.

Windows Service Names

Specify the names to be used for the Windows Service as follows:

- wrapper.ntservice.name=IdentityMinder
- wrapper.ntservice.displayname=CA Identity Manager
- wrapper.ntservice.description=CA Identity Manager

Install the Windows Service

Follow these steps:

1. Run the following batch file to test the configuration:
   `CAIdentityManager.bat`

   If CA Identity Manager starts in a console, continue on to Step 2.
2. Close the CA Identity Manager console.
3. Run the following batch file to install the Windows Service:
   `InstallCAIdentityManagerService.bat`

Example of a wrapper.conf File

A complete and working wrapper.conf file is provided here for reference:

```
#********************************************************************
# Wrapper Properties
#********************************************************************
# Local Environment Variables
set.JAVA_HOME=C:\CA\j2sdk1.4.2_14
set.NETE_SPS_ROOT=C:\CA\eTrust SiteMinder
set.NETE_SPS_PATH=%NETE_SPS_ROOT%\agentframework\bin
set.IM_EAR=../server/default/deploy/iam_im.ear
set.SYSTEM_PATH=%SystemRoot%\SYSTEM32;%SystemRoot%;%SystemRoot%\SYSTEM32\WBEM
set.PATH=%IGW_LOC%;%NETE_SPS_PATH%;%NETE_PS_PATH%;%IM_EAR%/library;%SYSTEM_PATH%

# Java Application
wrapper.java.command=%JAVA_HOME%\bin\java

# Java Main class.  This class must implement the WrapperListener interface
#  or guarantee that the WrapperManager class is initialized.  Helper
#  classes are provided to do this for you.  See the Integration section
#  of the documentation for details.
wrapper.java.mainclass=org.tanukisoftware.wrapper.WrapperSimpleApp

# Java Classpath (include wrapper.jar)  Add class path elements as
#  needed starting from 1
wrapper.java.classpath.1=../lib/wrapper.jar
wrapper.java.classpath.2=%JAVA_HOME%\lib\tools.jar
wrapper.java.classpath.3=./run.jar

# Java Library Path (location of Wrapper.DLL or libwrapper.so)
wrapper.java.library.path.1=../lib
wrapper.java.library.path.2=../server/default/lib
wrapper.java.library.path.3=%PATH%
```

```
# Java Additional Parameters
wrapper.java.additional.1=-server
wrapper.java.additional.2=-Dprogram.name=run.bat
wrapper.java.additional.3=-Djava.security.policy=.\workpoint_client.policy
wrapper.java.additional.4=-XX:MaxPermSize=128m
wrapper.java.additional.5=-Dsun.rmi.dgc.client.gcInterval=3600000
wrapper.java.additional.6=-Dsun.rmi.dgc.server.gcInterval=3600000
wrapper.java.additional.7=-Xms256m
wrapper.java.additional.7=-Xmx512m

# Initial Java Heap Size (in MB)
wrapper.java.initmemory=256


# Maximum Java Heap Size (in MB)
wrapper.java.maxmemory=512


# Application parameters.  Add parameters as needed starting from 1
wrapper.app.parameter.1=org.jboss.Main

#*********************************************************************
# Wrapper Logging Properties
#*********************************************************************
# Format of output for the console.  (See docs for formats)
wrapper.console.format=PM

# Log Level for console output.  (See docs for log levels)
wrapper.console.loglevel=INFO

# Log file to use for wrapper output logging.
wrapper.logfile=../logs/wrapper.log

# Format of output for the log file.  (See docs for formats)
wrapper.logfile.format=LPTM

# Log Level for log file output.  (See docs for log levels)
wrapper.logfile.loglevel=DEBUG

# Maximum size that the log file will be allowed to grow to before
#  the log is rolled. Size is specified in bytes.  The default value
#  of 0, disables log rolling.  May abbreviate with the 'k' (kb) or
#  'm' (mb) suffix.  For example: 10m = 10 megabytes.
wrapper.logfile.maxsize=10m

# Maximum number of rolled log files which will be allowed before old
#  files are deleted.  The default value of 0 implies no limit.
wrapper.logfile.maxfiles=5
```

```
# Log Level for sys/event log output.  (See docs for log levels)
wrapper.syslog.loglevel=NONE

#********************************************************************
# Wrapper Windows Properties
#********************************************************************
# Title to use when running as a console
wrapper.console.title=CAIdentityMinder


#********************************************************************
# Wrapper Windows NT/2000/XP Service Properties
#********************************************************************
# WARNING - Do not modify any of these properties when an application
#  using this configuration file has been installed as a service.
#  Please uninstall the service before modifying this section.  The
#  service can then be reinstalled.


# Name of the service
wrapper.ntservice.name=CAIdentityMinder


# Display name of the service
wrapper.ntservice.displayname=CA IdentityMinder

# Description of the service
wrapper.ntservice.description=CA IdentityMinder


# Service dependencies.  Add dependencies as needed starting from 1
wrapper.ntservice.dependency.1=


# Mode in which the service is installed.  AUTO_START or DEMAND_START
wrapper.ntservice.starttype=AUTO_START

# Allow the service to interact with the desktop.
wrapper.ntservice.interactive=false
```

Appendix F: Windows Services Started by CA Identity Manager

The following are the services started on Windows when you install and start all components of CA Identity Manager:

- CA Directory *hostname*-impd-co
- CA Directory impd-inc
- CA Directory impd-main
- CA Directory impd-notify
- CA Directory impd-router
- CA Identity Manager Connector Server (C++)
- CA Identity Manager Connector Server (Java)
- CA Identity Manager Provisioning Server
- Enterprise Common Services (Transport)
- Enterprise Common Services GUI Framework
- Enterprise Common Services Store-And-Forward Manager

This list of services may useful to you for troubleshooting purposes.

Index

J

L

M

N

O

P

R

S

T

U

V

W