Oracle® Communications Instant Messaging Server

Installation and Configuration Guide

Release 10.0.1

E76290-01

August 2016

ORACLE®

Oracle Communications Instant Messaging Server Installation and Configuration Guide, Release 10.0.1

E76290-01

Contents

B comm_dssetup.pl Reference

C Instant Messaging Server Configuration Script

Preface

This guide provides instructions for installing and configuring Oracle Communications Instant Messaging Server.

Audience

This document is intended for system administrators or software technicians who install and configure Instant Messaging Server. This guide assumes you are familiar with the following topics:

- Oracle Directory Server Enterprise Edition and LDAP
- System administration and networking

Related Documents

For more information, see the following documents in the Instant Messaging Server documentation set:

- *Instant Messaging Server Release Notes*: Describes the fixes, known issues, troubleshooting tips, and required third-party products and licensing.
- *Instant Messaging Server System Administrator's Guide*: Provides instructions for administering Instant Messaging Server.
- *Instant Messaging Server Security Guide*: Provides guidelines and recommendations for setting up Instant Messaging Server in a secure configuration.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

1

Instant Messaging Server Installation Overview

This chapter provides an overview of the Oracle Communications Instant Messaging Server installation process.

Overview of Instant Messaging Server Installed Components

During the installation process, you install and configure the following components:

- Java
- Instant Messaging Server

Instant Messaging Server depends on Oracle Communications Directory Server Enterprise Edition for LDAP services. If your site does not currently have Directory Server deployed and you need to install it, see the Oracle Directory Server Enterprise Edition documentation for instructions, at:

http://docs.oracle.com/cd/E29127_01/index.htm

For Instant Messaging Server to use notifications offline, you must have an email server installed, such as Oracle Communications Messaging Server. For Instant Messaging Server to use the calendar agent, you must have Oracle Communications Calendar Server installed. See the Messaging Server and Calendar Server documentation for information on installing those products.

Overview of the Instant Messaging Server Installation Procedure

The installation procedure follows these steps:

1. Plan your installation. When planning your installation, do the following:
   - Determine the scale of your implementation, for example, a small development system, or a large production system.
   - Determine how many physical machines you need, and which software components to install on each machine.
   - Plan the system topology, for example, how the system components connect to each other over the network.
2. Review system requirements. System requirements include:
   - Hardware requirements, such as disk space.
   - System software requirements, such as operating system (OS) versions and OS patch requirements.
   - Information requirements, such as IP addresses and host names.

3. Install and configure software upon which Instant Messaging Server is dependent, including Java.
4. Prepare the Directory Server schema by installing and running the most current **comm_dssetup** script from the Instant Messaging Server software.
5. Install and configure Instant Messaging Server.
6. Perform post-installation configuration tasks.
7. Verify the installation.

After Instant Messaging Server is installed, you might perform additional security-related tasks, such as configuring Secure Sockets Layer (SSL) communications between Instant Messaging Server front ends and back ends. For more information, see *Instant Messaging Server Security Guide*.

Instant Messaging Server Installation Options

You install Instant Messaging Server in either interactive or silent mode. When you run the installer in silent mode, you are running a non-interactive session. The installation inputs are taken from the following sources:

- A silent installation file
- Command-line arguments
- Default settings

You can use silent mode to install multiple instances of the same software component and configuration without having to manually run an interactive installation for each instance.

Ensuring a Successful Instant Messaging Server Installation

Only qualified personnel should install the product. You must be familiar with the UNIX operating system. You should be experienced with installing Java-related packages.

Follow these guidelines:

- As you install each component, for example, verify that the component installed successfully before continuing the installation process.
- Pay close attention to the system requirements. Before you begin installing the software, make sure your system has the required base software. In addition, ensure that you know all of the required configuration values, such as host names and port numbers.
- As you create new configuration values, write them down. In some cases, you might need to re-enter configuration values later in the procedure.

Directory Placeholders Used in This Guide

Table 1–1 lists the placeholders that are used in this guide:

Table 1–1 Instant Messaging Server Directory Placeholders

Placeholder	Directory
InstantMessaging_home	Specifies the installation location for the Instant Messaging Server software. The default for both Solaris and Linux is **/opt/sun/comms/im**.
InstantMessaging_cfg	Specifies the installation location for the configuration directory. The default for Solaris is **/etc/opt/sun/comms/im/default/config**. The default for Linux is **/etc/opt/sun/im/default/config**.
InstantMessaging_database	Specifies the location for the database directory, if you are using a file-based property store. The default for Solaris is **/var/opt/sun/comms/im/default/db**. The default for Linux is **/var/opt/sun/im/default/db**.
InstantMessaging_runtime	Specifies the location for the configurable directory for the files generated by the server at runtime. The default for Solaris is **/var/opt/sun/comms/im/default** and Linux is **/var/opt/sun/im/default**.

2

Planning Your Instant Messaging Server Installation

This chapter provides information about planning your Oracle Communications Instant Messaging Server installation. It also describes the Instant Messaging Server logical and physical architectures.

About Instant Messaging Server

Instant Messaging Server enables secure, real-time communication and collaboration, combining presence awareness with instant messaging capabilities such as chat, conferences, and file transfers to create a rich collaborative environment. These features enable one-to-one and group collaboration through either short-lived communications or persistent venues such as conference rooms. Instant Messaging Server ensures the integrity of communications through its multiple authentication mechanisms and Secure Sockets Layer (SSL) connections. For more information about Instant Messaging Server, see *Oracle White Paper—Oracle Communications Instant Messaging Server* at:

http://www.oracle.com/us/industries/communications/oracle-instant-messaging-wp-1449642.pdf

Instant Messaging Server Components

Instant Messaging Server consists of the following internal core components that must integrate and inter-operate with external services to provide an instant messaging environment.

- **Instant Messaging Server**. The Instant Messaging server component provides core services required for real time communications such as the presence engine, message handling and routing, roster management, security and authorization. The Instant Messaging server supports the connection of an Instant Messaging multiplexer that concentrates connections over one socket.
- **Instant Messaging Multiplexor**. The Instant Messaging multiplexer adds scalability to the Instant Messaging environment. You can install multiple multiplexers as needed, depending on your configuration. As the user population grows beyond what is easily supported by a single Instant Messaging server, you can deploy additional servers to which you can connect additional multiplexers. In addition, you can configure an Instant Messaging multiplexor with a list of Instant Messaging Server hosts for failover.

- **Access, Communication, and Transfer Protocols**. These protocols, such as LDAP, HTTP, TCP/IP, and SMTP, can be found in "Instant Messaging Server Supported Standards".
- **Instant Messaging API**. Enables you to customize Instant Messaging in a variety of ways. For example, you can write custom clients and archive providers, add user presence information to other applications, develop custom authentication mechanisms, and so forth. For more information, see the topic on APIs in *Instant Messaging Server System Administrator's Guide*.

Instant Messaging Server Gateways

Instant Messaging Server provides the following gateways to enable connectivity to other systems:

- **XMPP/HTTP Gateway**. Enables Instant Messaging Server to provide access to HTTP-based clients, such as HTML-based clients, and clients behind firewalls that allow HTTP traffic, but do not permit XMPP traffic. The gateway proxies instant messaging traffic to the XMPP server on behalf of HTTP clients.
- **XMPP WebSocket Gateway**. Enables Instant Messaging Server to support the WebSocket protocol for XMPP.
- **SMS Gateway**. Enables Instant Messaging Server to deliver chat messages and alerts in the form of SMS to offline users.
- **SIP Gateway**. Instant Messaging Server implements a SIP/SIMPLE (Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions/Session Initiation Protocol) gateway, enabling federation and translation between the two protocols, and interoperation between XMPP and SIP/SIMPLE servers.

Components Related to Instant Messaging Server

The software components discussed in this section work with Instant Messaging Server, but are installed separately. See "Instant Messaging Server Planning Considerations" for more information on how these components interact with Instant Messaging Server.

Directory Server

(Required) Instant Messaging Server requires a directory server, such as Oracle Directory Server Enterprise Edition or Oracle Unified Directory for end user authentication and search, and to store conference rooms. Additionally, if you enable conference history persistence, that too is stored in directory server. This directory can either be dedicated for use by Instant Messaging Server, or be shared by other components.

By default, Instant Messaging Server relies on the common end-user attributes **cn** and **uid** to search for end-user and group information. If you want, you can configure the server to use another attribute for search. In addition, Instant Messaging properties (such as contact lists and subscriptions) are stored in the directory server. For instructions on configuring the server to use a non-default attribute for user search, see *Instant Messaging Server System Administrator's Guide*.

By default Instant Messaging Server uses the directory server to store user properties and data, which includes multiuser chat history. Instead of using the directory server to store multiuser chat history, you can use Oracle Database. For more information on

storing multiuser chat history in Oracle Database, see *Instant Messaging Server System Administrator's Guide*.

Web Container

(Optional) You must install a web container for the Instant Messaging Server components described in Table 2–1.

Table 2–1 Required Web Containers for Instant Messagng Server Components

Web Container	Instant Messaging Components
Oracle GlassFish Server	Use GlassFish Server if you are hosting Bidirectional-streams Over Synchronous HTTP (BOSH) or using Presence API components. In addition, GlassFish Server is required as a web container if you are deploying the Oracle Communications Convergence client.
Oracle WebLogic Server	WebLogic Server supports the PresenceAPI and XMPP WebSocket Gateway component.
Oracle Converged Application Server	Converged Application Server supports the SIP/SIMPLE component.

Note: Install the web container before configuring Instant Messaging Server.

SMTP Server

(Optional) An SMTP messaging server, such as Oracle Communications Messaging Server, is used to forward instant messages, in the form of email, to end users who are offline. The SMTP server can also be used to archive instant messaging communications. The SMTP server does not have to reside on the same host as the Instant Messaging Server host.

Calendar Server

(Optional) Oracle Communications Calendar Server is used to notify users of calendar-based events and also users' presence status while in either an event or a to-do.

Instant Messaging Server Supported Standards

This section lists national, international, industry, and de-facto standards related to electronic messaging and for which support is claimed by Oracle Communications Instant Messaging Server. Most of these are Internet standards, published by the RFC Editor and approved by the Internet Engineering Task Force (IETF). Standards for documents from other sources are noted.

Protocols Support by Instant Messaging Server

Instant Messaging Server supports the following protocols:

- Extensible Messaging and Presence Protocol(XMPP):
  - XMPP Core Protocols (RFC 3920, RFC 3921)
  - XMPP Extensions

- Short message peer-to-peer protocol (SMPP)

Table 2–2 lists the supported XMPP Extensions.

Table 2–2 Supported XMPP Extensions

Number	Name	Comments
XEP-0004	Data Forms	No comments
XEP-0016	Privacy Lists	No comments
XEP-0022	Message Events	No comments
XEP-0030	Service Discovery	No comments
XEP-0045	Multiuser Chat	No comments
XEP-0047	In-Band Bytestreams	No comments
XEP-0048	Bookmarks	No comments
XEP-0049	Private XML Storage	No comments
XEP-0054	vcard-temp	No comments
XEP-0055	Jabber Search	No comments
XEP-0065	SOCKS5 Bytestreams	No comments
XEP-0066	Out of Band Data	Implemented by Client
XEP-0071	XHTML-IM	Implemented by Client
XEP-0077	In-Band Registration	No comments
XEP-0078	Non-SASL Authentication	No comments
XEP-0079	Advanced Message Processing	No comments
XEP-0085	Chat State Notifications	Implemented by Client
XEP-0092	Software Version	No comments
XEP-0095	Stream Initiation	No comments
XEP-0096	SI File Transfer	No comments
XEP-0100	Gateway Interaction	No comments
XEP-0106	JID Escaping	No comments
XEP-0114	Jabber Component Protocol	No comments
XEP-0124	Bidirectional-streams Over Synchronous HTTP (BOSH)	No comments
XEP-0126	Invisibility	No comments
XEP-0153	vCard-Based Avatars	No comments
XEP-0160	Best Practices for Handling Offline Messages	No comments
XEP-0166	Jingle	Implemented by Client
XEP-0167	Jingle RTP Sessions	Implemented by Client
XEP-0177	Jingle Raw UDP Transport Method	Implemented by Client
XEP-0191	Simple Communications Blocking	No comments
XEP-0206	XMPP Over BOSH	No comments
XEP-0107	User Mood	No comments
XEP-0108	User Activity	No comments

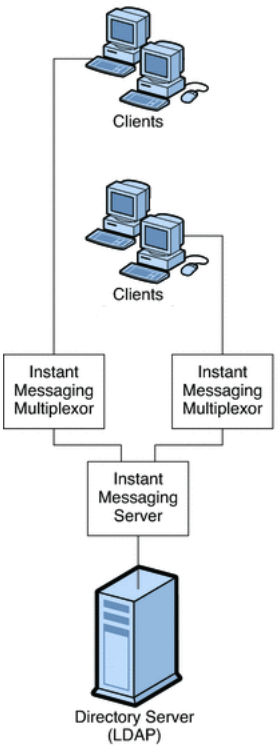
Table 2–2 (Cont.) Supported XMPP Extensions

Number	Name	Comments
XEP-0118	User Tune	No comments
XEP-0163	Personal Eventing Protocol	No comments
XEP-0060	Publish-Subscribe	Partially implemented
XEP-0199	XMPP Ping	Support for client-to-server and server-to-client pings
XEP-0215	XMPP External Service Discovery	Implemented by custom plugin
XEP-0280	Message Carbons	No comments
XEP-0313	Message Archive Management	Implemented by custom plug-in except for XEP-0059 support
XEP-0333	Chat Markers	No comments
RFC-7395	XMPP Over Web Sockets	No comments

Instant Messaging Server Software Architecture

Figure 2–1 shows the Instant Messaging Server software architecture.

Figure 2–1 Instant Messaging Server Software Architecture



Clients send messages to the multiplexor, which forwards the messages to the Instant Messaging server. Clients send messages to one another through a multiplexor, which forwards the messages to the Instant Messaging server.

The directory server stores and retrieves local user and group delivery information such as preferences, location, and to which multiplexor to route messages for this user. When the Instant Messaging server receives a message, it uses this information to

determine where and how the message should be delivered. In addition, the directory server can contain user information such as contact lists and subscriptions.

In this basic configuration, Instant Messaging Server directly accesses the directory server to verify user login name and passwords for mail clients that use Instant Messaging.

Outgoing instant messages from clients go directly to the multiplexor. The multiplexor sends the message to the appropriate Instant Messaging server, which in turn forwards the message to another Instant Messaging server, or if the message is local, to the multiplexor with which the recipient is associated.

New users are created by adding user entries to the directory server. Entries in the directory can be created through Instant Messaging Server (by enabling new-user registration feature) or changed by using the tools provided with the directory server. You can then assign services to the user. For more information about new user registration for Instant Messaging Server, see the topic on administering end users in *Instant Messaging Server System Administrator's Guide*.

You administer Instant Messaging Server components through a set of command-line interfaces and text-based configuration files.

Note: Typical Instant Messaging Server deployments are not installed on a single machine. They also have additional features like multiplexing and high availability enabled. See "Instant Messaging Server Planning Considerations" for more information.

Instant Messaging Server Planning Considerations

This section contains the following planning topics you must consider before installing Instant Messaging Server:

- Planning to Protect Instant Messaging Server
- Planning Instant Messaging User Authentication
- Planning for Anonymous Directory Server Searching
- Planning Instant Messaging Server Privacy, Security, and Site Policies
- Planning to Protect the Instant Messaging Archive
- Planning for a Basic Installation
- Planning for Email Notification Architecture and Calendar Alerts

Planning to Protect Instant Messaging Server

Instant Messaging Server supports Transport Layer Security (TLS). Instant Messaging Server uses a startTLS extension to the TLS 1.0 protocol for client-to-server and server-to-server encrypted communications and for certificate-based authentication between servers.

When planning for SSL for Instant Messaging Server, keep in mind the following:

- You can secure the Instant Messaging Server deployment by enabling SSL on the web container port (either Web Server or Application Server) and accessing Instant Messaging Server functionality by using the XMPP/HTTP Gateway (httpbind).

- Set the proper file and directory permissions for the Instant Messaging Server configuration files (**im.conf.xml** and **httpbind.conf** in the *InstantMessaging_cfg*/**config** directory):
  - Solaris OS:

    ```
    /etc/opt/sun/comms/im/default/config/
    ```

  - Red Had Linux:

    ```
    /etc/opt/sun/im/default/config/
    ```

    Instant Messaging Server runs as the user specified in the **iim.conf.xml** file. This user needs read access to the file. If you use httpbind, the user that runs the web container should be able to access the Instant Messaging Server directory path and configuration file. When you create additional Instant Messaging server or multiplexor instances manually, you must also ensure that the file and directory permissions are correct. The default installation sets the file and directory permissions. The default instance directory has the following permissions:

    ```
    drwx------   5 root     other        512 Oct 16 14:24 default
    ```

- Take care while enabling Instant Messaging Server monitoring, as this exposes server statistics that could be considered security issues. The default configuration does not enable the monitoring feature. You enable this property through the **iim.conf.xml** file.
- Enable debug logging only when needed, as this impacts overall system performance. Though passwords are not logged, the protocol communication between users is logged, which could be a potential security issue.
- When you enable startTLS, use a single server certificate for both client-to-server and server-to-server communication.
- An Instant Messaging Server deployment that leverages LDAP needs proper authentication for access.

The Instant Messaging Server default installation supports only SASL Plain. If you require a higher level of security, use the Instant Messaging public Service Provider Interface. SASL Plain and jabber:iq:auth are two forms of plain text authentication. That is, in both, the password is sent in the clear (encoded perhaps, but still clear text) and so both are insecure forms of authentication. Nevertheless, this is an issue only if end-to-end encryption (through startTLS for direct socket connection, and HTTPS for httpbind) is not enabled. If end-to-end encryption is enabled, the password is not "seen" in the clear on the network.

Alternatively, if you do not want to transmit passwords in the clear (even if over an encrypted stream), use the Instant Messaging SPI for plugging in authentication mechanism's at the server side through SASLRealm. You can implement custom SASL mechanisms as implementations but you will then need an Instant Messaging client that supports this custom mechanism.

See *Instant Messaging Server Security Guide* for more information on using TLS.

Planning Instant Messaging User Authentication

User authentication enables your users to log in through their Instant Messaging clients to chat and access other features of Instant Messaging.

Instant Messaging Server and Passwords

User IDs and passwords are stored in your LDAP directory. Password security criteria, such as minimum length, are determined by directory policy requirements. Password security criteria is not part of Instant Messaging Server administration. See the directory server documentation to understand directory server password policies.

Instant Messaging Server and LDAP

All Instant Messaging Server deployments require a directory server to perform end user authentication and to search for end users. For various ways to secure the directory, see the directory server documentation.

The default Instant Messaging Server configuration makes the following assumptions regarding the schema used by this directory:

- End user entries are identified by the **inetOrgPerson** object class.
- Group entries are identified by the **groupOfUniqueNames** or **groupofURLs** object class.
- The email address of an end user is provided by the **mail** attribute.
- The display name of an end user or group is provided by the **cn** attribute.
- The list of members of a group is provided by the **uniqueMember** attribute (**groupOfUniqueNames** object class).

Note: Some user attributes might contain confidential information. Ensure that your directory access control is set up to prevent unauthorized access by non-privileged users.

Planning for Anonymous Directory Server Searching

Instant Messaging Server needs to be able to search the directory to function correctly. If you want, you can configure your directory to be searchable by anonymous users. To configure your directory to be readable or searchable by anonymous users, see the topic on enabling the directory server searches on a specific user in *Instant Messaging Server System Administrator's Guide*.

Planning Instant Messaging Server Privacy, Security, and Site Policies

Instant Messaging Server provides the ability to control access to Instant Messaging Server features and preserve end-user privacy.

Instant Messaging Server Site Policies

Site policies specify end-user access to specific functionality in Instant Messaging Server. When developing your site policies for Instant Messaging Server, keep in mind the following questions:

- Can users access the presence status of other end users?
- Can users send alerts to other end users?
- Do you want users to save properties on the server?
- Who do you want to be able to create and manage conference rooms?

For more information, see *Instant Messaging Server Security Guide*.

Controlling Instant Messaging Server End User and Administrator Privileges

Different sites using Instant Messaging Server have different needs in terms of enabling and restricting the type of access end users have to the Instant Messaging service. The process of controlling end user and administrator Instant Messaging Server features and privileges is referred to as *policy management*. You can manage policies to adjust end-user privileges in the following areas: conference room management, the ability to change preferences in the User Settings dialog, and ability to send alerts. It also enables specific end users to be assigned as system administrators.

For more information, see *Instant Messaging Server Security Guide*.

Planning to Protect the Instant Messaging Archive

Instant Messaging Server has the capability to archive instant messages for later retrieval and searching. If you enable the email archive, you must decide which administrators are to receive email containing archived instant messages. You can configure a separate list of administrators to receive polls, news, conference, alerts, or chat sessions. You can also configure Instant Messaging Server to use the extended RFC 822 header. Doing so enables mail clients to filter messages based on the header content.

Planning for a Basic Installation

The basic Instant Messaging Server architecture provides such functionality as presence, roster management, chat, and conferences. To provide this basic functionality, you must install the following components:

- Instant Messaging server and one or more Instant Messaging multiplexors
- Directory server

In the basic architecture:

- The directory server provides user entries for authentication and lookup.
- Clients always connect to the Instant Messaging server through an Instant Messaging multiplexor.

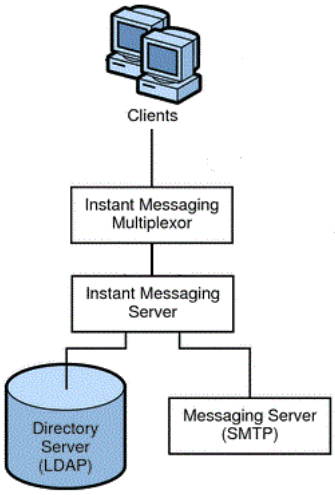
Planning for Email Notification Architecture and Calendar Alerts

You can deploy Instant Messaging Server to support email notification to offline users, and Instant Messaging based notification of calendar events to users.

Figure 2–2 shows an Instant Messaging Server deployment that supports email notification to offline users.

An Instant Messaging Server architecture that supports email notification provides the same functionality as Basic Instant Messaging Architecture. To provide this functionality, you must include the components listed in "Planning for a Basic Installation". To support email alerts, you also install an SMTP server such as Oracle Communications Messaging Server.

Figure 2–2 Instant Messaging Architecture with Email Notification



To enable email notification, you are prompted during installation to identify the SMTP server to use with Instant Messaging. If you do not have an SMTP server installed, you must install one before installing the Instant Messaging software.

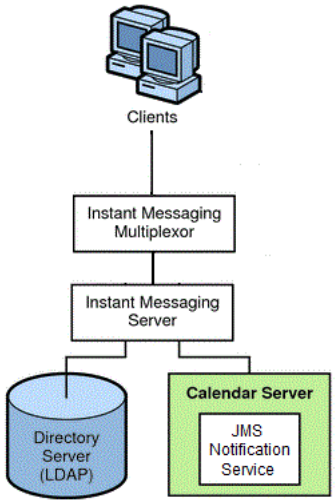
Authentication flow in this architecture is the same as in a basic deployment.

In this example:

- The directory server provides user entries for authentication and lookup.
- The Instant Messaging server forwards messages intended for offline users to the SMTP server. The SMTP server then sends the message as an email to the user's mailbox.
- Clients always connect to the Instant Messaging server through an Instant Messaging multiplexor.

Figure 2–3 shows Instant Messaging Server with calendar notification enabled on the network. If you do not have Calendar Server installed, you must install it before installing the Instant Messaging software.

Figure 2–3 Instant Messaging Architecture with Calendar Alerts



In this example:

- The directory server provides user entries for authentication and lookup.
- Java Message Service (JMS) sends notifications of calendar events to the Instant Messaging server which then forwards the notification on to the appropriate end user.
- Clients always connect to the Instant Messaging server through an Instant Messaging multiplexor.

System Deployment Planning

This section contains the following system-level planning topics you must consider before installing Instant Messaging Server:

Planning for High Availability

You can use server pooling to provide high availability (HA) for your Instant Messaging Server deployment. Server pools provide redundancy so that if one server in the pool fails, affected clients can reconnect and continue their sessions through another server in the pool with a minimum of inconvenience. Additionally, if you set up your deployment with load balancers, users can immediately reconnect and be directed by a load balancer to another node in the pool. For more information, see the topic on scaling Instant Messaging Server by using server pooling in *Instant Messaging Server System Administrator's Guide*.

You can also configure an Instant Messaging multiplexor with a list of Instant Messaging Server hosts for failover. For more information, see the topic on multiplexor failover in *Instant Messaging Server System Administrator's Guide*.

Providing Instant Messaging Client Access Through a Firewall

The XMPP/HTTP Gateway (httpbind) provides Instant Messaging access to XMPP-based clients, such as HTML based clients, and clients that are behind firewalls that allow HTTP traffic only and do not permit XMPP traffic. The gateway proxies Instant Messaging Server traffic to the XMPP server on behalf of HTTP clients.

When planning to use the XMPP/HTTP Gateway, keep in mind the following:

- Use port 5222 at the Gateway if the Gateway is communicating to the server through a multiplexor. Also, use port server port 5269 if no multiplexor is involved. You can specify port 5222 or 5269 in the **httpbind.conf** file.
- The XMPP/HTTP gateway supports startTLS. When you enable startTLS on the server, all communications is encrypted.
- Do not expose the Instant Messaging server to direct access. In a typical deployment scenario, locate the multiplexor in the DMZ, and open the multiplexor to server communication port (45222 usually) in the second firewall.

Using Load Balancing

Instant Messaging Server supports the use of load balancers located in front of the Instant Messaging multiplexors.

Planning Backup Strategies

Backing up and restoring data is one of the most important administrative tasks for your Instant Messaging Server deployment. You must implement a backup and restore policy for your Instant Messaging Server deployment to ensure that data is not lost if the system crashes, hardware fails, or information is accidentally deleted.

You must back up the following Instant Messaging Server information:

- Configuration Information
- Instant Messaging end user data

The configuration information is stored in the configuration directory (*InstantMessaging_cfg*). The Instant Messaging data is stored in the database directory (*InstantMessaging_database*).

For more information on backing up your Instant Messaging Server deployment, see *Instant Messaging Server System Administrator's Guide*.

Sample Instant Messaging Server Physical Architecture

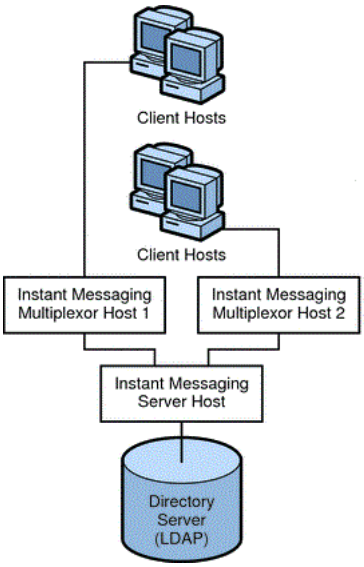
This section explains variations to the deployment scenario described in "Planning for a Basic Installation". For example, you can install the various required servers and components in the following physical configurations, or any combination of each example:

- Physical Deployment Example: Multiplexors on Separate Hosts
- Physical Deployment Example: Multiple Instant Messaging Hosts

Physical Deployment Example: Multiplexors on Separate Hosts

Figure 2–4 shows a configuration consisting of two multiplexors installed on two separate hosts. The Instant Messaging server is installed on a different host. This configuration enables you to place a multiplexor outside your company's firewall. Installing multiplexors on multiple hosts distributes the load of Instant Messaging Server across multiple systems.

Figure 2–4 Instant Messaging Multiplexors Installed on Separate Hosts

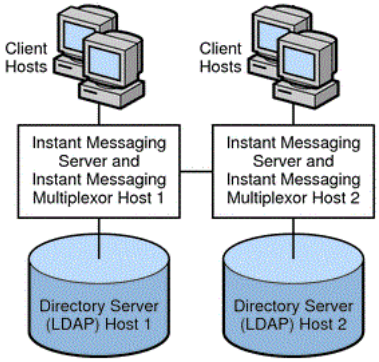


Note: The multiplexor can be resource-intensive, so putting it on a separate host can improve the overall performance of the system.

Physical Deployment Example: Multiple Instant Messaging Hosts

Figure 2–5 shows a configuration consisting of two Instant Messaging servers. This configuration is used when the site contains multiple administrative domains. The server configuration on each Instant Messaging server host has to be set up so that end users on one Instant Messaging server can communicate with end users on other Instant Messaging servers.

Figure 2–5 Multiple Instant Messaging Server Hosts



About Installing a Secure System

In conjunction with the TLS protocol, Instant Messaging Server provides client-to-server and server-to-server encrypted communications and certificate-based authentication between servers. For information about secure installation and configuration of Instant Messaging Server, see *Instant Messaging Server Security Guide*.

3

Instant Messaging Server System Requirements

This chapter describes the hardware, operating system, software, and database requirements for installing Oracle Communications Instant Messaging Server.

Software Requirements

This section describes the software and information requirements to install Instant Messaging Server.

Supported Operating Systems

Table 3–1 lists operating systems that support Instant Messaging Server.

Table 3–1 Instant Messaging Server Operating System Requirements

Product	Version
Oracle Solaris on SPARC	11
Oracle Solaris on x64	11
Oracle Linux on x64 (64-bit)	7
Red Hat Enterprise Linux on x64 (64-bit)	7

Required Software

Table 3–2 lists other software required for installing and running Instant Messaging Server.

Table 3–2 Instant Messaging Server Software Requirements

Product	Version	Notes
Directory server	Oracle Directory Server Enterprise Edition 6.x, 7, 11.*x*	For a fresh installation, use the latest version of Directory Server 11gR1.
Application server	Oracle GlassFish Server 3.1.2 Oracle WebLogic Server 12.2.1 Oracle Converged Application Server 5.1	GlassFish Server supports the HTTPBind and PresenceAPI components. WebLogic Server supports the PresenceAPI and WebSockets components. Converged Application Server supports the SIP/SIMPLE component.
Java Development Kit (JDK)	Java 8 with latest critical patch update Java 7 with latest critical patch update	Java 8 is required for Instant Messaging Server and for components running on WebLogic Server. Java 7 is required for components running on GlassFish Server and Converged Application Server.

Table 3–3 lists the database software for Instant Messaging Server if you choose to store multiuser chat history in a database.

Table 3–3 Database Requirements

Product	Version	Notes
Oracle Database	11*g*R2 (11.2) 12*c* (12.1)	Currently, you can only store multiuser chat history and no other data in Oracle Database.

Client Requirements

Instant Messaging Server does not include a messaging client. Pidgin is the tested and supported XMPP client. Ideally any XMPP-compliant client should work with Instant Messaging Server.

Hardware Requirements

The number and configuration of the systems that you employ for your Instant Messaging Server installation depends on the scale and the kind of deployment you have planned.

Note: The sizing estimates in this section assume proper application configuration and tuning, in a manner consistent with leading practices of Oracle Communications consulting and performance engineering. This information is provided for informational purposes only and is not intended to be, nor shall it be construed as a commitment to deliver Oracle programs or services. This document shall not form the basis for any type of binding representation by Oracle and shall not be construed as containing express or implied warranties of any kind. You understand that information contained in this document will not be a part of any agreement for Oracle programs and services. Business parameters and operating environments vary substantially from customer to customer and as such not all factors, which may impact sizing, have been accounted for in this documentation.

Table 3–4 provides the minimum hardware requirements for Instant Messaging Server.

Table 3–4 Instant Messaging Server Minimum Hardware Requirements

Component	Requirement
Disk Space	Approximately 300 MB required for Instant Messaging Server software.
RAM	At least 256 MB of RAM. The amount of RAM needed depends on the number of concurrent client connections, and whether the server and multiplexor are deployed on the same host.

Information Requirements

This section describes the information that you must provide during the installation and initial configuration process.

Component Information

Table 3–5 lists the component information that you provide during initial configuration.

Table 3–5 Component Information

Information Type	Default Value
Server components	**Selected**
Web components	**Selected**

Service Runtime Information

Table 3–6 lists the service runtime information that you provide during initial configuration.

Table 3–6 Service Runtime Information

Information Type	Default Value	Comments
Runtime user ID	**inetuser**	If the **configure** utility does not create a UNIX user, you must create it manually. After you create the user for Instant Messaging Server, set permissions appropriately for the directories and files owned by that user. Do not choose **root** as a server user ID.
Runtime group ID	**integroup**	If the **configure** utility does not create a UNIX group, you must create it manually. After you create the group for Instant Messaging Server, set permissions appropriately.
Runtime directory	**/var/opt/sun/comms/im**	Runtime files are read, created, and modified by the server during its normal operations. Some examples include log files, and persistent state information tied to client actions such as roster information, conferences, and so on. If you are configuring Instant Messaging Server for high availability, this path must be globally available. The **configure** utility appends a directory (**/default**) to the path you provide for the runtime files. The name of this directory is the instance to which the runtime files apply. Later, you can create multiple Instant Messaging Server instances by creating additional instance directories with different names (for example **/secure**) and copying over files from the **/default** instance runtime directory.

Network Access Information

Table 3–7 lists the network access information that you provide during initial configuration.

Table 3–7 Network Access Information

Information Type	Default Value	Comments
Domain name	*host domain name*	No comment.
XMPP port	**5222**	The port number on which the Instant Messaging server listens for incoming requests from clients.
Multiplexed XMPP port	**45222**	The port number on which the Instant Messaging server listens for incoming requests from the multiplexor.
XMPP Server Port	**5269**	The port number on which the Instant Messaging server listens for incoming requests from other Instant Messaging servers. In addition, if no multiplexor is installed, the server listens for incoming requests from clients on this port. The port is also used by components such as HTTPBIND gateway, Calendar Agent, and the SMS gateway for creating a component session with the server.
Disable server (enable only multiplexor)	**No**	Select this option if the instance you installed acts as a multiplexor and not a server. If you select this option, you must provide a value for Remote Instant Messaging Server Host Name.
Enable SSL	**Yes**	When enabling SSL, you are prompted for a keystore file and password file. Also, the respective server configuration is mandatorily set to TLS for all communication. To disable mandating TLS, set **iim_server.requiressl** to **false** by using the **imconfutil** command.
Server keystore file	None	No comment.
Server password file	None	No comment.

If you decide to enable SSL, the respective server configuration is mandatorily set to TLS for all communication. To disable mandating TLS, set **iim_server.requiressl=false** by using the **imconfutil** command.

Directory Server Information

Table 3–8 lists the directory server information that you provide during initial configuration.

Table 3–8 LDAP Information

Information Type	Default Value	Comments
LDAP host name	*FQDN of host*	In a deployment with a directory server, specifies the host name of the directory server that contains user and group information for Instant Messaging. For example, **directory.example.com**.
LDAP port number	**389**	In a deployment with a directory server, specifies the port number on which the directory server listens for incoming requests. For example, **389**.
SSL enabled	**No**	No comment.

Table 3–8 (Cont.) LDAP Information

Information Type	Default Value	Comments
Base DN	**dc=siroe,dc=com**	In a deployment with a directory server, specifies the base distinguished name in the directory tree that contains user and group information for Instant Messaging. For example, **o=example.com**.
Directory manager DN	**Directory Manager**	Instant Messaging Server uses this Bind DN to search users and groups in the directory. Leave this blank if the directory can be searched anonymously. You can change the bind credentials later if required.
Directory manager password	No default value.	In a deployment with a directory server, the Bind DN password.

Email Information

Table 3–9 lists the email information that you provide during initial configuration.

Table 3–9 Email Information

Information Type	Default Value	Comments
Enable Email Integration	**Yes**	If selected, enables Instant Messaging Server email integration. Dependencies: SMTP Server such as Oracle Communications Messaging Server
SMTP Server	**smtphost**	The host name of the SMTP server used to send email notification of messages to offline users. For example, **mail.example.com**. If the SMTP server does not use port 25, specify the port along with the host name. For example, if the SMTP server uses port 1025, then use **mail.example.com:1025**. Dependencies: SMTP server such as Oracle Communications Messaging Server
Enable email archiving	**yes**	If selected, enables Instant Messaging Server email archiving. Dependencies: SMTP Server such as Oracle Communications Messaging Server

HTTP Gateway Information

Table 3–10 lists the HTTP gateway information that you provide during initial configuration.

Table 3–10 HTTP Gateway Information

Information Type	Default Value	Comments
Deploy IM HTTP gateway	**Yes**	Determines if the XMPP/HTTP gateway is deployed. If you choose to deploy the gateway, the **configure** utility creates a default gateway configuration file (**httpbind.conf**) in the default Instant Messaging server instance's *InstantMessaging_cfg* directory if one does not already exist. If **httpbind.conf** already exists, the **configure** utility does not alter or overwrite the file. If you are configuring Instant Messaging Server to support Convergence, do not enable the XMPP/HTTP Gateway Deployment here. Set this value to **false**. The XMPP/HTTP Gateway is deployed through the Convergence server. Its value is set when you configure Convergence.
Context root	**http://***imhost***:80/httpbind**	Defines the URI for the HTTP component of the XMPP/HTTP gateway.
Web container path	*Web container base directory*	No comment.
Web administrator URL	No default value.	No comment.
Web administrator user ID	**admin**	No comment.
Web administrator password	No default value.	No comment.

Calendar Agent Information

Table 3–11 lists the calendar agent information that you provide during initial configuration.

Table 3–11 Calendar Agent Information

Information Type	Default Value	Comments
Enable calendar agent	**No**	If you choose to enable the Calendar agent, you must provide the following information: From the **configure** panel: ■ Choose to Enable Calendar Agent by typing **yes** ■ Choose to Enable local component by typing **yes** ■ Specify XMPP server host name ■ Specify XMPP server port ■ Specify JMQ user name ■ Specify JMQ password ■ Specify Notification Server host name ■ Specify Notification Server port ■ Specify Topic If you choose not to enable the Calendar agent, you can manually configure the Calendar agent later.
Enable local component	**No**	No comment.

SMS Gateway Information

Table 3–12 lists the Short Message Services gateway information that you provide during initial configuration.

Table 3–12 SMS Gateway Information

Information Type	Default Value	Comments
Enable SMS gateway	**No**	Enables the Instant Messaging server to deliver chat messages and alerts in the form of SMS to the Instant Messaging users who are offline.
Enable local component	**No**	No comment.

Services Information

Table 3–13 lists the Instant Messaging services startup information that you provide during initial configuration.

Table 3–13 Services Information

Information Type	Default Value
Start services after successful configuration	**Yes**
Start services when system starts	**Yes**

4

Instant Messaging Server Pre-Installation Tasks

This chapter describes the pre-installation tasks that you must complete before you can install Oracle Communications Instant Messaging Server.

Installing Java

The 32-bit and 64-bit Java Development Kits (JDKs) must be installed before installing Instant Messaging Server or any of its components. Install both JDKs, rather than the JRE, on your Instant Messaging Server hosts.

See "Required Software" for information about Java version requirements for the various Instant Messaging Server components.

To get the Java software, go to the Java SE Downloads at:

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

Installing GlassFish Server

To install and configure Oracle GlassFish Server, see *Oracle GlassFish Server 3.1.2 Installation Guide* at:

http://docs.oracle.com/cd/E26576_01/doc.312/e24935/installing.htm#ggssq

Installing WebLogic Server

To install and configure Oracle WebLogic Server, see *Oracle Fusion Middleware Installing and Configuring Oracle WebLogic Server and Coherence* at:

https://docs.oracle.com/middleware/1212/core/WLSIG/

Installing the Directory Server

Instant Messaging Server uses a directory server to store and access data for individual users, groups, and domains.

If your site does not currently have Directory Server deployed, and you need to install it, see the Oracle Directory Server Enterprise Edition documentation at:

http://docs.oracle.com/cd/E29127_01/index.htm

Prior to installing and configuring Instant Messaging Server, you must also prepare the directory server schema by running the **comm_dssetup.pl** script. This script, which

is provided as part of the Instant Messaging Server software, adds the necessary schema to the directory. See "Preparing Directory Server" for more information.

To understand the schema that is used by Instant Messaging Server, refer to *Unified Communications Suite Schema Reference*.

5

Installing Instant Messaging Server

This chapter describes how to install and configure Oracle Communications Instant Messaging Server.

Before installing Instant Messaging Server, read these chapters:

- Instant Messaging Server Installation Overview
- Planning Your Instant Messaging Server Installation
- Instant Messaging Server System Requirements
- Instant Messaging Server Pre-Installation Tasks

Installation Assumptions

The instructions in this chapter assume:

- You are deploying Instant Messaging Server on a single host, or multiple hosts or Solaris zones.
- Oracle Directory Server Enterprise Edition is already installed.
- (Optional) You have installed and configured a web container for Instant Messaging Server components that require one.

Installing Instant Messaging Server

The tasks to install Instant Messaging Server are as follows:

- Downloading the Instant Messaging Server Software
- Preparing Directory Server
- Installing the Instant Messaging Server Software

Downloading the Instant Messaging Server Software

1. Download the Instant Messaging Server software for your operating system, and the Oracle Communications Directory Server Setup **comm_dssetup.pl** script, from the Oracle software delivery website, located at:

   `http://edelivery.oracle.com/`

2. Copy the Instant Messaging Server ZIP file to a temporary directory on your Instant Messaging Server hosts and extract the files.

3. Copy the Directory Server Setup ZIP file to a temporary directory on your Directory Server hosts and extract the files, to be able to install and run the **comm_dssetup.pl** script.

Preparing Directory Server

You prepare your Directory Server by running the **comm_dssetup.pl** script against it. You can run the **comm_dssetup.pl** script in either interactive or silent mode. For silent mode instructions, see "Running the comm_dssetup.pl Script in Silent Mode".

Running the comm_dssetup.pl Script in Interactive Mode

To prepare Directory Server and run the **comm_dssetup.pl** script in interactive mode:

1. On the host where Directory Server is installed, log in as or become the superuser **(root)**.
2. Start Directory Server, if necessary.
3. Change to the directory where you extracted the Directory Server Setup ZIP file and run the installer.

   ```
   commpkg install
   ```

   For more information about running the installer, see "commpkg Reference".
4. Select **Comms DSsetup** and proceed with the installation.
5. Run the **comm_dssetup.pl** script in interactive mode (without any arguments), then enter your choices when prompted.

   ```
   /usr/bin/perl comm_dssetup.pl
   ```

   For more information, see "comm_dssetup.pl Reference".

   Note: You can use either LDAP Schema 2 or Schema 1.

6. If necessary, provision users in the Directory Server.

   If Directory Server is already installed at your site, users have already been provisioned. If you have just installed Directory Server at your site, then you need to provision users. For information, see the discussion on provisioning users and schema in the *Schema Reference*.

Installing the Instant Messaging Server Software

To install the Instant Messaging Server software:

1. On the Instant Messaging Server host, log in as or become the superuser **(root)**.
2. Go to the directory where you extracted the Instant Messaging Server files.
3. Run the Installer.

   ```
   commpkg install
   ```

   For more information about running the installer, see "commpkg Reference".
4. Select **Instant Messaging Server** and proceed with the installation.

Installing Instant Messaging Server in Silent Mode

When you run the Instant Messaging Server installer in silent mode, you are running a non-interactive session. The installation inputs are taken from the following sources:

- A silent installation file (also known as a state file)
- Command-line arguments
- Default settings

You can use silent mode to install multiple instances of the same software component and configuration without having to manually run an interactive installation for each instance.

This section includes:

Performing a Instant Messaging Server Silent Installation

To perform a Instant Messaging Server silent installation:

1. Obtain the state file by one of the following means.
   - Run an interactive installation session and use the state file that is created in the **/var/opt/CommsInstaller/logs/** directory. The state file name is similar to **silent_CommsInstaller_20070501135358**. A state file is automatically created for every run of the installation.
   - Create a silent state file without actually installing the software during the interactive session by using the **--dry-run** option, then modifying the state file. For example:

     ```
     commpkg install --dry-run
     ```

2. Copy the state file to each host machine and edit the file as needed. See "Silent Mode File Format".
3. Run the silent installation on each host. For example:

   ```
   commpkg install --silent input_file
   ```

   where *input_file* is the path and name of the silent state file, for example **/var/opt/CommsInstaller/logs/silent_CommsInstaller_20070501135358**.

   For details about the **--silent** option, see "install Verb Syntax".

   Note: Command-line arguments override the values and arguments in the state file.

About Upgrading Shared Components

By default, shared components that require user acceptance for upgrading are not upgraded when you run a silent installation. The option to upgrade shared components in the silent state file is automatically disabled. That is, the option is set to **UPGRADESC=No**. This is true even if you explicitly asked to upgrade shared components when you ran the interactive installation that generated the silent state

file. That is, you ran either **commpkg install --upgradeSC y** or you answered "yes" when prompted for each shared component that needed upgrading.

Disabling upgrading shared components in the silent state file is done because the other hosts on which you are propagating the installation might have different shared components installed, or different versions of the shared components. Therefore, it is safer to not upgrade the shared components by default.

You can upgrade shared components when you run a silent installation by performing either of the following actions:

- Use the **--upgradeSC y** option when you run the silent installation. (The command-line argument overrides the argument in the state file.)
- Edit the **UPGRADESC=No** option in the silent state file to: **UPGRADESC=Yes**.

Caution: If you do not upgrade shared components your installation might not work properly.

Silent Mode File Format

The silent mode file (also known as a state file) is formatted like a property file: blank lines are ignored, comment lines begin with a number sign (#), and properties are key/value pairs separated by an equals (=) sign. Table 5–1 shows which options you can change and provides examples:

Table 5–1 Silent Mode File Options

Option	Description	Example
VERB	Indicates which function to perform. For a silent install, this is set to **install**.	VERB=install
ALTDISTROPATH	Indicates an alternate distro path.	ALTDISTROPATH=SunOS5.10_i86pc_DBG.OBJ/release
PKGOVERWRITE	A boolean indicating whether to overwrite the existing installation packages. (See the **--pkgOverwrite** switch).	PKGOVERWRITE=YES
INSTALLROOT	Specifies installation root.	INSTALLROOT=/opt/sun/comms
ALTROOT	A boolean indicating whether this is an alternate root install.	ALTROOT=yes
EXCLUDEOS	Specifies to not upgrade operating system patches.	EXCLUDEOS=YES
EXCLUDESC	Specifies to exclude shared component patches.	EXCLUDESC=no
COMPONENTS	A space separated list of mnemonics of the components to be installed. You can precede the mnemonic with a ~ to indicate that only the shared components for that product be installed.	COMPONENTS=IM
ACCEPTLICENSE	This option is no longer used.	Not applicable

Table 5–1 (Cont.) Silent Mode File Options

Option	Description	Example
UPGRADESC	Indicates whether all shared components should or should not be upgraded without prompting.	UPGRADESC=no
INSTALLNAME	The friendly name for the INSTALLROOT.	INSTALLNAME=
COMPONENT_ VERSIONS	This option is unused.	Not applicable

To display a complete list of the product names (such as MS, MS64, IM) to use with the **COMPONENTS** property, run the **commpkg info --listPackages** command. This command displays the mnemonics for each product. For more information, see the discussion on the **commpkg info** command in "commpkg Reference".

Installing Instant Messaging Server on Solaris Zones

This information explains how to install Instant Messaging Server on Solaris OS Zones.

The topics in this section include:

- Installing on Solaris OS Zones: Best Practices
- Installing into a Non-Global Whole Root Zone
- Installing into a Non-Global Sparse Root Zone

Installing on Solaris OS Zones: Best Practices

You can install Instant Messaging Server in the global zone, whole root non-global zones, and sparse non-global zones. Follow these guidelines:

- Treat the global zone as an “administration zone.”

  Install shared components and OS patches in the global zone that are to be shared among all zones. However, do not install and run products from the global zone.

- Use whole root non-global zones to run Instant Messaging Server.

  Do not use the global zone or sparse zones. A whole root zone can have versions that are different from other whole root zones, thus giving it a measure of being “self-contained.”

Be aware of the following zone aspects:

- You can have different shared component versions in the whole root non-global zone, but it isn't entirely insulated. If you do a packaging or patching operation in the global zone for a shared component, that operation is also attempted in the whole root zone. Thus, to truly have different shared component versions, use an alternate root.
- To avoid affecting whole root zones you can attempt to never install and patch shared components in the global zone. However, it might not be realistic to never have to install or patch a shared component in the global zone. For example, NSS is a shared component, but it is part of Solaris OS. So to expect to never install and patch NSS in the global zone seems unrealistic, especially given it is a security component.

- Although it isn't a recommended best practice, you can use Instant Messaging Server in sparse non-global zones. Do note that shared components cannot be installed into the default root because many of them install into the read-only shared file system (**/usr**). Thus, you must run the installer in the global zone to install shared components into the default root. Prepend your selection with ~ in the global zone to install only the dependencies (that is, shared components). You do not have to install in the global zone first before installing in the sparse zone. The installer allows you to continue even when you do not install all the dependencies. However, upgrading the shared components in the global zone affects the sparse non-global zones, thus requiring downtime for all affected zones simultaneously.

Note: Sparse root zones are not available beginning with Oracle Solaris 11.

Installing into a Non-Global Whole Root Zone

The non-global whole root zone scenario is the equivalent of installing Instant Messaging Server on a single box with no zones. Simply install Instant Messaging Server as you normally would.

Caution: Any operations performed in the global zone (such as installations, uninstallations, and patching) affect the whole root zones.

Installing into a Non-Global Sparse Root Zone

Although it isn't a recommended best practice, you can use Instant Messaging Server in a non-global sparse root zone on Solaris 10. To install Instant Messaging Server in a non-global sparse root zone, you first need to install or upgrade the applicable OS patches and shared components in the global zone. You are unable to do so in the sparse root zone, because the **/usr** directory (where the shared components reside) is a read-only directory in the sparse root zone.

1. Follow the pre-installation requirements as described in "Instant Messaging Server Pre-Installation Tasks".
2. Verify that you are about to install the shared components and OS patches in the global zone and not the sparse root zone. To verify you are in the global zone, run **zonename**. The output should be global.
3. Run the installer in the global zone and only install or upgrade the OS patches and the Shared Components. Do not install Instant Messaging Server in the global zone. To do this, add a ~ (tilde) to the component number you want to install in the sparse zone.

   For example, if you plan to install Instant Messaging Server in the sparse zone, you select **~1** during the global zone installation. The installer will know to only install dependencies and not the product itself.
4. Once you have the shared components and OS patches installed, install Instant Messaging Server in the sparse root zone.

Configuring Instant Messaging Server

You must configure Instant Messaging Server to complete the installation. You use the Instant Messaging Server configuration command-line script, **configure**, to perform this initial runtime configuration.

Creating a UNIX System User and Group

System users run specific server processes. Certain privileges need to be designated for these users to ensure they have appropriate permissions for the processes they run. Normally, the **configure** utility creates the following users and groups:

- User: **inetuser**
- Group: **inetgroup**

If the **configure** utility does not create a UNIX user and group for Instant Messaging Server, you need to create them manually as described in this section. After you create the user and group, set permissions appropriately for the directories and files owned by that user.

Do not choose **root** as a server user ID.

To create the appropriate UNIX user and group:

1. Log in as superuser (**root**).
2. Create a group to which your system user belongs. For example, to create a group named **imgroup** on an Oracle Solaris platform, enter the following command:

   ```
   groupadd imgroup
   ```

3. Create the system user and associate it with the group you just created. In addition, set the password for that user. For example, to create a user named **imuser** and associate it with the group **imgroup** on an Oracle Solaris platform, enter the following command:

   ```
   useradd -g imgroup imuser
   ```

   For more information on adding users and groups, refer to your operating system documentation.
4. Examine the **/etc/groups** file to ensure that the user and group were added.

Running the configure Utility

After you install Instant Messaging Server, use the **configure** utility to configure the software and to generate the configuration files.

This section contains the following topics:

Configuring Instant Messaging Server After Installation

1. Change to the *InstantMessaging_home***/sbin** directory. By default, the *InstantMessaging_home* directory is **/opt/sun/comms/im**.

2. Use one of the following options to run the **configure** utility.

   - Command-line:

     ```
     configure --nodisplay
     ```

   - From a state file:

     ```
     configure --silent statefile
     ```

   where *statefile* is the path to the state file you want to use. If you are configuring by using a state file and using the **--silent** option, you are not be prompted for configuration information. Instead, the values from the state file are used to configure the software. See "Performing a Silent Instant Messaging Server Configuration" for information on generating a state file. If you are not performing a silent installation, a series of prompts appears, requesting information that sets up the initial configuration for Instant Messaging Server. The prompts that appear vary depending on the components you installed. Fill in the requested information using the values from your Instant Messaging Server checklist. See "Configuring Instant Messaging Server".

   Note: Do not select the Disable Server option.

3. To use the XMPP/HTTP Gateway, modify the location of the default log file for the XMPP/HTTP Gateway in the **httpbind_log4j.conf** file.

   You need to do this on Linux, however, you only need to do this on Oracle Solaris if you chose to use a location for logs other than the default.

   To do this:

   a. Open the **httpbind_log4j.conf** file. This file is stored at the location you specified in the **httpbind.conf** file as the value for the **httpbind.log4j.config** parameter. By default the file is stored in the following directory under the default Instant Messaging Server instance: *InstantMessaging_cfg_home*/**httpbind_log4j.conf**.

   b. Set the value of the **log4.appender.appender_ID.file** parameter to the location where log files are stored. By default, on Red Hat Linux and Oracle Linux, this value is **/var/opt/sun/im/default/log**. If you chose another location for log files when you ran configure, enter that path as the value for the parameter.

   c. Configure client systems to support Instant Messaging Server.

   Note: If **httpbind** and **im** are configured to run on different hosts, you must explicitly add the **c2s** protocol to the **s2s** listener by using the **imconfutil** command to set the **set-listener-prop** property. This is common for all components, and not just **httpbind**. If **httpbind** or any other component is enabled on the same machine during **im** configuration, this step is not required, as it is automatically carried out by the **configure** utility.

Performing a Silent Instant Messaging Server Configuration

To run a silent configuration, you first complete a false configuration to create a state file. During this false configuration session, your responses to the **configure** utility are captured in the state file, but no software is modified. In the state file, your responses are retained as a list of parameters, each representing a single prompt or field.

You can then run the **configure** utility on many hosts by using the state file as input. This process enables you to quickly propagate one configuration across multiple hosts in your enterprise. See "Instant Messaging Server Configuration Script" for information on using the state file to configure a new instance of Instant Messaging Server.

To generate a state file:

1. Log in as superuser (**root**).
2. Change to the *InstantMessaging_home***/sbin** directory. By default, the *InstantMessaging_home* directory is **/opt/sun/comms/im**.
3. Run the **configure** utility by entering the following:

   ```
   configure --no --nodisplay --saveState statefile
   ```

   Where *statefile* is the name you want to use for the state file. To use the state file to configure a different installation of Instant Messaging Server, use the following command:

   ```
   configure --nodisplay --silent statefile
   ```

   As you proceed through the **configure** utility, your answers are captured in the state file. When you complete the configuration, the state file is available in the location that you specified.

Examples of the configure Utility

This section provides **configure** utility examples.

- To configure and save the inputs that you provide in the state file:

  ```
  configure --nodisplay --savestate /tmp/imstate
  ```

- To configure and use the values from the state file:

  ```
  configure --nodisplay --state /tmp/imsilent
  ```

- To configure through the silent mode and use the values from the state file:

  ```
  configure --silent statefile
  ```

- To configure and use the values from the state file:

  ```
  configure --nodisplay --state /tmp/imsilent --savestate /tmp/imstate --no
  ```

  The command saves a state file. It does not perform the actual configuration as the **--no** option is used.

- To generate a zip state file:

  ```
  configure --nodisplay --savestate /tmp/imnew.zip
  ```

- To configure and use the values from the zip state file:

  ```
  configure --nodisplay --state /tmp/imnew.zip
  ```

- To configure through silent mode and use the values from the zip state file:

  ```
  configure --silent imnew.zip
  ```

- To use a savestate file with encrypted passwords, first create the plaintext savestate file, change the passwords, then generate the passwords by using the **imconfutil generate-password** command.

Sample configure Utility Configuration Responses

The following shows a sample configuration in response to prompts presented by the **configure** utility. The configuration uses default values for all options.

- Component Selection

```
Select all components you wish to configure.

1. [X] Server components
2. [X] Web components
```

- Service Runtime Options

```
Runtime User ID : [inetuser]
Runtime Group ID: [inetgroup]
Runtime Directory [/var/opt/sun/comms/im]
```

- Network Access Points

```
Domain Name example.com
XMPP Port [5222]
Multiplexed XMPP Port [45222]
Gateway Connector Port [55222]
XMPP Server Port [5269]
Disable Server (enable only multiplexor) [no]
Enable SSL [yes]:
Server keystore file:
Server password file:
```

  If you decide to enable SSL, the respective server configuration is mandatorily set to TLS for all communication. To disable mandating TLS, set **iim_server.requiressl=false** by using the **imconfutil** command.

- LDAP Configuration

```
LDAP Host Name [imhost.siroe.com]
LDAP Port Number [389]
SSL Enabled [no]
Base DN [dc=siroe,dc=com]
Base DN cn=Directory Manager
Base Password
```

- Mail Server Options

```
Enable Email Integration [yes]
SMTP Server [smtphost]
Enable Email Archiving [yes]
```

- HTTP Gateway Deployment Configuration

```
Deploy IM HTTP Gateway [yes]
Context Root [http://imhost:80/httpbind]
Web Container Path [Web container base directory]
Web Administrator URL [ ]
Web Administrator User ID [admin]
Web Administrator Password
```

- Calendar Agent configuration

```
Enable Calendar Agent [no]
Enable local component [no]
```

- SMS Gateway Configuration

```
Enable SMS Gateway [no]
Enable local component [no]
```

- Instant Messaging Server Services Startup

```
Start Services After Successful Configuration [yes]
Start Services When System starts [yes]
```

Creating Multiple Instances from a Single Instant Messaging Server Installation

You can create multiple instances of Instant Messaging Server on a single host from one installation. You might want to do this to create a secure version of Instant Messaging Server, or to support multiple directory namespaces. A namespace is a node in the directory under which each UID is unique. All instances of Instant Messaging Server on a single host share binaries but have unique versions of runtime and configuration files.

To Create an Additional Instance of Instant Messaging Server

This procedure assumes that you have used default installation and configuration values for the *InstantMessaging_home* and *InstantMessaging_runtime* directories. If you installed using default values, the original runtime directory for Oracle Solaris is:

```
/var/opt/sun/comms/im/default
```

For Red Hat Linux and Oracle Linux, the original runtime directory is:

```
/var/opt/sun/im/default
```

If you used paths other than the defaults, substitute your paths for the paths used in this procedure.

1. Create a runtime directory for the new instance. For example, to create runtime directory **xyz** on Oracle Solaris, type:

   ```
   mkdir /var/opt/sun/comms/im/xyz
   ```

   On Red Hat Linux and Oracle Linux, type:

   ```
   mkdir /var/opt/sun/im/xyz
   ```

2. Create a log directory for the new instance. For example, to create log directory **xyz**, on Oracle Solaris, type:

   ```
   mkdir /var/opt/sun/comms/im/xyz/log
   ```

   On Red Hat Linux and Oracle Linux, type:

   ```
   mkdir /var/opt/sun/im/xyz/log
   ```

3. If you are using a file-based property store for user data, you need to create a database directory (*InstantMessaging_database*) for the new instance. For example, to create database directory **xyz**, on Oracle Solaris, type:

   ```
   mkdir /var/opt/sun/comms/im/xyz/db
   ```

   On Red Hat Linux and Oracle Linux, type:

   ```
   mkdir /var/opt/sun/im/xyz/db
   ```

4. Copy the contents of the *InstantMessaging_home* directory and all of its subdirectories into the newly created directories: For example, on Oracle Solaris, type:

   ```
   cp -r /etc/opt/sun/comms/im/default /etc/opt/sun/comms/im/xyz
   ```

   On Red Hat Linux and Oracle Linux, type:

   ```
   cp -r /etc/opt/sun/im/default /etc/opt/sun/im/xyz
   ```

5. Open the new instance's **imadmin** command in a text editor. By default, this command is stored under the *InstantMessaging_home* directory you just created for the new instance. For Oracle Solaris, the location is:

   ```
   /etc/opt/sun/comms/im/xyz/imadmin
   ```

   For Red Hat Linux and Oracle Linux, the location is:

   ```
   /etc/opt/sun/im/xyz/imadmin
   ```

6. In the **imadmin** command, change the configuration file path to the path for the new configuration file for the new instance. For example, on Oracle Solaris, change

   ```
   /etc/opt/sun/comms/im/default/config/iim.conf
   ```

   to:

   ```
   /etc/opt/sun/comms/im/xyz/config/iim.conf
   ```

   On Red Hat Linux and Oracle Linux, change

   ```
   /etc/opt/sun/im/default/config/iim.conf
   ```

   to:

   ```
   /etc/opt/sun/im/xyz/config/iim.conf
   ```

   The **.xml** suffix is not required for the **iim.conf** file and the **imadmin** command automatically adds the **.xml** suffix.

7. Save and close the **imadmin** command.

8. Use the **imconfutil** command to set configuration properties for the new instance. By default, the **iim.conf.xml** file is stored in the *InstantMessaging_cfg* directory you created for the new instance. For Oracle Solaris, the location is:

   ```
   /etc/opt/sun/comms/im/xyz/config/iim.conf.xml
   ```

   For Red Hat Linux and Oracle Linux, the location is:

   ```
   /etc/opt/sun/im/xyz/config/iim.conf.xml
   ```

   The configuration properties you need to set are:

   - **iim_server.port** (default=5269)
   - **iim_mux.listenport** (default=5222)
   - **iim_mux.serverport** (default=45222)
   - **iim.instancedir**

     Set to the runtime directory for the new instance, for example, on Oracle Solaris, change

     ```
     /var/opt/sun/comms/im/default
     ```

to:

```
/var/opt/sun/comms/im/xyz
```

On Red Hat Linux and Oracle Linux, change

```
/var/opt/sun/im/default
```

to:

```
/var/opt/sun/im/xyz
```

9. Ensure that file and directory ownership and permissions are the same for all instances.
10. Start the new instance.
    - On Solaris OS:

      ```
      /etc/opt/sun/comms/im/xyz/imadmin start
      ```

    - On Red Hat Linux and Oracle Linux:

      ```
      /etc/opt/sun/im/xyz/imadmin start
      ```

6

Upgrading Instant Messaging Server

This chapter explains how to upgrade your existing system to the latest release of Oracle Communications Instant Messaging Server.

About Upgrading Instant Messaging Server

The process for upgrading the Instant Messaging server and multiplexor is the same. The upgrade procedure automatically copies the pre-upgrade release product configuration and other data to the post-upgrade version. If Instant Messaging Server is configured to provide email notifications, or calendar alerts, the configuration data of these features is migrated to the post-upgrade version.

If your deployment consists of multiple Instant Messaging Server nodes, you can complete the upgrade in a "rolling" fashion by upgrading one node at a time. For example, if your deployment consists of six Instant Messaging Server nodes, you would upgrade the first node while the other five nodes were up and running, then upgrade the second node, and so on.

Supported Upgrade Paths

You can upgrade from release 9.0.x or 10.0 to 10.0.x

If you are not yet running a version of Instant Messaging Server 9, see "Upgrading Instant Messaging Server (Prior to Version 9 to 10.0.x)".

> **Note:** Java is no longer bundled with the Instant Messaging Server installer and requires manual installation. It is important that you install the correct version of Java for Instant Messaging Server. See "Installing Java"for information.

Upgrading Instant Messaging Server (9.0.x or 10.0 to 10.0.x)

To upgrade to 10.0.x:

1. Download the Instant Messaging Server software for your operating system and extract the files to a temporary directory on your Instant Messaging Server hosts.

   See "Downloading the Instant Messaging Server Software" for more information.

2. Stop Instant Messaging Server.

   ```
   imadmin stop
   ```

3. Run the **commpkg upgrade** command.

For more information, see "upgrade Verb Syntax".

4. Select the Instant Messaging Server 10 component from the Product Selection list.
5. Respond to the prompts to upgrade.
6. (Optional) If you have deployed any Web applications, redeploy them.

   ```
   iwadmin redeploy app_name
   ```

   where *app_name* can be **im**, **httpbind**, or **all**.

   This step completes the upgrade process and redeploys the specified component(s).
7. If you had previously configured your deployment to use Service Management Facility (SMF), run the following command to enable SMF, as the upgrade does not preserve SMF status.

   ```
   imadmin smf-register
   ```

8. When upgrading to Instant Messaging Server 10.0.x from a release prior to 9.0.1.4.0, check if you are using any of the following items and do not want to reconfigure:

   - Components: Calendar Agent, HTTPBind, and so on
   - S2S Federation
   - Server Pool
   - Server health monitoring using Watchdog

   If you use any of these items, and if you continue to use the existing configuration file and do not want to reconfigure, then you must set the **iim_server.useport** property to **true** by using the **imconfutil** command. For example:

   ```
   imconfutil -c InstantMessaging_home/config set-prop iim_server.useport=true
   ```

9. Uninstall the old IMAPI package/rpm:

   ```
   commpkg uninstall --components IMAPI
   ```

10. For "rolling upgrades" (deployments of multiple Instant Messaging Server hosts), disable server-to-server ping support.

    ```
    InstantMessaging_home/sbin/imconfutil -u -c InstantMessaging_
    home/config/iim.conf.xml set-prop iim_server.peerpingtimeout=-1
    ```

11. Install Java 8 and point the Java 8 installation directory to the **/usr/jdk/latest** directory.

    Note: See Table 3–2, " Instant Messaging Server Software Requirements" for more information on web containers and Java requirements.

12. If you are using inter-domain communication, the upgrade process disables communication between domains, which is the default setting. To enable inter-domain communication, set the **iim_server.hosteddomains.allowcrossdomainsaccess** property to **`true`**.
13. Restart Instant Messaging Server.

```
imadmin start
```

Upgrading Instant Messaging Server (Prior to Version 9 to 10.0.x)

Upgrading to Instant Messaging Server 10.0 from an Instant Messaging Server release prior to version 9 is a two step process. You must first upgrade to Instant Messaging Server 9. Then, you upgrade to Instant Messaging Server 10.0.x.

To upgrade from a version prior to Instant Messaging Server 9:

1. To upgrade to Instant Messaging Server 9, see the *Unified Communications Suite 7 Update 2 Installation and Configuration Guide*.
2. Run the following commands for the Instant Messaging Server 8 parameters that did not get migrated:

   ```
   imconfutil set-listener-prop -u -c InstantMessaging_home/config/iim.conf.xml
   c2s port=45222 worker-out=muxout
   worker-in=muxin protocols=c2s
   imconfutil set-prop -u -c InstantMessaging_home/config/iim.conf.xml iim_
   server.deliverofflinechat=true
   iim_mux.jvm.maxmemorysize=2048 iim_ldap.conferencecontainer="ou=sunConferences"
   iim.policy.cachevalidity=3600
   ```

3. To upgrade from Instant Messaging Server 9 to Instant Messaging Server 10.0, see "Upgrading Instant Messaging Server (9.0.x or 10.0 to 10.0.x)".

Post-Upgrade Tasks

See known problems in *Instant Messaging Server Release Notes* for post-upgrade tasks that might be necessary.

Upgrading from 9.0.x to 10.0.x in a Highly Available Environment

Upgrading Instant Messaging Server in an HA environment consists of upgrading the Instant Messaging Server software followed by upgrading the Instant Messaging Server Sun Cluster Agent.

To Upgrade to Instant Messaging Server 10.0.x in an HA Environment

1. Disable the Instant Messaging Server resource.

   ```
   scswitch -n -j im-server-resource
   ```

2. Make sure that the non-active node does not have access to the configuration directory.
3. Run the **commpkg upgrade** command on all cluster nodes.
4. Copy the Instant Messaging 9 configuration file **iim.conf.xml** to the **iim.conf** file with the same permissions.

   Also copy the **iim.conf.xml** file to **iim.conf** after any future configuration changes as cluster uses the **iim.conf** file.
5. To use the new 'GatewayConnector' service in HA, update this service configuration with the virtual host name or IP address and port number as follows:

   ```
   imconfutil --config config_file set-prop iim_gwc.hostport=virtual host-name or
   ip:port
   ```

For example:

```
imconfutil --config /DATA1/default/config/iim.conf.xml set-prop iim_
gwc.hostport=192.10.12.11:22222
```

6. Enable the Instant Messaging server resource.

```
scswitch -e -j im-server-resource
```

To Upgrade to Instant Messaging Server 10.0.x Sun Cluster Agent (IM_SCHA)

Run the **commpkg upgrade** command on all nodes on the cluster. If cluster node is a non-global zone, run **commpkg upgrade** in global zone as well as in non-global zones.

Rolling Back an Upgrade

If the upgrade fails or if you need to go back to the previously working version of Instant Messaging Server, you can roll back the upgrade process.

To roll back the upgrade process:

1. Stop all services.

```
imadmin stop
```

2. Remove the Instant Messaging Server 10 packages.

   For example, on Solaris, use the **pkgrm** command.

3. Install a prior version of Instant Messaging Server by using the installer for that software's version.

7

Uninstalling Instant Messaging Server

This chapter describes how to uninstall Oracle Communications Instant Messaging Server.

Uninstalling Instant Messaging Server

The **commpkg uninstall** command enables you to uninstall Instant Messaging Server. However, the **commpkg uninstall** command does not remove OS patches or shared components installed by **commpkg install**.

To uninstall Instant Messaging Server:

1. Log in as **root**.
2. Change to the *InstantMessaging_home*/**CommsInstaller/bin** directory.
3. Run the **commpkg uninstall** command.
4. Choose Instant Messaging Server from the list of installed Communications Suite components.
5. When prompted, enter **Yes** to continue.
6. Uninstall the IMAPI package/rpm:

```
commpkg uninstall --components IMAPI
```

8

Installing Patches

This chapter describes how to install patches on Oracle Communications Instant Messaging Server.

See the patch ReadMe file, included in the patch download, for information about the contents of a patch.

About Patching Instant Messaging Server

Instant Messaging Server patches are posted on the My Oracle Support web site:

https://support.oracle.com

To install patches, use the **patchadd** command on Solaris OS, and use the **rpm** command on Linux.

> **Important:** Always read the patch ReadMe file in its entirety before installing a patch.

Some patches contain fixes and functionality that may not be of any interest to you or may apply to features that you have not installed or purchased. Read the patch ReadMe file to determine if you must install the patch.

Some patches are password protected. To request the password to download a protected patch, open a Service Request on the My Oracle Support web site.

Planning Your Patch Installation

Before installing a patch, verify your version of Instant Messaging Server and ensure the patch is not already installed.

Oracle recommends scheduling your patch installation during non-peak hours to minimize the disruption to your operations.

Oracle recommends installing a patch on a test system with a copy of your production data before installing the patch on your production system. Test the patch by logging into Instant Messaging Server and verifying the version number of installed components.

Installing a Patch

Oracle Solaris 11 introduced the Image Packaging System (IPS) for software installs and updates. IPS changes the way Unified Communications Suite delivers patches,

because IPS does not support the **patchadd** command. On Solaris 11 systems, you must use Automated Release Update (ARU) patches. These patches differ from the older SRV4 Sun-style patches, which are not supported on Solaris 11. You can use ARU patches on other Solaris releases as well. To install a Unified Communications Suite ARU patch, you use the **commpkg upgrade** command.

Installing an ARU Patch

To install an ARU patch on Instant Messaging Server:

1. Stop Instant Messaging Server services.

   ```
   imadmin stop
   ```

2. Apply the patch by running the following command.

   ```
   commpkg upgrade
   ```

Installing an SRV4 Patch

To install an SRV4-style patch on Instant Messaging Server:

1. Stop Instant Messaging Server services.

   ```
   imadmin stop
   ```

2. Apply the patch by running the **patchadd** command.

   See the **patchadd** man page for more information.

Installing a Linux Patch

To install a Linux patch on Instant Messaging Server:

1. Stop Instant Messaging Server services.

   ```
   imadmin stop
   ```

2. Apply the patch by running the **rpm -F** *rpmname* command.

   See the **rpm** man page for more information.

A

commpkg Reference

This appendix provides information about the Oracle Communications Instant Messaging Server **commpkg** command.

Overview of the commpkg Command

The **commpkg**, command, also referred to as the installer, comprises several commands (verbs) that enable you to install, uninstall, and upgrade Instant Messaging Server software and its shared components. The **commpkg** command is installed in the directory in which you extract the product software.

Syntax

commpkg [*general_options*] *verb* [*verb_options*]

Table A–1 describes the **commpkg** command general options.

Table A–1 commpkg Command General Options

Option	Description
-? or **--help**	Displays help.
-V or **--version**	Displays the installer version.
--OSversionOverride	Overrides the operating-system version check.
--fixEntsys [**y** | **n**]	Fixes an invalid Sun Java Enterprise System (Java ES) **entsys symlink**, making the link point to the latest Java version upgraded by **commpkg**. The Java ES symlink is located in **/usr/jdk/entsys-j2se**. Choose **--fixEntsys y** to fix the Java ES symlink to the Java files. If you do not specify this switch, **commpkg** prompts you if the symlink is invalid. However, in silent mode, the default is not to fix the symlink (the equivalent of using a value of n). To fix the symlink in silent mode, type **commpkg install --fixEntsys y --silent** *INPUTFILE* on the command-line.

Table A–2 describes the **commpkg** command verbs.

Table A–2 commpkg Command Verbs

Verb	Description
install	Performs software installation.
uninstall	Uninstalls software but does not remove OS patches or shared components installed by **commpkg install**.

Table A–2 (Cont.) commpkg Command Verbs

Verb	Description
info	Displays product information on the paths (also known as *installroots*) where Instant Messaging Server is installed, and the products that are installed in those paths.
upgrade	Performs software upgrade.
verify	Verifies installed product.

install Verb Syntax

commpkg install [*install_options*] [*ALTROOT*|*name*]

> **Tip:** Installing Only Shared Components: To install just the product's shared components, launch the installer then prefix your product selection with a tilde (~). You can type multiple selections by using a comma to separate the entries.

Table A–3 describes the **commpkg install** verb options.

Table A–3 commpkg install Options

commpkg install Options	Description
-? or **--help**	Displays help.
-V or **--version**	Displays the installer version.
--excludeOS	Does not apply operating system patches during product installation.
--excludeSC	Does not install, upgrade, or patch any shared components.
ALTROOT | *name*	Use this option to install multiple instances of the product on the same host or Oracle Solaris zone. You can also use this option to perform a side-by-side upgrade of the product. This option is available on Solaris only. Specifies an alternate root directory for a multi-instance installation. The *InstallRoot* (the top-level installation directory for all products and shared components) is the alternate root. If you specify a *name*, it will be a friendly name associated with the *ALTROOT* that is registered in the software list. If you specify the *name* and it exists in the software list, the corresponding *ALTROOT* is used. If you also specify the **--installroot** option, it must correspond to the entry in the software list. If you specify name and it does not exist in the software list, it is added to the software list. Specifying any *name* other than "" implies an **ALTROOT**. A value for *name* of "" is reserved for the default root.
--installroot	Specify location of INSTALLROOT, the top level installation directory for all products and shared components. The top-level installation directory for individual products are subdirectories under INSTALLROOT. Default is **/opt/sun/comms**.
--distro *path*	Specifies the *path* to packages or patches for the products. Default: Location of **commpkg** script

Table A–3 (Cont.) commpkg install Options

commpkg install Options	Description
--silent *INPUTFILE*	Runs a silent installation, taking the inputs from the *INPUTFILE* and the command-line arguments. The command-line arguments override entries in the *INPUTFILE*. Installation proceeds without interactive prompts. Use **--dry-run** to test a silent installation without actually installing the software. Specify **NONE** for *INPUTFILE* to run in silent mode without using an input file. When you specify **NONE**, the installation uses default values.
--dry-run or **-n**	Does not install software. Performs checks.
--upgradeSC [**y**|**n**]	Upgrades or does not upgrade shared components as required. If this option is not specified, you are prompted for each shared component that must be upgraded by using package removal and installation. Default: **n** **Caution:** Upgrading shared components by using package removal and installation is irreversible. However, if you do not upgrade required shared components, products might not work as designed. The **--excludeSC** flag has precedence over this flag.
--auditDistro	Audits the installation distribution to verify that the patches and packages matches the versions in the product files internal to the installer.
--pkgOverwrite	Overwrites the existing installation package. You might use this option when you are installing a shared component in a global zone where either the shared component does not exist in a global zone, or the shared component exists in the whole root zone. The default is not to override the existing package. In general, shared components should be managed in the global zone.
--components *comp1 comp2...*	A space delimited set of component products. Each product has mnemonic associated with it. Use **commpkg info --listPackages** to see the mnemonic for a product. In most shells you must escape the space between each mnemonic, that is, by adding double quotes around all the components.
--skipOSLevelCheck	(Solaris only) The installer always checks that the operating system is at a certain minimum patch level. Using this option skips the check.

uninstall Verb Syntax

commpkg uninstall [*verb_options*] [*ALTROOT*|*name*]

Table A–4 describes the **commpkg uninstall** verb options.

Note: A fast way to uninstall a product that was installed in an alternate root is to simply remove the entire alternate root directory.

Table A–4 commpkg uninstall Options

commpkg install Options	Description
-? or **--help**	Displays help.
-V or **--version**	Displays the installer version.
--silent *INPUTFILE*	Runs a silent uninstall, taking the inputs from the *INPUTFILE* and the command-line arguments. The command-line arguments override entries in the *INPUTFILE*. Uninstall proceeds without interactive prompts. Use **--dry-run** to test a silent installation without actually installing the software.
--dry-run or **-n**	Does not install software. Performs checks.
ALTROOT | *name*	Use this option to uninstall multiple instances of the product on the same host or Oracle Solaris zone. You can also use this option to perform a side-by-side upgrade of the product. This option is available on Solaris only. Specifies an alternate root directory for a multi-instance uninstallation. The *InstallRoot* (the top-level installation directory for all products and shared components) is the alternate root. If you specify a *name*, it will be a friendly name associated with the *ALTROOT* that is registered in the software list. If you specify the *name* and it exists in the software list, the corresponding *ALTROOT* is used. If you also specify the **--installroot** option, it must correspond to the entry in the software list. If you specify name and it does not exist in the software list, it is added to the software list. Specifying any *name* other than "" implies an **ALTROOT**. A value for *name* of "" is reserved for the default root.

upgrade Verb Syntax

commpkg upgrade [*verb_options*] [*ALTROOT*|*name*]

Table A–5 describes the **commpkg upgrade** verb options.

Table A–5 commpkg upgrade Options

Options	Description
-? or **--help**	Displays help.
-V or **--version**	Displays the installer version.
--excludeOS	Does not apply operating system patches during product upgrade.
--excludeSC	Does not install, upgrade, or patch any shared components.

Table A–5 (Cont.) commpkg upgrade Options

Options	Description
ALTROOT | *name*	This option is available on Solaris only. Specifies an alternate root directory during a multiple host installation. The *InstallRoot* (the top-level installation directory for all products and shared components) is the alternate root. If you specify a *name*, it is an "alias" associated with the alternate root that is registered in the software list. You can use this option to upgrade multiple product instances on the same host or Solaris zone. Additionally, you can use this option to perform a side-by-side product upgrade.
--distro *path*	Specifies the *path* to packages and patches for the products. Default path: Location of the **commpkg** command.
--silent *INPUTFILE*	Runs a silent upgrade, taking the inputs from the *INPUTFILE* and the command-line arguments. The command-line arguments override entries in the *INPUTFILE*. Upgrade proceeds without interactive prompts. Use **--dry-run** to test a silent upgrade without actually installing the software. Specify **NONE** for *INPUTFILE* to run in silent mode without using an input file. When you specify **NONE**, the upgrade uses default values.
--dry-run or **-n**	Does not upgrade software but performs checks. This option creates a silent upgrade file in the **/tmp** directory.
--upgradeSC [**y**|**n**]	Indicates whether to upgrade shared components as required. If this option is not specified, you are prompted for each shared component that must be upgraded by the package uninstall/install. Default: **n** **Caution:** Upgrading shared components is irreversible. However, if you do not upgrade required shared components, products might not work as designed. The **--excludeSC** flag has precedence over this flag.
--pkgOverwrite	This option is only for Solaris systems. Overwrites the existing installation package. You might use this option when you are installing a shared component in a global zone where either the shared component does not exist in a global zone, or the shared component exists in the whole root zone. The default is not to override the existing package. In general, shared components should be managed in the global zone.
--components *comp1 comp2...*	Specifies products to be upgraded. Separate each component product with a space. (Thus, the list is a space-delimited set.) To specify each component product, use the mnemonic associated with that product. To display a list of the mnemonics for all the component products, use the **commpkg info --listpackages** command.
--usePkgUpgrade	If the upgrade can be performed by using a patch or an in-place package upgrade, this option uses the in-place package upgrade. The default is to use a patch to upgrade, if possible. **Note:** Patches are used only on Solaris.

verify Verb Syntax

commpkg verify [*verb_options*] [*ALTROOT*|*name*]

Tip: When verifying the package installation in an alternate root, be aware that Instant Messaging Server uses the operating system components installed in the default root. Some products might also use shared components in the default root. Thus, verify the package installation in the default root as well.

Table A–6 describes the **commpkg verify** verb options:

Table A–6 commpkg verify Options

Options	Description
-? or **--help**	Displays help.
-V or **--version**	Displays the installer version.
--excludeOS	Do not verify operating system components.
--excludeSC	Do not verify shared components.
--components *comp1 comp2...*	A space delimited set of component products. Each product has mnemonic associated with it. Use **commpkg info --listPackages** to see the mnemonic for a product. In most shells you must escape the space between each mnemonic, that is, by adding double quotes around all the components.
ALTROOT | *name*	Use this option to verify multiple instances of the product on the same host or Solaris zone. This option is available on Solaris only. Specify *ALTROOT* or *name* for an alternate root directory on which to perform the package verification.

info Verb Syntax

commpkg info [*verb_options*] [*ALTROOT*|*name*]

Table A–7 describes the **commpkg info** verb options.

Table A–7 commpkg info Options

Options	Description
-? or **--help**	Displays help.
-V or **--version**	Displays the installer version.
--clean	Removes entries in the software list. If *ALTROOT* | *name* is specified, the option removes the entry from the software list. If no *ALTROOT* | *name* is specified, the option removes all entries from the software list.
--listPackages	Lists the packages that comprise Instant Messaging Server, shared components, and operating system auxiliary products. This option also displays the mnemonic for Instant Messaging Server and components such as **comm_dssetup.pl**.
--verbose	Prints product information installed in the *installroots*. To print information for a specific *installroot*, run the following command: **commpkg info --verbose** *installroot*

Table A–7 (Cont.) commpkg info Options

Options	Description
--components *comp1 comp2...*	A space delimited set of component products. Each product has mnemonic associated with it. Use **commpkg info --listPackages** to see the mnemonic for a product. In most shells you must escape the space between each mnemonic, that is, by adding double quotes around all the components.

About the Alternate Root

You can install multiple copies of the same product version on the same Solaris machine or Solaris zone by using the alternate root feature of the **commpkg install** command. For example, you might deploy a host with an installation in the default root directory, **/opt/sun/comms**, and a second, separate installation in the **/opt/sun/comms2** alternate root directory. The alternate root installation directory is the top-level directory underneath which the Instant Messaging Server component product and shared components are installed in their respective directories.

Some possible uses for multiple installations include:

1. Performing a side-by-side upgrade.
2. Enabling an installation to be easily moved from one machine to another.

> **Note:** The alternate root feature is available only on Solaris. This feature is a "power user" feature. If you are interested in installing more than one instance of the same version of Instant Messaging Server on the same physical host, another option is to use Solaris zones. For more information, see "Installing on Solaris OS Zones: Best Practices".

ALTROOT | name Syntax and Examples

You can use the optional *ALTROOT | name* option with the **commpkg install**, **commpkg upgrade**, **commpkg uninstall**, and **commpkg verify** commands. You use either *ALTROOT* or *name*. The *name* acts as an alias for the alternate root installation path. The *name* is registered in an internal software list maintained by the installer. You can use *name* for the alternate root's path in commands that accept the alternate root. The distinction between the alternate root and name is that the alternate root always begins with a slash (/) whereas the name does not.

Syntax:

commpkg [**install**|**upgrade**|**unistall**|**verify**] [*ALTROOT*|*name*]

Example 1:

```
commpkg install /opt/sun/comms2
```

In this example, the path **/opt/sun/comms2** is the alternate root, which becomes the top-level directory underneath which Instant Messaging Server software and shared components are installed.

Example 2:

```
commpkg install Comms2
```

In this example, **Comms2** is the name for the alternate root. During the installation process, the installer prompts you to type in the alternate root installation directory.

Example 3:

In this example, you avoid installing the shared components in the alternate root by using the **--excludeSC** option:

```
commpkg install --excludeSC /opt/sun/comms2
```

Example 4:

To install only the shared components, run the **commpkg install** command and select the product you want to install, but prepend a tilde (~).

For example, if you plan to install Instant Messaging Server in the alternate root, you select ~1 during the default installation. This tells the installer to install the dependencies but not the product itself.

Notes on the *ALTROOT | name* command-line argument:

- Specifying a slash (/) as an alternate root is the same as specifying the default root installation directory. That is, specifying a slash is interpreted by the installer as having specified no alternate root.
- Likewise, specifying "" as an alternate root is interpreted as having specified no alternate root. (The "friendly name" for the default alternate root is "".)
- If you specify the **--installroot** option in addition to *ALTROOT | name*, the two must match.
- Operating system patches are always installed into the default root (/). Some shared components are installed into the *ALTROOT* and some are installed into the default root (/).
- You can quickly uninstall an *ALTROOT* installation by using the **rm -r** command on the alternate root directory. The next time that you run the **commpkg info** command, the internal software list that maintains the alternate root information is updated.
- You can create as many alternate roots as you like. Running the **commpkg info** command reports on the various alternate roots.

Understanding the Difference Between ALTROOT and INSTALLROOT

The following concepts define an alternate root (ALTROOT):

- An alternate root directory is a Solaris feature that is used by the **commpkg** command to enable multiple product installations on the same host.
- The default alternate root is the standard root directory (/) and is always present.

The following concepts define an installation root (*InstallRoot*):

- An *InstallRoot* is the top-level umbrella installation path for Instant Messaging Server.
- On the default alternate root (that is, /), you can specify an *InstallRoot*.
- On an alternate root, the *InstallRoot* is the same as the alternate root.

Default Root

If you use the default root, the default *InstallRoot* is:

/opt/sun/comms/

Using Both Default Root and Alternate Root

Suppose you want to install one instance of Instant Messaging Server in the **/opt/sun/mycompany/comms/** directory, and another instance of the same product in the **/opt/sun/mycompany/comms2/** directory. You would use the following commands:

For the default root:

```
commpkg install --installroot /opt/sun/mycompany/comms
```

For the alternate root:

```
commpkg install /opt/sun/mycompany/comms2/
```

Running Multiple Installations of the Same Product on One Host: Conflicting Ports

By default, after you initially configure the product on alternate roots, the ports used by the different product installations are the same and thus conflict with each other.

This is not a problem if you install multiple installations of the same product on the same host but only intend to have one instance running at one time. For example, you may perform a side-by-side upgrade scenario in which you plan to stop the old instance before you start the new instance.

However, you may plan to test the new instance while the old instance is still running (and supporting end users). In this scenario, the ports are used simultaneously, and you must take steps to resolve the port conflicts.

B

comm_dssetup.pl Reference

This appendix provides information about the Oracle Communications Instant Messaging Server **comm_dssetup.pl** script. You must prepare your Oracle Directory Server Enterprise Edition (Directory Server) hosts by running the **comm_dssetup.pl** before you install and configure Instant Messaging Server.

About the comm_dssetup.pl Script

This section provides information you need to understand before running the **comm_dssetup.pl** script.

The **comm_dssetup.pl** script performs the following steps:

1. Prompts you for your deployment's Directory Server and schema information.

   For a list of the specific information this step requests, see "Information Needed to Run the comm_dssetup.pl Script".

2. Generates a shell script and LDIF file from the information that you supply that is used to modify the Directory Server LDAP.

3. Runs the generated shell script and modifies your Directory Server.

At the end of each step, the **comm_dssetup.pl** script prompts you to continue. No changes are made to the Directory Server LDAP until the last step.

Directory Server Considerations for the comm_dssetup.pl Script

When running the **comm_dssetup.pl** script, consider the following points.

- **comm_dssetup.pl** configures local Directory Server instances, and thus you must:
  - Install the **comm_dssetup.pl** script on every host on which a Directory Server instance resides.
  - Run the **comm_dssetup.pl** script on the same host as your Directory Server. The tool runs locally for a specific instance (specified by path of Directory Server or path of instance).
- You can run the **comm_dssetup.pl** script against any Directory Server instance on the local host. If you have multiple Directory Information Trees (DITs) on one host, you can maintain and update one installation of **comm_dssetup.pl**, and apply it to every Directory Server instance on the host.
- **comm_dssetup.pl** must configure every Directory Server instance for the same DIT. This assumes that:

  - A Directory Server has already been installed, configured, and is running before you launch the **comm_dssetup.pl** script.
  - When adding an additional Directory Server host (such as a replica), at a future date, you must run the **comm_dssetup.pl** script against it, too.

- If you have customized your Directory Server, the following considerations might apply:
  - If you have indexed some attributes, you might have to reindex those attributes after running the **comm_dssetup.pl** script.
  - If you have added other LDIF files (schema definitions), they should not be affected, so no action should be necessary. However, back up your custom schema definition files before running the **comm_dssetup.pl** script.

    The **comm_dssetup.pl** script backs up old schema files to the **/var/tmp/dssetup_timestamp/save** directory.
  - For all Directory Server customizations, including the first two just listed, stop the **comm_dssetup.pl** script after it generates the script and before it actually updates the LDAP directory. Then inspect the script to evaluate how its proposed actions affect your LDAP directory. Take whatever actions you think necessary to protect your customizations before running the script against your Directory Server.

Information Needed to Run the comm_dssetup.pl Script

Table B–1 describes the information about your deployment that you need to gather before running the **comm_dssetup.pl** script.

Table B–1 comm_dssetup.pl Information

Information Item Needed	Default Value
Directory Server root path name	The default depends on the Directory Server version detected. The **comm_dssetup.pl** script does attempt to heuristically determine the value.
Which instance of Directory Server to use? (if more than one)	The default depends on the Directory Server version detected. The **comm_dssetup.pl** script does attempt to heuristically determine the value.
Directory Manager Distinguished Name (DN)	"**cn=Directory Manager**"
Directory Manager's Password	NA
Directory Server being used for user/group data? (yes), or configuration data only? (no)	**yes**

Table B–1 (Cont.) comm_dssetup.pl Information

Information Item Needed	Default Value
User and group root suffix (if yes to previous question)	The default depends on what is detected. The **comm_dssetup.pl** script does attempt to heuristically determine the value.
Schema version? (pick one of the following): ■ 1 - Schema 1 ■ 1.5 - Schema 2 Compatibility Mode ■ 2 - Schema 2 Native Mode For more information on how to choose a schema, see "About the comm_dssetup.pl Script Schema Choices". If you have one version of the schema installed and want to upgrade to a higher level, refer to *Sun Java System Communications Services 6 2005Q4 Schema Migration Guide* before running the script.	**2** However, if you run **comm_dssetup.pl** again, it defaults to the value that you chose the previous time.
If you choose Schema 1 or 1.5, you need a DC tree. If the DC tree does not yet exist, the **comm_dssetup.pl** script creates only the root suffix node, its does not create the rest of the DC tree. You must create the rest of your DC tree yourself.	**o=internet** However, if you run **comm_dssetup.pl** again, it defaults to the value that you chose the previous time.

About the Directory Server Root Path Name and Instance

The **comm_dssetup.pl** script prompts you for both the Directory Server root path and the Directory Server instance. The script then combines these two items into an absolute path name to the Directory Server instance. For example, if your Directory Server instance resides under the **/var/opt/sun/directory/slapd-varrius** directory, then you specify **/var/opt/sun/directory** for the Directory Server root path and **slapd-varrius** for the Directory Server instance.

The reason for having two **comm_dssetup.pl** prompts to specify one absolute path is historical. Prior to Directory Server 6, Directory Server had the concept of a "server root" under which all Directory Server instances (and the Directory Server binaries) resided. After Directory Server 6, the concept of the "server root" was removed. Directory Server instances (and the Directory Server binaries) do not all have to reside under a single umbrella "server root" directory.

About the comm_dssetup.pl Script Schema Choices

Instant Messaging Server supports the following schema choices:

- LDAP Schema 2 native mode

  Corresponds to **comm_dssetup.pl** script schema version choice 2. This is the default for a fresh installation.

- LDAP Schema 1

  Corresponds to the **comm_dssetup.pl** script schema version choice 1.

- LDAP Schema 2 compatibility mode

  Corresponds to **comm_dssetup.pl** script schema version choice 1.5.

About LDAP Schema 2

LDAP Schema 2 is a set of provisioning definitions that describes the types of information that can be stored as entries by using the Directory Server LDAP.

The native mode uses search templates to search the Directory Server LDAP. Once the domain is found by using the domain search template, the user or group search templates are used to find a specific user or group.

You should use native mode if you are installing Instant Messaging Server for the first time and you do not have other applications in your deployment that are dependent on a two-tree provisioning model.

Note: If you have an existing deployment that uses Schema 1, and you want to integrate other Communications Suite products, you should migrate your directory to Schema 2. Refer to *Sun Java System Communications Services 6 2005Q4 Schema Migration Guide* for information on how to migrate from LDAP Schema version 1 to LDAP Schema version 2.

About LDAP Schema 1

LDAP Schema 1 is a provisioning schema that consists of both an Organization Tree and a DC Tree. In Schema 1, when a search is conducted for user or group entries, it looks at the user's or group's domain node in the DC Tree and extracts the value of the **inetDomainBaseDN** attribute. This attribute holds a DN reference to the organization subtree containing the actual user or group entry.

About LDAP Schema 2 Compatibility Mode

Schema 2 compatibility mode is an interim mode between Schema 1 and Schema 2 native mode. Schema 2 compatibility mode supports both schemas and enables you to retain the existing two-tree design you already have.

Use Schema 2 Compatibility if you have existing applications that require Schema 1, but you also need functionality that requires Schema 2.

Note: Schema 2 compatibility mode is provided as a convenience in migrating to the Schema 2 Native mode. Do not use Schema 2 compatibility mode as your final schema choice. The migration process from Schema 1 to Schema 2 compatibility mode and then finally to Schema 2 native mode is more complex that simply migrating from Schema 1 to Schema 2 native mode. See *Sun Java System Communications Services 6 2005Q4 Schema Migration Guide* for more information.

Attribute Indexes Created by the comm_dssetup.pl Script

Attribute indexes improve the performance of search algorithms. The **comm_dssetup.pl** script offers you the choice to index attributes.

Table B–2 lists all the attributes the **comm_dssetup.pl** script indexes, grouped by suffix category. It also lists the type of indexes created for each attribute. For more information about Directory Server indexing, see the Directory Server documentation at:

`http://docs.oracle.com/cd/E20295_01/index.htm`

Table B–2 Attributes Indexed by comm_dssetup.pl

Suffix	Attributes Indexed	Type of Indexes Added
User/Group	**mail**	**pres, eq, approx, sub**
User/Group	**mailAlternateAddress**	**pres, eq, approx, sub**
User/Group	**mailEquivalentAddress**	**pres, eq, approx, sub**
User/Group	**mailUserStatus**	**pres, eq**
User/Group	**member**	**eq**
User/Group	**ou**	**pres**
User/Group	**cosspecifier**	**pres**
User/Group	**groupid**	**pres, eq, sub**
User/Group	**icsCalendar**	**pres, eq, approx, sub**
User/Group	**icsCalendarOwned**	**pres, eq, approx, sub**
User/Group	**uniqueMember**	**eq**
User/Group	**memberOf**	**eq, sub**
User/Group	**cn**	**eq**
User/Group	**mgrpUniqueId**	**eq**
User/Group	**deleted**	**pres, eq**
User/Group	**davuniqueid**	**pres, eq**
User/Group	**inetCos**	**eq**
User/Group	**imUserStatus**	**pres, eq**
User/Group (additional for Schema 2)	**inetDomainBaseDN**	**pres, eq**
User/Group (additional for Schema 2)	**sunPreferredDomain**	**pres, eq**
User/Group (additional for Schema 2)	**associatedDomain**	**pres, eq**
User/Group (additional for Schema 2)	**o**	**pres, eq**
User/Group (additional for Schema 2)	**mailDomainStatus**	**pres, eq**
User/Group (additional for Schema 2)	**sunOrganizationAlias**	**pres, eq**
DC Tree (for Schema 1)	**inetDomainBaseDN**	**pres, eq**
DC Tree (for Schema 1)	**mailDomainStatus**	**pres, eq**
DC Tree (for Schema 1)	**inetCanonicalDomainName**	**pres, eq**
Personal Address Book (PAB) (**o=pab**) Note: For old Address Book	**memberOfManagedGroup**	**pres, eq**
Personal Address Book (PAB) (**o=pab**) Note: For old Address Book	**memberOfPAB**	**pres, eq**

Table B–2 (Cont.) Attributes Indexed by comm_dssetup.pl

Suffix	Attributes Indexed	Type of Indexes Added
Personal Address Book (PAB) (**o=pab**) Note: For old Address Book	**memberOfPABGroup**	**pres**,**eq**
Personal Address Book (PAB) (**o=pab**) Note: For old Address Book	**un**	**eq**
New PAB (**o=PiServerDb**)	**displayname**	**pres**, **eq**, **sub**
New PAB (**o=PiServerDb**)	**MemberOfPiBook**	**eq**
New PAB (o=PiServerDb)	**MemberofPiGroup**	**eq**
o=mlusers for future mailserv feature	**mail**	**eq**
o=mlusers for future mailserv feature	**mlsubListIdentifier**	**eq**
o=mlusers for future mailserv feature	**mlsubMail**	**eq**

To add additional indexes on your own, see the Directory Server documentation.

Running the comm_dssetup.pl Script

You can run the **comm_dssetup.pl** script in either interactive or silent mode. Interactive mode is described in "Running the comm_dssetup.pl Script in Interactive Mode".

Running the comm_dssetup.pl Script in Silent Mode

To run the **comm_dssetup.pl** script in silent mode:

1. On the host where Directory Server is installed, log in as or become the superuser (**root**).
2. Start Directory Server, if necessary.
3. Change to the directory where you installed or copied the Directory Server Setup **comm_dssetup.pl** script.
4. Run the script followed by the silent mode options.

   All options are required. For more information, see "Silent Mode Options".

   ```
   /usr/bin/perl comm_dssetup.pl
   [-i yes|no] [-R yes|no] [-c DirectoryServerRoot]
   [-d DirectoryInstance] [-r DCTreeSuffix]
   [-u UserGroupSuffix] [-s yes|no] [-D DirectoryManagerDN]
   [-j DirectoryManagerPasswordFile] [-b yes|no]
   [-t 1|1.5|2] [-m yes|no] [-S PathtoSchemaFiles]
   ```

   The script creates the following LDIF file and shell script to update the LDAP indexes and schema:

   - **/var/tmp/***dssetup_timestamp***/dssetup.ldif**
   - **/var/tmp/***dssetup_timestamp***/dssetup.sh**

5. If you answered **no** to the **-R** and **-m** options, you must manually run the **comm_dssetup.sh** script that was created.

   If you answered **yes** to the **-R** and **-m** options, the **dssetup.sh script** is run automatically.

Silent Mode Options

Table B–3 table describes the **comm_dssetup.pl** silent mode options.

Table B–3 comm_dssetup.pl Silent Mode Options

Option and Argument	Description
-i yes | no	Specifies whether to configure new indexes. **yes** - Add new Directory Server indexes. **no** - Do not add indexes.
-R yes | no	Specifies whether to reindex automatically. **yes** - Reindex without prompting the user. **no** - Do not reindex without prompting the user. The **-m** option must also be specified for **yes** for the **-R** option to take effect.
-c *DirectoryServerRoot*	Specifies the Directory Server root path, for example, **/var/opt/sun/directory**.
-d *DirectoryInstance*	Specifies the Directory Server instance subdirectory under the Directory Server root path, for example, **slapd-varrius**.
-r *DCTreeSuffix*	Specifies the DC tree root suffix (for Schema 1 and Schema 2 compatibility modes only), for example, **o=internet**.
-u *UserGroupSuffix*	Specifies the user and group root suffix, for example, **o=usergroup**.
-s yes | no	Specifies whether to update the schema. **yes** - Update the schema. **no** - Do not update schema.
-D *DirectoryManagerDN*	Specifies the Directory Manager Distinguished Name (DN), for example, **"cn=Directory Manager"**. The value must be enclosed by double quotation marks (" ") to enable the **comm_dssetup.pl** script to interpret a value with a space correctly.
-j *DirectoryManagerPasswordFile*	Specifies the file containing the Directory Manager DN password.
-b yes | no	Specifies to use this Directory Server for users and groups. **yes** - Use this directory to store both configuration and user group data. **no** - Use this directory to store only configuration data.
-t 1 | 1.5 | 2	Specifies the schema version.
-m yes | no	Specifies whether to modify the Directory Server. **yes** - Modify the Directory Server without prompting the user. **no** - Do not modify the Directory Server without prompting the user.
-S *PathtoSchemaFiles*	Specifies the path to the directory where the schema files are located for example, **./schema**.

C

Instant Messaging Server Configuration Script

This appendix provides information about the Oracle Communications Instant Messaging Server configuration script.

configure Script

The **configure** script enables you to perform an initial configuration of your Instant Messaging Server deployment. Table C–1 describes the **configure** options. The *statefile* can be a simple text file with plaintext passwords, a text file with encrypted passwords, or a zip file, which contains the text of the state file and the decryption keys.

Table C–1 configure Options

Option	Description
--nodisplay	Required if the **--silent** option is not used. Optional if the **--silent** option is used. Use this option to configure Instant Messaging Server in command-line mode.
--help	Optional. Displays the help content for this command.
--verbose	Optional. Prints information messages to the standard output.
--savestate *statefile*	Optional. Should be used with the **--nodisplay** option. If you use this option, the inputs that you provide during configuration are saved in the state file. Specify the name and location of the state file along with this option. Your responses are stored as a list of parameters in the state file. Each parameter represents a single entry or field value.
--silent *statefile*	Required if the **--nodisplay** option is not used. Use this option to run the **configure** script in the silent mode. Specify the name and path of the state file with this option. If you are configuring Instant Messaging Server by using a state file, you are not prompted to specify the configuration information. Instead, the values from the state file are used to configure the server.
--state *statefile*	Optional. During configuration, the **configure** script provides default values for configuration. You can either use the default values or specify your own value. If you use this option, the **configure** script uses all the default values for configuration.
--no	Optional. Use this option to perform a dry run of the configuration.
--novalidate	Optional. If you use this option, the **configure** script does not validate the inputs that you provide during configuration.
--debug	Optional. Use this option to view the debug messages on your terminal.
--key	Optional. Use this option only when in a text-based state file, you have changed passwords byusing the **imconfutil generate-password** command. In that case, the command to be is **configure --silent /tmp/saved-state-file --key /tmp/mykey.txt --nodisplay**.

Note: The **configure** script ignores any incorrect or invalid command-line options and proceeds with the configuration process by using the valid options.

Using the --key Option to Perform a Silent Configuration

To perform a silent configuration by using the **--key** option:

1. Generate the key using **imconfutil generate-key** command, and direct the output to a key file:

   ```
   imconfutil generate-key > /tmp/mykey.txt
   ```

2. Generate the encrypted passwords by using the **generate-password** command, giving the mandatory location of the generated key file:

   ```
   imconfutil generate-password -k /tmp/mykey.txt your-plaintext-password
   ```

3. Change the password in the state file, either manually or though a script.

4. Run the **configure** command, specifying the location of the key file:

   ```
   configure --silent /tmp/saved-state-file --key /tmp/mykey.txt --nodisplay
   --verbose --debug
   ```

   Use the **generate-password** command to encrypt plaintext passwords in the state file. If password encryption is not required, use the state file with plaintext keys, without specifying the **--key** option:

   ```
   configure --silent /tmp/delete/config/saved-state-file-plaintext --nodisplay
   --verbose --debug
   ```

   For more information on the **generate-password** command, see the imconfutil command reference in the *Instant Messaging Server System Administrator's Guide*.