

**Oracle® Communications Instant Messaging
Server**

System Administrator's Guide

Release 8.0

July 2015

ORACLE®

Oracle Communications Instant Messaging Server System Administrator's Guide, Release 8.0

Copyright © 2007, 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

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

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

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

1. Configuring Instant Messaging After Installation	4
2. Setting up and Launching Instant Messenger	21
3. Configuring Instant Messaging for High Availability (Oracle Solaris Only)	24
4. Instant Messaging Configuration File and Directory Structure Overview	49
5. Enabling Single Sign-On (SSO) for Instant Messaging	52
6. Configuring Hosted Domain Support	55
7. Administering Instant Messaging Components	59
8. Federating Deployment of Multiple Instant Messaging Servers	66
9. Optimizing an Instant Messaging Server Pool By Using the Redirect Server	69
10. Scaling an Instant Messaging Deployment By Using Server Pooling	76
11. Using Shoal for Server Pool Messaging	81
12. Securing Instant Messaging Using TLS and Legacy SSL	84
13. Configuring the Voice Chat	94
14. Administering Instant Messaging End Users	96
15. Configuring Instant Messaging and Presence Service Protocol	101
16. Configuring LDAP Failover for Instant Messaging	103
17. Configuring Java Message Service Support for Calendar Server Alerts	105
18. Instant Messaging Server New Features	111
19. SMS Gateway for Instant Messaging	112
20. Multiuser Chat Support for IMPS	116
21. Gateways for AIM, MSN, and Yahoo	119
22. Configure Command	126
23. Managing Archiving for Instant Messaging	133
24. Migrating the Property Store From File to LDAP	148
25. Migrating the Multiplexor Certificate and Enabling SSL	151
26. Managing Instant Messaging's LDAP Access Configuration	153
27. Managing Instant Messaging and Presence Policies	157
28. Managing Instant Messenger	172
29. Managing Logging for Instant Messaging	190
30. Troubleshooting and Monitoring Instant Messaging	200
31. Using Calendar Pop-up Reminders	210
32. Using the Instant Messaging XMPP and HTTP Gateway	216
33. Configuring External Gateways with Instant Messenger	224
34. Switching httpbind from servlet to async Mode	226
35. Writing a Custom SSO Module for Instant Messaging Server	230
36. Performance, Scalability, and Sizing Considerations for Instant Messaging	236
37. Reference Information	241
38. Instant Messaging Configuration Parameters in iim.conf	242
39. Instant Messaging XMPP and HTTP Gateway Configuration Parameters in httpbind.conf	272
40. Instant Messaging imadmin Tool Reference	276
41. Instant Messaging APIs	282
42. Instant Messaging LDAP Schema	284

Chapter 1. Configuring Instant Messaging After Installation

Configuring Oracle Communications Instant Messaging Server After Installation

After you install the Instant Messaging software by using the Communications Suite installer, you must configure the Instant Messaging server and client to complete the installation. You perform this initial runtime configuration by running the Instant Messaging configuration program, `configure`.

Topics:

- [Before Configuring Instant Messaging](#)
- [Completing the Configuration Checklist](#)
- [Creating a UNIX System User and Group](#)
- [configure Utility](#)
- [Adding Instant Messaging and Presence Services to a Sub-organization in Access Manager for Single Sign-On and Policy Management Support](#)
- [Creating Multiple Instances from a Single Instant Messaging Installation](#)

Before Configuring Instant Messaging

Before you install `configure` Instant Messaging, read and understand the information in the [Communications Suite Deployment Planning Guide](#), perform the installation as described in most current Communications Suite Installation Guide, complete the configuration checklist, and finally configure the software. In addition, if you are configuring Instant Messaging with Sun Cluster for High Availability, you need to read [Configuring Instant Messaging for High Availability](#) before completing the steps in this information.

Completing the Configuration Checklist

Gather this information before you begin. You are prompted for some or all of the information depending on the components you installed.

Print out the following table and write the values for your deployment in the space provided. You can reuse this checklist for multiple installations of Instant Messaging. This table contains passwords and other sensitive information, so you should store this information in a safe place.

(Oracle Solaris Only) If you will be configuring High Availability service for Instant Messaging, see [Configuring Instant Messaging for High Availability](#) for specific information about values you can use for these parameters and additional parameters for your checklist.


Table 1-1 Configuration Parameters for Instant Messaging

Parameter	Description	Your Value
-----------	-------------	------------

Installation Directory	<p><i>im-svr-base</i></p> <p>Directory in which Instant Messaging is installed. By default, Instant Messaging is installed into the <code>/opt</code> directory as follows:</p> <p>Oracle Solaris: <code>/opt/sun/comms/im</code></p> <p>Red Hat Linux: <code>/opt/sun/im</code></p> <p>(Oracle Solaris Only) If you will be configuring High Availability service for Instant Messaging, see Selecting the Installation Directory for information about choosing an installation directory.</p>	
Instant Messaging Server Host and Domain Name	<p>Host name on which Instant Messaging is installed and the domain name associated with the host.</p> <p>For example: Host Name: <code>instantmessaging.siroe.com</code></p> <p>Domain Name: <code>siroe.com</code></p> <p>(Oracle Solaris Only) If you will be configuring High Availability service for Instant Messaging, use the logical host name.</p>	
Instant Messaging Server Port Number	<p>The port number on which the Instant Messaging Server listens for incoming requests from the multiplexor.</p> <p>Default: 45222</p>	
Instant Messaging Server-to-Server Port Number	<p>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 Instant Messenger clients on this port.</p> <p>Default: 5269</p>	
Multiplexor Port Number(Multiplexor Configuration Only)	<p>The port number on which the Instant Messaging Server listens for incoming requests from Instant Messenger clients.</p> <p>Default: 5222</p>	
Disable Server	<p>Select this option if the instance you installed will act as a multiplexor and not a server. If you select this option, you must provide a value for Remote Instant Messaging Server Host Name.</p>	
Remote Instant Messaging Server Host Name(Multiplexor Configuration Only)	<p>The host name of the Instant Messaging Server for which this multiplexor routes messages. If the multiplexor and server are installed on the same host, use <code>localhost</code>. (Oracle Solaris Only) If you will be configuring High Availability service for Instant Messaging, use the logical host's name.</p> <p>Dependencies: The Disable Server parameter must be selected, that is, server functionality is disabled.</p>	

Sun Java System Access Manager Configuration	<p>If the <code>configure</code> utility detects that you have installed the Access Manager SDK, you will be prompted to provide answers for the following questions related to Access Manager:</p> <ul style="list-style-type: none"> • Are you planning to leverage an Access Manager deployment for SSO? If you enter <code>yes</code>, <code>configure</code> sets the <code>iim_server.usesso</code> parameter in <code>iim.conf</code> to 1. See Table A-4 for more information about this parameter. • Are you planning to leverage an Access Manager deployment for Policy? If you choose <code>yes</code>, you need to run the <code>imadmin assign_services</code> command when you are finished running the <code>configure</code> utility. See To Configure Instant Messaging After Installation and Assigning Instant Messaging and Presence Services to End Users for more instructions on using the <code>imadmin assign_services</code> command. If you choose <code>no</code>, you will be asked whether you want to store user, conference room, and news channel properties in a file or in LDAP. • In addition, if Instant Messaging will use Access Manager policies in a Sun Java System Application Server deployment, you need to restart the Application Server when you finish configuring Instant Messaging. If you do not restart the Application Server, Instant Messaging services will not appear in the Access Manager console (<code>amconsole</code>). 	
Calendar Server and Calendar Agent Configuration	<p>The <code>configure</code> utility asks if you want to enable the Calendar agent. If you choose to enable the Calendar agent, you need to provide the following information: From the configurator panel:</p> <ul style="list-style-type: none"> • Choose to enable Calendar Agent by typing <code>yes</code>. • Choose to Enable local component by typing <code>yes</code>. • Specify XMPP server hostname. • Specify XMPP server port. • Specify Notification Server Hostname. • Specify Notification Server Port. • Specify Calendar alarm URL. <p>If you choose not to enable the Calendar agent, you can manually configure the Calendar agent later. More information about the Calendar agent configuration parameters and acceptable values is described in Using Calendar Pop-up Reminders.</p>	
Enable Email integration, Enable Email Archiving (Optional)	<p>If selected, enables Instant Messaging email archiving. Dependencies: SMTP Server such as Oracle Communications Messaging Server (formerly Sun Java System Messaging Server).</p>	
LDAP Host Name	<p>In a deployment with an LDAP server, the host name of the LDAP server that contains user and group information for Instant Messaging. For example, <code>directory.siroe.com</code>. Dependencies: LDAP server such as Oracle Directory Server Enterprise Edition (formerly Sun Java System Directory Server Enterprise Edition).</p>	

LDAP Port Number	In a deployment with an LDAP server, the port number on which the directory server listens for incoming requests. For example, 389. Dependencies: LDAP server such as Oracle Directory Server Enterprise Edition (formerly Sun Java System Directory Server Enterprise Edition).	
Base DN	In a deployment with an LDAP server, the base distinguished name in the directory tree that contains user and group information for Instant Messaging. For example, o=airius.com. Dependencies: LDAP server such as Oracle Directory Server Enterprise Edition (formerly Sun Java System Directory Server Enterprise Edition).	
Bind DN	In a deployment with Sun Java System Access Manager, during installation, you must provide the Directory Manager Bind DN and password. This Bind DN is used to update the directory schema with the Instant Messaging and presence service templates and attributes only. This requires Directory Manager access. The Directory Manager Bind DN and password are not saved or used beyond installation and initial configuration. In a deployment with an LDAP server but without Access Manager, Instant Messaging 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 as described in To Configure Bind Credentials for the Instant Messaging Server . Dependencies: LDAP server such as Oracle Directory Server Enterprise Edition (formerly Sun Java System Directory Server Enterprise Edition).	
Bind Password	In a deployment with an LDAP server, the Bind DN password.	
SMTP Server Host Name (Optional)	The host name of the SMTP server used to send email notification of messages to offline users. For example, mail.siroe.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: mail.siroe.com:1025. Dependencies: SMTP server such as Oracle Communications Messaging Server (formerly Sun Java System Messaging Server).	

<p>Database, Logs, and Runtime Files Pathname</p>	<p>The location where the runtime files, database, and logs are stored. Also referred to as <i>im-runtime-base</i>. 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 alert messages, roster information, conferences, news channels, and so on.</p> <p>If you are configuring High Availability (HA) for Instant Messaging, this path must be globally available. See Configuring Instant Messaging for High Availability (Oracle Solaris Only) for more information about HA. The <code>configure</code> utility appends a directory (<code>/default</code>) to the path you provide for the runtime files. The name of this directory is the instance to which the runtime files apply.</p> <p>Later, you can create multiple instances of Instant Messaging by creating additional instance directories with different names (for example <code>/secure</code>) and copying over files from the <code>/default</code> instance runtime directory. See Creating Multiple Instances from a Single Instant Messaging Installation for specific instructions. If you accept the following defaults when you run <code>configure</code>:</p> <p>Oracle Solaris: <code>/var/opt/SUNWiim/</code> Red Hat Linux: <code>/var/opt/sun/im/</code></p> <p>the <code>configure</code> utility creates the following directories for the runtime files:</p> <p>Oracle Solaris: <code>/var/opt/SUNWiim/default</code> Red Hat Linux: <code>/var/opt/sun/im/default</code></p> <p>In addition, the following two subdirectories are created under the runtime directory. The database directory (<i>im-db-base</i>) defaults are as follows:</p> <p>Oracle Solaris: <code>/var/opt/SUNWiim/default/db</code> Red Hat Linux: <code>/var/opt/sun/im/default/db</code></p> <p>The log directory defaults are as follows:</p> <p>Oracle Solaris: <code>/var/opt/SUNWiim/default/log</code> Red Hat Linux: <code>/var/opt/sun/im/default/log</code></p>	
<p>Resources, Help Files, and HTTP Gateway Pathname</p>	<p>Resource Directory. The directory in which the resource files, online help, and the XMPP/HTTP Gateway are installed. If you want to customize the resource files for your deployment, you should run <code>configure</code> utility, customize the files, then redeploy the resource files. You need to run <code>configure</code> first because the <code>configure</code> utility creates some of the index and <code>.jnlp</code> files that you can customize. See Redeploying Resource Files for information.</p> <p>Default: <code>/opt/sun/comms/im/html</code></p>	
<p>XMPP/HTTP Gateway Deployment</p>	<p>Determines whether or not the XMPP/HTTP gateway will be deployed. If you choose to deploy the gateway, the <code>configure</code> utility creates a default gateway configuration file (<code>httpbind.conf</code>) in the default Instant Messaging server instance's <i>im-cfg-base</i> directory if one does not already exist. If <code>httpbind.conf</code> already exists, the <code>configure</code> utility does not alter or overwrite the file.</p> <p>Default: <code>True</code> (gateway is deployed)</p> <div data-bbox="500 1707 1284 1948" style="background-color: #e6f2ff; padding: 10px; border: 1px solid #c0c0c0;"> <p> Note If you are configuring the 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 Convergence server, its value is set when you configure Convergence.</p> </div>	

XMPP/HTTP Gateway URI	Defines the URI for the HTTP component of the XMPP/HTTP gateway. Default: <code>http://<web-svr-host>:80/httpbind</code>	
SMS Gateway	Enables the Instant Messaging server to deliver chat messages and alerts in the form of SMS to the Instant Messaging users who are offline. From the configurator tool panel: 1. Choose the Enable SMS Gateway option by typing <code>yes</code> . 2. Choose the Enable Local Component option by typing <code>yes</code> .	
IMPS Gateway	Enables the Instant Messaging and presence service in mobile devices. When configuring the Instant Messaging server, set the <code>Deploy IM IMPS Gateway</code> option to <code>yes</code> .	
AOL/MSN/Yahoo Gateways	Enables you to communicate with external IM servers. From the configurator panel, choose: 1. Enable Yahoo Gateway: <code>true</code> if you want to enable the gateway on the server. 2. Enable Local Component: <code>true</code> to enable the gateway on the local machine (default port and host name is used). Entering <code>false</code> will allow you to enable it on another machine (need to enter the machine's port and host name).	
Codebase	The URL from which Instant Messenger accesses resources, including the start page for initial downloads of the Instant Messaging client. The installation program installs the resource files into the following locations: Oracle Solaris: <code>/opt/sun/comms/im/html</code> Red Hat Linux: <code>/opt/sun/comms/im/html</code> The <code>configure</code> utility uses the codebase to determine which web container instance to use. If it succeeds, the <code>configure</code> utility deploys the Instant Messenger resources as a web application in the web container, according to the URL provided. If no supported web container is detected, you are prompted for a file system location in which to copy or link the resources. If you are using SSO with Sun Java System Access Manager, the Access Manager server and Instant Messaging server must be configured to use the same web container. See your web container documentation for more information about deploying resource files as a web application. See Changing the Codebase if you need to modify the location of the resource files after initial 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, you need to create them manually as described in this section. After you create the user and group for Instant Messaging, you should then set permissions appropriately for the directories and files owned by that user.

Do not choose `root` as a server user ID unless you are deploying Instant Messaging with Access Manager. In this case, you need to use `root` in order to allow access to the Access Manager configuration.

To Create the Appropriate UNIX User and Group

1. Log in as superuser.
2. Create a group to which your system user will belong.
For example, to create a group named `imgroup` on an Oracle Solaris platform, type the following:

```
groupadd imgroup
```
3. Create the system user and associate it with the group you just created 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, type the following:

```
useradd -g imgroup imuser
```


For more information on adding users and groups, refer to your operating system documentation.
4. Ensure that the user and group have been added to the `/etc/groups` file.

configure Utility

You use the `configure` utility after you install Instant Messaging to configure the software and to generate the configuration files you use to administer Instant Messaging.

This section has the following topics:

- [Overview of the `configure` Utility](#)
- [Syntax and Options of the `configure` Utility](#)
- [Configuring Instant Messaging After Installation](#)
- [Performing a Silent Instant Messaging Configuration](#)
- [Examples of the `configure` Utility](#)
- [Sample Configuration Using the `configure` Utility](#)

Overview of the `configure` Utility

If you want to customize the resource files for your deployment, you should run the `configure` utility, customize the files, then redeploy the resource files. You need to run `configure` first because the `configure` utility creates some of the index and `.jnlp` files that you can customize. See [Redeploying Resource Files](#) for information. Also see [Completing the Configuration Checklist](#) for information on locating these files after configuration.

The utility displays panels that prompt you for information and provide additional instructions for you to configure your Instant Messaging system.

The Instant Messaging software is not configured by the installer. Instead, you need to run the `configure` utility after you install the software.

If you are using the BEA web container, you need to create a PASSFILE before you can configure Instant Messaging. If you are not using the BEA web container, skip to [Configuring Instant Messaging After Installation](#). To create the PASSFILE for the BEA web container, perform the following steps:

1. Create a file named `installation directory/SUNwiim/lib/PASSFILE`.
2. Add the following lines to the file you created:

```
DS_DIRMGR_DN=_Directory Manager Bind DN_  
DS_DIRMGR_PASSWORD=_Directory Manager Bind Password_  
DS_HOST=_LDAP Host Name_  
DS_PORT=_LDAP Port Number_  
DS_BASE_DN=_Base DN_
```

3. Fill in the values for each of the variables.

Syntax and Options of the `configure` Utility

The `configure` utility has the following syntax:

Note
Instant Messaging version 8 release does not support Graphical User Interface based configuration.

```
./configure --nodisplay --silent --savestate --state --verbose --no  
--novalidate --debug --help
```

The `configure` utility has the following options:

Note
The `--id`, `--noconsole`, and `--loglevel` options are removed in Instant Messaging version 8.

`--nodisplay`

Required if the `--silent` option is not used. Optional if the `--silent` option is used. Use this option to configure the Instant Messaging server in the command-line mode.

`--help`

Optional. Displays the help content for this command.

`--verbose`

Optional. Prints information messages to the standard output.

`--savestate <statefilename>`

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 <statefilename>`

Required if the `--nodisplay` option is not used. Use this option to run the `configure` command in the silent mode. Specify the name and path of the state file with this option. If you are configuring the 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 <statefilename>`

Optional. During configuration, the `configure` utility provides default values for configuration. You can either use the default values or specify your own value. If you use this option, the `configure` utility 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` utility does not validate the inputs that you provide during configuration.

`--debug`

Optional. Use this option to view the debug messages on your terminal.



Note

The `configure` utility ignores any incorrect or invalid command-line options and proceeds with the configuration process by using the valid options.

Configuring Instant Messaging After Installation

1. Change to the directory in which you installed Instant Messaging.
By default, this directory is `/opt/sun/comms/im` on the Oracle Solaris platform, and `/opt/sun/comms/im` on the Red Hat Linux platform.
2. Invoke the `configure` utility.
Command-line:

```
configure --nodisplay
```


From a state file:

```
configure --nodisplay --state statefile
```


where *statefile* is the path to the state file you want to use. If you are configuring using a state file, you will 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 Configuration](#) for information on generating a state file.
If you are configuring using the GUI or the command line, a series of prompts appears, requesting information that will set up the initial configuration for Instant Messaging. The prompts that appear vary depending on the components you installed. Fill in the requested information using the values from your Instant Messaging checklist. See [Completing the Configuration Checklist](#).
3. If you install the Sun Java System Access Manager on a different host from the Instant Messaging server, you need to manually copy the `imServices` files from the Instant Messaging server host to the Access Manager host after you run the `configure` utility.
To do this:
 - a. Locate the `imService_*.properties` files on the Instant Messaging server host.
By default, these files are located under `/opt/SUNWiim/lib/` on Oracle Solaris and `/opt/sun/im/lib/` on Red Hat Linux.
 - b. Copy the files to the `locale` directory on the Access Manager host.
By default this directory is `/opt/SUNWam/locale` on Oracle Solaris and `/opt/sun/identity/locale` on Red Hat Linux.
4. If you are using Access Manager to manage Instant Messaging policies, run the `imadmin assign_services` command.

```
imadmin assign_services
```


You will be prompted for the Base DN of the organization under which user entries are stored. This command adds Instant Messaging and presence services to existing users under the organization you specify.
5. Restart Sun Java System Application Server.
If Instant Messaging will use Access Manager policies in a Sun Java System Application Server deployment, you need to restart the Application Server when you finish configuring Instant Messaging. If you do not restart the Application Server, Instant Messaging services will not appear in the Access Manager console (`amconsole`).
6. If you intend to use the XMPP/HTTP Gateway, you may need to modify the location of the default log file for the XMPP/HTTP gateway in `httpbind_log4j.conf` if:
On Oracle Solaris, you chose to use a location for logs other than the default
On Red Hat Linux, regardless of the path you chose
To do this:
 - a. Open the `httpbind_log4j.conf` file.
This file is stored at the location you specified in `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 instance:

```
im-cfg-base/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, 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.

7. If necessary, configure Access Manager-based services for SSO and policy management. See [Adding Instant Messaging and Presence Services to a Sub-organization in Access Manager for Single Sign-On and Policy Management Support](#) for information.
8. Configure the web container and client systems to support Instant Messaging. For instructions, see [Setting up and Launching Instant Messenger](#).

Performing a Silent Instant Messaging 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 using the state file as input. This process allows you to quickly propagate one configuration across multiple hosts in your enterprise. See [Configuring Instant Messaging After Installing or Upgrading](#) for information on using the state file to configure a new instance of Instant Messaging. To generate a state file, perform the following steps:

1. Log in as superuser.
2. Change to the directory in which you installed Instant Messaging.
By default, this directory is `/opt/sun/comms/im` on Oracle Solaris, and `/opt/sun/comms/im` on Red Hat Linux.
3. Run the `configure` utility by typing the following at the command-line:
`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, use the following command:
`configure --nodisplay --silent --state 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 lists a few examples of using the `configure` command.

To configure through the Command-Line Interface (CLI) mode and save the inputs that you provide in the state file, type the following command:

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

To configure through the CLI mode and use the values from the state file, type the following command:

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

To configure through the silent mode and use the values from the state file, type the following command:

```
./configure --silent <stateFileName>
```

To configure through the CLI mode and use the values from the state file, type the following command. The command saves a state file. It does not do the actual configuration as the `--no` option is used.

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

Sample Configuration Using the `configure` Utility

The following table lists a sample configuration using default values for all options.

Category and Options	Sample or Default value	Your value
Component Selection		
<pre>Select all components you wish to configure. 1. [X] Server components 2. [X] Client components</pre>	1, 2	
User Management Options		
<pre>Use Access Manager for Single-Sign-On [no]</pre>	no	
<pre>Use Access Manager for Policy [no]</pre>	no	
<pre>Instant Messaging user properties can be maintained using one of the following storage systems: 1. On the file system 2. In the directory Enter the number corresponding to your choice: [1]</pre>	1	
Service Runtime Options		
<pre>Runtime User ID : [inetuser]</pre>	inetuser	
<pre>Runtime Group ID: [inetgroup]</pre>	inetgroup	

Runtime Directory [/var/opt/SUNWiim]	/var/opt/SUNWiim	
Network Access Points		
Domain Name	foo.sun.com	
XMPP Port [5222]	5222	
Multiplexed XMPP Port [45222]	45222	
XMPP Server Port [5269]	5269	
Disable Server (enable only multiplexor) [no]	no	
LDAP Configuration		
LDAP Host Name [imhost.siroe.com]	imhost.siroe.com	
LDAP Port Number [389]	389	
Base DN [dc=siroe,dc=com]		
Base DN	cn=Directory Manager	
Base Password		

<pre>Are you sure you want to use this host for LDAP connections? 1. Choose New 2. Accept Enter the number corresponding to your choice: [1]</pre>	1	
Mail Server Options		
<pre>Enable Email Integration [yes]</pre>	yes	
<pre>SMTP Server [imhost]</pre>	imhost	
<pre>Enable Email Archiving [yes]</pre>	yes	
Messenger Resources Download Configuration		
<pre>Deploy Messenger Resources [yes]</pre>	yes	
<pre>Codebase [http://imhost:80/im]</pre>	http://imhost:80/im	
<pre>Enable Audio? [no]</pre>	no	
<pre>Webcontainer Path []</pre>	Web container base directory	
<pre>Web Administration URL []</pre>	https://machinename:port	
<pre>Web Administrator User Id [admin]</pre>	admin	
<pre>Web Administrator Password</pre>		
HTTP Gateway Deployment Configuration		

<input type="checkbox"/> Deploy IM HTTP Gateway [yes]	yes	
<input type="text" value="Context Root [http://imhost:80/httpbind]"/>	http://imhost:80/httpbind	
IMPS Gateway Deployment Configuration		
<input type="checkbox"/> Deploy IM IMPS Gateway [yes]	yes	
<input type="text" value="Context Root [http://imhost:80/httpbind]"/>	http://imhost:80/httpbind	
Calendar Agent configuration		
<input type="checkbox"/> Enable Calendar Agent [no]	no	
<input type="checkbox"/> Enable local component [no]	no	
SMS Gateway Configuration		
<input type="checkbox"/> Enable SMS Gateway [no]	no	
<input type="checkbox"/> Enable local component [no]	no	
MSN Gateway Configuration		
<input type="checkbox"/> Enable MSN Gateway [no]	no	
<input type="checkbox"/> Enable local component [no]	no	
AIM Gateway Configuration		
<input type="checkbox"/> Enable AIM Gateway [no]	no	

Enable local component [no]	no	
YAHOO Gateway Configuration		
Enable YAHOO Gateway [no]	no	
Enable local component [no]	no	
Instant Messaging Services Startup Services Startup Configuration		
Start Services After Successful Configuration [yes]	yes	
Start Services When System starts [yes]	yes	

Adding Instant Messaging and Presence Services to a Sub-organization in Access Manager for Single Sign-On and Policy Management Support

If you are using Instant Messaging with other server products in the Communications Suite, such as Messaging Server, and you want to use Access Manager for single sign-on (SSO) or policy management, you need to manually configure Access Manager-based services for Instant Messaging. This is because configuration of some Communications Suite products, for example Messaging Server, creates one or more domains under the top-level organization in Access Manager. The `configure` utility automatically adds these services to the top-level organization if you select `yes` when prompted to leverage an Access Manager deployment for SSO or policy management.

To Manually Assign Instant Messaging and Presence Services to a Sub-organization in Access Manager

1. In a web browser, log into the Access Manager admin console:
<http://hostname:port/amconsole>
For example,
<http://amserver.company22.example.com:80/amconsole>
2. Select Organizations from the View drop-down list in the navigation pane (left pane).
A list of the domains under the top-level organization is displayed in the left pane.
3. In the navigation pane, click the name of domain under the top-level organization to which you want to add services.
For example:
mydomain.example.com
4. In the navigation pane, select Services from the View drop-down list.
A list of services assigned to the domain appear in the navigation pane.
5. Click Add in the navigation pane.
The data pane (right pane) displays a list of services you can add to the domain.

6. Under Instant Messaging Configuration in the data pane, select the Instant Messaging service and Presence Service check boxes and click OK.
The services you selected are now listed in the navigation pane and have been assigned to the domain under the top-level organization.

Creating Multiple Instances from a Single Instant Messaging Installation

You can create multiple instances of Instant Messaging on a single host from one installation. You may want to do this in order to create a secure version of Instant Messaging, 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 on a single host share binaries but have unique versions of runtime and configuration files.

To Create an Additional Instance of Instant Messaging from an Existing Installation

This procedure assumes that you have used default installation and configuration values for *im-svr-base* and *im-runtime-base*. If you installed using the default values, the original runtime directory is:

Oracle Solaris: `/var/opt/SUNWiim/default`
Red Hat Linux: `/var/opt/sun/im/default`

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

1. Create a runtime directory for the new instance.
For example, to create a new runtime directory for instance *xyz*, type
`mkdir /var/opt/SUNWiim/xyz` on Oracle Solaris
`mkdir /var/opt/sun/im/xyz` on Red Hat Linux
2. Create a log directory for the new instance.
For example, to create a new log directory for instance *xyz*, type
`mkdir /var/opt/SUNWiim/xyz/log` on Oracle Solaris
`mkdir /var/opt/sun/im/xyz/log` on Red Hat Linux
3. If you are using a file-based property store for user data, you need to create a database directory (*im-db-base*) for the new instance.
For example, to create a new database directory for instance *xyz*, type
`mkdir /var/opt/SUNWiim/xyz/db` on Oracle Solaris
`mkdir /var/opt/sun/im/xyz/db` on Red Hat Linux
4. Copy the contents of the *im-svr-base* directory and all of its subdirectories into the newly created directories:
For example:
`cp -r /etc/opt/SUNWiim/default /etc/opt/SUNWiim/xyz` on Oracle Solaris
`cp -r /etc/opt/sun/im/default /etc/opt/sun/im/xyz` on Red Hat Linux
5. Open the new instance's *imadmin* script in a text editor.
By default, this script is stored under the *im-svr-base* directory you just created for the new instance.
Oracle Solaris: `/etc/opt/SUNWiim/xyz/imadmin`
Red Hat Linux: `/etc/opt/sun/im/xyz/imadmin`
6. In the *imadmin* script, 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/SUNWiim/default/config/iim.conf` to
`/etc/opt/SUNWiim/xyz/config/iim.conf`
On Red Hat Linux, change `/etc/opt/sun/im/default/config/iim.conf` to
`/etc/opt/sun/im/xyz/config/iim.conf`
7. Save and close the *imadmin* script.
8. Open the new instance's *iim.conf* file in a text editor.

- By default, the `iim.conf` file is stored in the `im-cfg-base` directory you created for the new instance.
- Oracle Solaris: `/etc/opt/SUNWiim/xyz/config/iim.conf`
Red Hat Linux: `/etc/opt/sun/im/xyz/config/iim.conf`
9. Modify the port numbers in `iim.conf` so they do not conflict with the original instance. The default port numbers are as follows:
 - > Server port (`iim_server.port`) - 5269
 - > Multiplexor listen port (`iim_mux.listenport`) - 5222
 - > Multiplexor to server communication port (`iim_mux.serverport`) - 45222For more information about these parameters, see [Instant Messaging Configuration Parameters in `iim.conf`](#).
 10. Modify `iim.instancedir` to point to `im-svr-base`.
See [Instant Messaging Server Directory Structure](#) for information on `im-svr-base`.
 11. Modify `iim.instancevardir` to point to the runtime directory for the new instance.
For example:
 - On Oracle Solaris, change `/var/opt/SUNWiim/default` to `/var/opt/SUNWiim/xyz`
 - On Red Hat Linux, change `/var/opt/sun/im/default` to `/var/opt/sun/im/xyz`
 12. Save and close `iim.conf`.
 13. Ensure that file and directory ownership and permissions are the same for all instances.
 14. Make renamed copies of `/opt/sun/comms/im/html/locale/im.html`, `im.jnlp`, and `index.html` resource files, and modify the copies to point to the new instance's port number.
 15. Redeploy the renamed resource files.
See [Redeploying Resource Files](#) for instructions.
 16. Start the new instance:
 - Oracle Solaris: `/etc/opt/SUNWiim/xyz/imadmin start`
 - Red Hat Linux: `/etc/opt/sun/im/xyz/imadmin start`

Chapter 2. Setting up and Launching Instant Messenger

Setting up and Launching Instant Messenger

This information describes how to configure the web container and client systems to support Instant Messenger.

Topics:

- [Enabling Java Web Start](#)
- [Configuring Client Systems for Instant Messaging](#)
- [Launching Instant Messenger](#)

Enabling Java Web Start

To use Instant Messenger with Java Web start, you need to install the software, then configure your web container to work with Java Web Start. Instructions on installing Java Web Start are available at the following URL: <http://www.oracle.com/technetwork/java/javase/javawebstart/index.html>

To enable Java Web Start support in your web container, you need to edit the web container's `mime.types` file to include the following definition for JNLP:

Content Type: `application/x-java-jnlp-file`

Suffix: `jnlp`

This section provides the following instructions:

- [To Add the MIME Type to Web Server Enterprise Edition](#)
- [To Add the MIME Type to Apache Web Container](#)

To Add the MIME Type to Web Server Enterprise Edition

1. Type the following URL to access the administration server in your browser:

```
http://hostname.domain:admin-port
```

For example, `http://budgie.siroe.com:8888`.

Web Server displays a window prompting you for a user name and password.

2. Type the administration user name and password you specified during the web container installation.
The web container displays the Administration Server page.
3. On the Manage Servers page, click Manage.
The web container displays the Server Manager page.
4. Click the MIME Types link.
5. From the MIME file drop-down list, choose a MIME type to edit and click OK.
6. In the Global MIME Types page, select `type` from the Category drop-down list.
7. In the Content-Type text box, type `application/x-java-jnlp-file`

8. In the File-Suffix text box, type `jnlp`
9. Click New Type to create the MIME type.
10. Restart the web container for this change to take effect.

To Add the MIME Type to Apache Web Container

1. Add the following line to the `mime.types` file:
`application/x-java-jnlp-file jnlp`
By default, this file is located in the Apache Web Container configuration directory.

Configuring Client Systems for Instant Messaging

If the client machine has the appropriate version of Java installed, there are no additional requirements to use either Java Plug-in or Java Web Start. Netscape Navigator version 7 as well as the recent versions of the Mozilla browser include the latest version of Java, while Internet Explorer does not. If the client machine does not have the required version of Java installed, you need to install Java Web Start. You can download and Install Java from the following URL: <http://www.java.sun.com/j2se>

You can download and install Java Web Start from the following URL:
<http://java.sun.com/javase/downloads/index.jsp>

Launching Instant Messenger

You can invoke Instant Messenger as an applet within a web browser, or as a standalone application as described in the following sections:

- [Invoking Instant Messenger From a Web Browser](#)
- [Invoking Instant Messenger as a Standalone Application](#)

Invoking Instant Messenger From a Web Browser

To invoke Instant Messenger as an applet within a web browser, perform the following steps:

1. Start the web browser.
2. Go to the Instant Messaging home page.
By default, the home page is stored as `index.html`. Use the following format to locate the Instant Messaging home page.
`http://codebase/index.html`
Where *codebase* is the URL that corresponds to the location of the resource files on the web container.
3. Click Use Java Plug-In.
If you customized the home page and changed the link text, click the link that corresponds to invoking Instant Messenger as an applet within a browser. The link points to either `im.jnlp` (standard and TLS mode) or `imssl.jnlp` (legacy SSL mode).
When the Instant Messenger session is established using the Java Plug-in, the browser window must be dedicated to its use.
You cannot locate any other URLs with this browser window, nor can you close the browser window without terminating the Instant Messenger session.

Invoking Instant Messenger as a Standalone Application

To invoke Instant Messenger as a standalone application, perform the following steps:

1. Start the web browser.
2. Go to the Instant Messaging home page.

By default, the home page is stored as `index.html`. Use the following format to locate the Instant Messaging home page.

`http://codebase/index.html`

Where *codebase* is the URL that corresponds to the location of the resource files on the web container.

3. Click Start.

If you customized the home page and changed the link text, click the link that corresponds to invoking Instant Messenger using Java Web Start. The link points to either `im.html` (standard or TLS mode) or `imssl.html` (legacy SSL mode).

See [Customizing Instant Messenger](#) for information on customizing the resource pages.

Chapter 3. Configuring Instant Messaging for High Availability (Oracle Solaris Only)

Configuring Oracle Communications Instant Messaging Server for High Availability (Oracle Solaris Only)

Configuring Instant Messaging for high availability (HA) provides for monitoring of and recovery from software and hardware failures. The high availability feature is implemented as a failover data service, not a scalable service, and is supported on Oracle Solaris only. This chapter describes an Instant Messaging HA configuration using Oracle Solaris Cluster software. See [HA Related Documentation](#) for more information about scalable and failover data services provided by Oracle Solaris Cluster.

Topics:

- [Instant Messaging HA Overview](#)
- [Setting Up HA for Instant Messaging](#)
- [Stopping, Starting, and Restarting the Instant Messaging HA Service](#)
- [Stopping, Starting, and Restarting Instant Messaging Components in a Deployment with Sun Cluster](#)
- [Managing the HA RTR File for Instant Messaging](#)
- [Starting and Stopping the Instant Messaging HA Service](#)
- [Removing HA for Instant Messaging](#)
- [HA Related Documentation](#)

Instant Messaging HA Overview

You use Oracle Solaris Cluster with Instant Messaging to create a highly available deployment. This section provides information about HA requirements, terms used in examples in this chapter, and permissions you need to configure HA in the following sections:

- [Instant Messaging HA Configuration Software Requirements](#)
- [Instant Messaging HA Configuration Permission Requirements](#)
- [Instant Messaging HA Configuration Terms and Checklist](#)

Before you begin, you should be familiar with general HA concepts, and Oracle Solaris Cluster software in particular. For more information, see [HA Related Documentation](#).

Instant Messaging HA Configuration Software Requirements

An Instant Messaging HA configuration requires the software shown in [Table 4-1](#).

Table 4-1 Software Requirements for Instant Messaging HA Configuration

Software and Version	Notes and Patches
Oracle Solaris 9	All versions of Oracle Solaris 9 are supported. Oracle Solaris 9 requires Oracle Solaris Cluster 3.0 U3 at a minimum. Oracle Solaris 9 includes Oracle Solaris Logical Volume Manager (LVM).
Oracle Solaris 10	All versions of Oracle Solaris 10 are supported.
Oracle Solaris Cluster 3.1	Oracle Solaris Cluster software must be installed and configured on all nodes in the cluster. To install Oracle Solaris Cluster, use the Communications Suite installer by following the installation process in the http://download.oracle.com/docs/cd/E19893-01/819-7560/index.html . After you install the Oracle Solaris Cluster software, you must configure the cluster. For information, refer to the Oracle Solaris Cluster System Administration Guide for Oracle Solaris. For related documentation, see HA Related Documentation . Sun Cluster Patches For Solaris 9 and 10, you can download patches from http://support.oracle.com .
Veritas Volume Manager (VxVM) 3.x	Requires version 3.5 at a minimum, plus required patches.
Veritas File System (VxFS) 3.x	Requires version 3.5 at a minimum, plus required patches. HAStoragePlus requires patch 110435-08 at a minimum.

Instant Messaging HA Configuration Permission Requirements

To install and configure an Instant Messaging HA configuration, log in as or become superuser (`root`) and specify a console or window for viewing messages sent to `/dev/console`.

Instant Messaging HA Configuration Terms and Checklist

Table 4-2 describes the variable terms used in the examples in this chapter for configuration examples. In addition, you will need to gather the information before you configure HA for Instant Messaging. You will be prompted for this information during configuration. Use this checklist in conjunction with the checklist in Table 1-1.

Table 4-2 HA Configuration Checklist

Name in Example	Description	Your Value
<i>/global/im</i>	Global file system mount point used with a cluster file system or HAStoragePlus.	
<i>/local/im</i>	Local directory to use as a mount point for the shared disk if you are using HAStoragePlus.	
<i>im-logical-host</i>	Logical host name	
<i>im-logical-host-ip</i>	Logical host IP numeric address	
<i>im-node-1</i>	Node 1 FQDN	
<i>im-node-2</i>	Node 2 FQDN	
<i>im-resource-group</i>	Instant Messaging resource group	
<i>im-resource-group-store</i>	Instant Messaging storage resource	
<i>im-resource</i>	Instant Messaging resource	
<i>im-runtime-base</i> (Includes <i>im-runtime-base/db</i> and <i>im-runtime-base/logs</i>)	For the location of the runtime directory (which includes the database and log subdirectories), select global, shared partitions. For example:* Instant Messaging runtime directory (<i>im-runtime-base</i>): <i>/global/im/var/opt/SUNWiim/default</i> * Database subdirectory (<i>im-db-base</i>): <i>/global/im/var/opt/SUNWiim/default/db</i> * Log subdirectory: <i>/global/im/var/opt/SUNWiim/default/logs</i> See Instant Messaging Configuration File and Directory Structure Overview for more information about the runtime directory and the database and logs subdirectories.	

Setting Up HA for Instant Messaging

The following is a high-level list of the steps necessary to install and configure an Instant Messaging HA configuration with two nodes:

- [Choosing a Local or Shared Disk for Configuration Files and Binaries](#)
- [Preparing Each Node in the Cluster](#)
- [Selecting the Installation Directory \(*im-svr-base*\)](#)
- [Installing Sun Java System Products and Packages](#)
- [Configuring the HA Environment](#)
- [Configuring the Logical Host](#)
- [Registering and Activating the Storage Resource](#)
- [Registering the Resource Type and Creating a Resource](#)
- [Verifying the Instant Messaging HA Configuration](#)
- [Troubleshooting the Instant Messaging HA Configuration](#)

Choosing a Local or Shared Disk for Configuration Files and Binaries

Before you begin, you need to decide which of the following deployments best suits your needs. In both environments, shared components are installed locally on every node in the cluster. In addition, in both environments, runtime files are installed on a shared disk.

- **Using a local disk for configuration files and binaries.** The advantage to this setup is that

upgrading Instant Messaging requires minimal downtime because you can upgrade on nodes where Instant Messaging is offline. The disadvantage is that you need to ensure that the same configuration and version of Instant Messaging exists on all nodes in the cluster. In addition, if you choose this option, you need to determine whether you will be using HAStoragePlus to mount a file system from a shared disk on each node when Instant Messaging data services are brought online, or if you will be using the cluster file system for global runtime files.

- **Using a shared disk for configuration files and binaries.** This setup is easier to administer, but you need to bring Instant Messaging down on all nodes in the cluster before upgrading.

Preparing Each Node in the Cluster

On each node in the cluster, you need to create the Instant Messaging runtime user and group under which the components will run. The UID and GID numbers must be the same on all nodes in the cluster.

- **Runtime User ID.** The user name under which Instant Messaging server runs. This name should **not** be `root`. The default is `inetuser`.
- **Runtime Group ID.** The group under which Instant Messaging server runs. The default is `inetgroup`.

Although the `configure` utility can create these names for you, you can create them before you run the configuration program, as part of the preparation of each node as described in this chapter. In addition, depending on whether you are using a local or shared disk, you may not run `configure` on a particular node and must manually create the runtime user and group ID.

The runtime user and group ID names must be in the following files:

- `inetuser`, or the name you select, in `/etc/passwd` on all nodes in the cluster
- `inetgroup`, or the name you select, in `/etc/group` on all nodes in the cluster

See [Creating a UNIX System User and Group](#) for instructions. Refer to your operating system documentation for detailed information about users and groups.

Selecting the Installation Directory (*im-svr-base*)

For Instant Messaging, the Java Enterprise System installer uses `/opt/SUNWiim` on Solaris as the default installation directory (*im-svr-base*). However, if you are using a shared disk for configuration files and binaries, you must specify a global (shared) installation directory. For example: `/global/im/opt/SUNWiim`.

If you are using a local disk, you can install Instant Messaging to the default directory. However, you should install Instant Messaging in the same directory on each machine in the node.

Installing Sun Java System Products and Packages

You install products and packages using the Communications Suite installer program. For more information about the installer, refer to the <http://download.oracle.com/docs/cd/E19893-01/819-7560/index.html>.

Table 4-3 lists the products or packages required for a multiple node cluster configuration.

Table 4-3 Products and Packages Required for a Multiple Node Instant Messaging HA Configuration

Product or Package	Node 1	Node <i>n</i>
Oracle Solaris Cluster Software	Yes	Yes
Instant Messaging 7.2 Server	Yes	Yes, if you are using a local disk for configuration files and binaries. No, if you are using a shared disk for configuration files and binaries.
Oracle Solaris Cluster Agent for Instant Messaging (SUNWiimsc)	Yes	Yes, if you are using a local disk for configuration files and binaries. No, if you are using a shared disk for configuration files and binaries.
Shared components If you are using HAStoragePlus, you must also install SUNWscu	Yes	Yes

Configuring the HA Environment

The steps you need to perform vary depending on whether or not you are using a local or shared disk for configuration files and binaries.

If you are using a local disk for configuration files and binaries, follow the steps in the following two procedures:

- [To Configure HA on Node 1 Using a Local Disk for Configuration Files and Binaries](#)
- [To Configure HA on Node *n* Using a Local Disk for Configuration Files and Binaries](#)

If you are using a shared disk for configuration files and binaries, follow the steps in the following two procedures:

- [To Configure HA on Node 1 Using a Shared Disk for Configuration Files and Binaries](#)
- [To Configure HA on Node *n* Using a Shared Disk for Configuration Files and Binaries](#)

To Configure HA on Node 1 Using a Local Disk for Configuration Files and Binaries Before You Begin

Fill out the checklists in [Table 1-1](#) and [Table 4-2](#) and have your answers readily available.

1. Install products and packages using the Java Enterprise System installer.
See [Selecting the Installation Directory \(*im-svr-base*\)](#) for specific instructions on choosing an installation directory.
See [Table 4-3](#) for a list of required products and packages for HA. Refer to <http://download.oracle.com/docs/cd/E19893-01/819-7560/index.html> for specific instructions.
2. If you are using HAStoragePlus for the runtime files, mount a shared disk to a local directory, otherwise skip to [Step 3](#).
For example:
3. Create the mount point (`/local/im/im-runtime-base/`) if it does not already exist.
When prompted during configuration in [Step 4](#) you will specify this directory (`/local/im/im-runtime-base/`) as the Instant Messaging Server Runtime Files Directory.
4. Use the `mount` command to mount the disk on `/local/im/im-runtime-base`.
5. Run the `configure` utility.
See [Configuring Instant Messaging After Installation](#) for instructions.
6. When prompted for the Instant Messaging Server Runtime Files Directory, enter one of the following:
 - If you are using an HAStoragePlus for the runtime files, enter `/local/im/im-runtime-base/`.

- If you are using a cluster file system for the runtime files, enter `/global/im/im-runtime-base/`. Where `/global/im` is the global directory in the cluster file system.
1. When prompted for the Instant Messaging host name, enter the logical host.
Choose to accept the logical host even if the `configure` utility cannot connect to the specified host. The logical host resource may be offline at the time you run the `configure` utility.
 2. Do not choose to start Instant Messaging after configuration or on system startup.
In an HA configuration, the Instant Messaging service also requires the logical host to be online for Instant Messaging to work properly.
 3. If you are using HASToragePlus for runtime files, unmount the shared disk.

To Configure HA on Node *n* Using a Local Disk for Configuration Files and Binaries Before You Begin

Ensure that you have completed HA configuration on Node 1 as described in the previous procedure ([To Configure HA on Node 1 Using a Local Disk for Configuration Files and Binaries](#)).

Have your answers for the checklists in [Table 1-1](#) and [Table 4-2](#) readily available.

1. Install products and packages using the Java Enterprise System installer.
Choose the same path you used when you installed Instant Messaging on node 1 for each subsequent node in the cluster. See [Selecting the Installation Directory \(*im-svr-base*\)](#) for specific instructions.
See [Table 4-3](#) for a list of required products and packages for HA. Refer to <http://download.oracle.com/docs/cd/E19893-01/819-7560/index.html> for specific instructions.
2. Run the `configure` utility.
See [Configuring Instant Messaging After Installation](#) for instructions.
3. When prompted for the Instant Messaging Server Runtime Files Directory, enter the same value that you provided for Node 1.
4. When prompted for the Instant Messaging host name, enter the same logical host you provided for Node 1.
Choose to accept the logical host even if the `configure` utility cannot connect to the specified host. The logical host resource may be offline at the time you run the `configure` utility.
5. When prompted for the user and group, enter the same value that you provided for Node 1.
6. Do not choose to start Instant Messaging after configuration or on system startup.
In an HA configuration, the Instant Messaging service also requires the logical host to be online for Instant Messaging to work properly.

To Configure HA on Node 1 Using a Shared Disk for Configuration Files and Binaries Before You Begin

Fill out the checklists in [Table 1-1](#) and [Table 4-2](#) and have your answers readily available.

You must use a cluster file system if you are using a shared disk for configuration files and binaries, not HASToragePlus.

1. Install products and packages in a directory in the cluster file system using the Java Enterprise System installer.
When you install Instant Messaging, you must specify a directory other than the default directory. See [Selecting the Installation Directory \(*im-svr-base*\)](#) for specific instructions.
See [Table 4-3](#) for a list of required products and packages for HA. Refer to <http://download.oracle.com/docs/cd/E19893-01/819-7560/index.html> for specific instructions.
2. Create a soft link from `/etc/opt/SUNWim` that points to `/global/im/etc/opt/SUNWim`.
3. Run the `configure` utility from the global directory where you installed Instant Messaging (`/global/im/im-svr-base/configure`).
See [Configuring Instant Messaging After Installation](#) for instructions.
4. When prompted for the Instant Messaging Server Runtime Files Directory, enter the value for /

global/im/im-runtime-base.

5. When prompted for the Instant Messaging host name, enter the logical host.
Choose to accept the logical host even if the `configure` utility cannot connect to the specified host. The logical host resource may be offline at the time you run the `configure` utility.
6. Do not choose to start Instant Messaging after configuration or on system startup.
In an HA configuration, the Instant Messaging service also requires the logical host to be online for Instant Messaging to work properly.

To Configure HA on Node *n* Using a Shared Disk for Configuration Files and Binaries Before You Begin

Ensure that you have completed HA configuration on Node 1 as described in the previous procedure ([To Configure HA on Node 1 Using a Shared Disk for Configuration Files and Binaries](#)).

Have your answers for the checklists in [Table 1-1](#) and [Table 4-2](#) readily available.

1. Create a soft link from `/etc/opt/SUNWiim` that points to `/global/im/etc/opt/SUNWiim`.
2. Create a soft link for the resource type registration (RTR) file:

```
ln -s /global/im/im-svr-base/cluster/SUNW.iim \
/usr/cluster/lib/rgm/rtreg/SUNW.iim
```

Configuring the Logical Host

Before starting Instant Messaging, you need to create a resource group, add the logical host, and bring the resource group online.

To Configure the Resource Group With the Logical Host

1. Create an Instant Messaging failover resource group named *im-resource-group*:

```
# scrgadm -a -g im-resource-group -h im-node-2,im-node-1
```

2. Add the logical host name *im-logical-host* to the resource group.
Instant Messaging will listen on this host name.

```
# scrgadm -a -L -g im-resource-group -l im-logical-host
```

3. Bring the resource group online:

```
# scswitch -Z -g im-resource-group
```

Registering and Activating the Storage Resource

Before you can bring the Instant Messaging data service online, you need to register and activate the storage resource as described in this section.

To Register and Enable the Storage Resource

1. Register the storage resource.
If you are using HAStoragePlus with a global file system (GFS), set the mount point as the value

for the *FileSystemMountPoints* property. For example:

```
# scrgadm -a -j im-resource-group-store -g im-resource-group -t
SUNW.HAStorage \
-x FileSystemMountPoints=/global/im -x AffinityOn=True
```

Otherwise, specify the mount point as the value for the *ServicePaths* property. For example:

```
# scrgadm -a -j im-resource-group-store -g im-resource-group -t
SUNW.HAStorage \
-x ServicePaths=/global/im -x AffinityOn=True
```

2. Enable the storage resource:

```
# scswitch -e -j im-resource-group-store
```

Registering the Resource Type and Creating a Resource

Before you start the HA Instant Messaging server or multiplexor, you need to register the resource type `SUNWiimsc` with Sun Cluster and create a resource.

To Register the Resource Type and Create a Resource

1. Register the resource type.

```
# scrgadm -a -t SUNW.iim
```

2. Create the resource.
Enter the following command on a single line:

```
# scrgadm -a -j im-resource -g im-resource-group -t SUNW.iim -x
Confdir_list=/global/im/im-resource-group -y
Resource_dependencies=im-resource-group-store
```

3. Enable the resource:

```
# scswitch -e -j im-resource
```

4. Start Instant Messaging components.

Verifying the Instant Messaging HA Configuration

After you start Instant Messaging, you need to verify the HA configuration as described in this section.

To Verify the HA Configuration for Instant Messaging

1. Check that all required processes are running.
2. Conduct a switchover of the service to the backup node to ensure the high availability.
For example, if the service is running on *im-node-1*, issue the following command to switch the

service to *im-node-2*.

```
# scswitch -z -g im-resource-group -h im-node-2
```

3. Check that all required processes are started on *im-node-2*.

Troubleshooting the Instant Messaging HA Configuration

To help with troubleshooting, error messages are written to the error log. The logs are controlled by the `syslog` facility. For information about using the logging facility, refer to the [HA Related Documentation](#) and to the man page for `syslog.conf`.

Stopping, Starting, and Restarting the Instant Messaging HA Service

To start and stop the Instant Messaging HA service, use the Sun Cluster `scswitch` command.

For more information about the Oracle Solaris Cluster `scswitch` command, refer to the Oracle Solaris Cluster Reference Manual for Oracle Solaris.

To Start the Instant Messaging HA Service

1. Type the following at the command line:

```
# scswitch -e -j im-resource
```

To Stop the Instant Messaging HA Service

1. Type the following at the command line:

```
# scswitch -n -j im-resource
```

To Restart the Instant Messaging HA Service

1. Type the following at the command line:

```
# scswitch -R -j im-resource
```

Stopping, Starting, and Restarting Instant Messaging Components in a Deployment with Sun Cluster

The `imadmin` command checks to ensure it is not running on a cluster node before attempting to stop, start, or restart an Instant Messaging component. If `imadmin` determines that it is running on a cluster node, it returns an error instead of performing the command. Use the Sun Cluster administrative utilities to stop, start, and restart Instant Messaging components in a deployment with Oracle Solaris Cluster.

Managing the HA RTR File for Instant Messaging

The resource type registration (RTR) file is an ASCII text file that describes a highly-available resource type that runs under the control of the Resource Group Manager (RGM). The RTR file is used as an input file by the `scrgadm` command to register the resource type into the cluster configuration. The Instant Messaging RTR file, `SUNW.im`, is created when you install the `SUNWiimsc` package during HA configuration.

This section provides information about managing this file in the following sections:

- [Instant Messaging HA Overview](#)
- [Setting Up HA for Instant Messaging](#)
- [Stopping, Starting, and Restarting the Instant Messaging HA Service](#)
- [Stopping, Starting, and Restarting Instant Messaging Components in a Deployment with Sun Cluster](#)
- [Managing the HA RTR File for Instant Messaging](#)
- [Starting and Stopping the Instant Messaging HA Service](#)
- [Removing HA for Instant Messaging](#)
- [HA Related Documentation](#)

Instant Messaging RTR File Parameters

The following table lists the extension properties in the Instant Messaging RTR file (`SUNW.im`) that are specific to Instant Messaging.

Table 4-4 `SUNW.im` Extension Properties

Extension Property	Default	Description
Server_Root	If you are using a local disk to store configuration files and binaries: <i>im-svr-base</i> If you are using a shared directory to store configuration files and binaries: /<global>/<im>/<im-svr-base>	Defines the absolute path to the Instant Messaging server installation directory. By default, <i>im-svr-base</i> is /opt/SUNWiim on Oracle Solaris.
Confdir_list	None	Defines the absolute path to the Instant Messaging configuration. This value is set during the installation of SUNWiimsc.
Monitor_retry_count	4	Defines how many times you want the process monitor facility (PMF) to attempt to restart the fault monitor if it determines it is not running.
Monitor_retry_interval	2 (minutes)	Time, in minutes, between restart attempts made by the PMF on the fault monitor.
Probe_timeout	30 (seconds)	Time, in seconds, that the Sun Cluster probe will wait for a successful connection to Instant Messaging.
Failover_enabled	True	Determines whether or not to failover to another node if the configured number of retries (<i>retry_count</i>) is exceeded during the configured retry interval (<i>retry_interval</i>). See http://download.oracle.com/docs/cd/E19787-01/819-3055/ for more information on retry and other parameters.

Customizing the RTR File for Instant Messaging

You can modify the values for several of the extension properties in the Instant Messaging RTR file (`SUNW.iim`) to configure your HA environment. Extension properties are properties that are specific to the resource type. These properties are inherited by every resource of the same type. Instant Messaging extension properties are described in [Table 4-4](#).

See the documentation for `rt_reg` and `property_attributes` in <http://download.oracle.com/docs/cd/E19787-01/819-3055/> for more information on the contents of resource type registration files and instructions on customizing values for extension properties.

Choosing a High Availability Model for Your Instant Messaging Deployment

This section lists the high availability models, and describes the procedure to install and configure the asymmetric and symmetric models for deployment. This section contains the following topics:

- [High-Level Task List for an Asymmetric HA Deployment](#)
- [High-Level Task List for a Symmetric HA Deployment](#)
- [Installing and Configuring in an Asymmetric HA Environment](#)
- [Installing and Configuring in a Symmetric HA Environment](#)

The following table summarizes the advantages and disadvantages of each high availability model. Use this information to decide the appropriate model for your deployment.

Model	Advantages	Disadvantages	Recommended Users
Asymmetric	<ul style="list-style-type: none"> • Simple Configuration • Backup node is 100% reserved. • Rolling upgrade with negligible downtime 	Machine resources are not fully utilized.	A small service provider with plans to expand in the future.
Symmetric	<ul style="list-style-type: none"> • Efficient use of system resources • Higher availability 	<ul style="list-style-type: none"> • Resource contention on the backup node. • HA requires fully redundant disks. 	A small corporate deployment that can accept performance penalties in the event of a single server failure.
N+1	<ul style="list-style-type: none"> • Load distribution • Easy expansion 	Management and configuration complexity.	A large service provider who requires distribution with no resource constraints.

High-Level Task List for an Asymmetric HA Deployment

The following is a list of the tasks necessary to install and configure Instant Messaging for asymmetric high availability:

1. Prepare the nodes.
 - a. Install the Oracle Solaris operating system on all the nodes of the cluster.
 - b. Install Oracle Solaris Cluster software on all the nodes of the cluster.
 - c. Install the Instant Messaging HA Agents package, `SUNWimsc`, on all the nodes of the cluster by using the Communication Suite 6 Update 1 installer.
 - d. Create a file system on the shared disk.
 - e. Install Instant Messaging on all the nodes of the cluster by using the Communications Suite 6 Update 1 installer.
 - f. Create a symbolic link from the Instant Messaging `/etc/opt/SUNWim` directory to the shared disk `IM_RUNTIME_DIR` directory on all the nodes of the cluster.
2. Configuring the first or the primary node.
 - a. Using the Oracle Solaris Cluster command-line interface, set up HA on the primary node.
 - b. Run the Instant Messaging `configure` utility on the primary node.
 - c. Using the Oracle Solaris Cluster command-line interface, create and enable a resource group for Instant Messaging.

For step-by-step instructions, see [Installing and Configuring in an Asymmetric HA Environment](#).

High-Level Task List for a Symmetric HA Deployment

The following is a list of the tasks necessary to install and configure Instant Messaging for Symmetric High Availability:

1. Prepare the nodes.
 - a. Install the Oracle Solaris operating system software on all the nodes of the cluster.
 - b. Install the Oracle Solaris Cluster software on all the nodes of the cluster.
 - c. Create four file systems. You can create a cluster file systems or global file systems or failover file systems or local file systems.
 - d. Create the necessary directories.
 - e. Install the Instant Messaging HA Agents package, `SUNWimsc`, on all nodes of the cluster by using the Communications Suit 6 installer.
2. Install and configure the first instance of Instant Messaging HA.
 - a. Using the Communications Suite 6 installer, install Instant Messaging on the first node of the cluster.
 - b. Using the Oracle Solaris Cluster command-line interface, configure HA on the first node.
 - c. Create a symbolic link from the Instant Messaging `/etc/opt/SUNWim` directory to the shared disk `IM_RUNTIME_DIR` directory on the first node.
 - d. Run the Instant Messaging `configure` utility on the first node.
 - e. Using the Oracle Solaris Cluster command-line interface, create and enable a resource group for Instant Messaging on the first node.
 - f. Using the Oracle Solaris Cluster command-line interface to test the successful creation of the resource group, perform a failover to the second node.
3. Install and configure the second instance of Instant Messaging HA.
 - a. Using the Communications Suite 6 installer, install Instant Messaging on the second node of the cluster.
 - b. Using the Oracle Solaris Cluster command-line interface, configure HA on the second node.
 - c. Create a symbolic link from the Instant Messaging `/etc/opt/SUNWim` directory to the shared disk `IM_RUNTIME_DIR` directory on the secondary node.
 - d. Run the Instant Messaging `configure` utility on the second node.
 - e. Using the Oracle Solaris Cluster command-line interface, create and enable a resource group for Instant Messaging on the second node.
 - f. Using the Sun Cluster command-line interface to test the successful creation of the resource group, perform a failover to the first node.

For step-by-step instructions, see [Installing and Configuring in a Symmetric HA Environment](#).

Installing and Configuring in an Asymmetric HA Environment

This section contains instructions for configuring an asymmetric high availability Instant Messaging cluster. This sections contains the following topics:

- Creating the File Systems for Your HA Deployment
- Creating the Instant Messaging Directory on All Shared Disks of the Cluster in Your HA Deployment
- Installing and Configuring HA for Instant Messaging 8

Creating File Systems for HA Deployment

Create a file system on the shared disk. The `/etc/vfstab` directory should be identical on all the nodes of the cluster.

For the cluster file system (CFS), the directory should be similar to the following example.

```
## Cluster File System/Global File System ##
/dev/md/penguin/dsk/d400 /dev/md/penguin/rdisk/d400 /global/im ufs 2 yes
global,logging
```

For the failover file system (FFS), the directory should be similar to the following example.

```
## Fail Over File System/Local File System ##
/dev/md/penguin/dsk/d400 /dev/md/penguin/rdisk/d400 /local/im ufs 2 no
logging
```



Note

The fields in these commands are separated by tabs and not spaces.

Creating the Instant Messaging Directory on all the Shared Disks of the Cluster in the HA Deployment

For all the nodes of the cluster, create a directory, `IM_RUNTIME_DIR`, to store the configuration details and data. For example, to create an Instant Messaging directory on a shared disk, type either one of the following:

```
mkdir -p /local/im
or
mkdir -p /global/im
```

Installing and Configuring HA for Instant Messaging 8 Software

This section contains instructions for the tasks involved in installing and configuring HA for Instant Messaging. Perform the following tasks to complete the configuration:

- [Preparing Each Node of the Cluster](#)
- [Setting Up the Primary Node](#)
- [Invoking the `configure` Utility on the Primary Node](#)

Preparing Each Node of the Cluster

For each node in the cluster, create the Instant Messaging runtime user and group under to run the components. The user ID (UID) and group ID (GID) numbers must be the same on all the nodes in the cluster.

- **Runtime User ID:** User name using which the Instant Messaging server runs. The default value is `inetuser`.
- **Runtime Group ID:** Group using which the Instant Messaging server runs. The default value is `inetgroup`.
Although the `configure` utility creates the IDs, you can create the IDs before you invoke the `configure` utility as part of the preparation of each node. Create the runtime user and group ID on a node where you will not invoke the `configure` utility, which is usually secondary node.

Make sure that the username, group name and the corresponding user ID and group ID are same in the following files on all nodes:

- `inetuser` or the name that you select in the `/etc/passwd` directory on all the nodes in the cluster
- `inetgroup` or the name that you select in the `/etc/group` directory on all the nodes in the cluster

Refer to your operating system documentation for detailed information about users and groups.

Installing Instant Messaging

Selecting the Installation Directory `IM_SVR_BASE`

For Instant Messaging and Instant Messaging Oracle Solaris Cluster agent `IM_SCHA`, the Communications Suite 6 installer uses the `/opt/sun/comms` directory on the Solaris operating system as the default installation directory. The value of the `IM_SVR_BASE` variable is `/opt/sun/comms/im`.

However, if you are using a shared disk for binaries, you must specify a cluster file system (CFS) or a failover file system (FFS) installation directory. For example, if `/global/im/` is the installation directory, then the value of `IM_SVR_BASE` is `/global/im/im`.

If you are using a local disk, you should install the Instant Messaging in the same directory on each machine in the node.

Note

- Configuration files and runtime files reside on a CFS or on a highly-available FFS. Binaries are installed on local file systems on each node at the same location. Enables rolling upgrade of the Instant Messaging software.
- Binaries, configuration files and runtime files either reside on a CFS or on a highly-available FFS. The Instant Messaging installation is required only on one node as the binaries are shared across all the nodes. Instant Messaging upgrade needs a server down time.

Installing Instant Messaging Products and Packages

Install products and packages by using the Communications Suite installer program. The following table lists the products or packages required for a multiple node cluster configuration.

Product or Package	Node 1	Node n
Oracle Solaris Cluster Software	Yes	Yes
Instant Messaging 8 2008 Server	Yes	Yes, if you use a local disk for configuration files and binaries. No, if you use a shared disk for configuration files and binaries.
Oracle Solaris Cluster Agent for Instant Messaging <code>SUNWiimsc</code>	Yes	Yes, if you use a local disk for configuration files and binaries. No, if you use a shared disk for configuration files and binaries.
Shared components	Yes	Yes

Instant Messaging HA Agent Installation

To install the Instant Messaging Oracle Solaris Cluster HA agent, perform the following steps:

1. Type the Communications Suite installer command in the global zone.

```
# commpkg install
```

Note

In case of Oracle Solaris 10 zones, run the above command from global and non-global zones.

2. Select the Instant Messaging Oracle Solaris Cluster HA Agent software when prompted.
3. Type the Oracle Solaris Cluster HA Agent preconfiguration command.

```
# IM_SCHA_BASE/bin/init-config
```

Note

In case of Solaris 10 zones, run the above command only from the global zone.

Setting Up the Primary Node

Use the Oracle Solaris Cluster command line interface to set up HA on the first node.

1. Register the Instant Messaging and HAStoragePlus resource.

```
./scrgadm -a -t SUNW.HAStoragePlus  
./scrgadm -a -t SUNW.iim
```

2. Create a failover Instant Messaging resource group. For example, for a two node asymmetric cluster setup, the following command creates the Instant messaging resource group IM-RG with the primary node as NODE1 and the secondary, or failover, node as NODE2.
./scrgadm -a -g IM-RG -h IM_NODE1,IM_NODE2
3. Create a logical hostname resource in the Instant Messaging resource group and change the resource group state to online. For example, the following instructions create the logical hostname resource LOG_HOST_RS and bring the resource group IM-RG to online state.

```
./scrgadm -a -L -g IM-RG -l LOG_HOST_RS  
./scrgadm -c -j LOG_HOST_RS -y \\  
    R_description="LogicalHostname resource for LOG_HOST_RS"  
./scswitch -Z -g IM-RG
```

4. Create and enable the HAStoragePlus resource. For example, the following commands create and enable the HAStoragePlus resource IM_HASP_RS.

```
scrgadm -a -j IM_HASP_RS -g IM-RG -t  
    SUNW.HAStoragePlus:4 -x FilesystemMountPoints=/IM_RUNTIME_DIR  
scrgadm -c -j IM_HASP_RS -y  
    R_description="Failover data service resource for  
SUNW.HAStoragePlus:4"  
scswitch -e -j IM_HASP_RS
```

5. Create a symbolic link from the Instant Messaging /etc/opt/SUNWiim directory to the shared disk IM_RUNTIME_DIR directory on all the nodes of the cluster.

For example, type the following commands on all the nodes of the cluster:

```
# pwd
/etc/opt/

# ln -s /IM_RUNTIME_DIR SUNWiim
```

Invoking the `configure` Utility on the Primary Node

1. Invoke the `configure` utility. For example, from the `/IM_SVR_BASE` directory type the following command:

```
# pwd
/IM_SVR_BASE

# ./configure
```

For further information about the `configure` utility, see [Configuring Instant Messaging After Installation](#).

2. When prompted for the Instant Messaging Server runtime files directory `IM_RUNTIME_DIR`, type either of the following commands:
 - a. If you are using failover file system for the runtime files, type `/local/im/`.
 - b. If you are using a cluster file system for the runtime files, type `/global/im/`.
3. If prompted for the Instant Messaging host name, type the logical host. Choose to accept the logical host even if the `configure` utility is unable to connect to the specified host. The logical host resource might be offline at the time when you invoke the `configure` utility.
4. Do not start Instant Messaging after configuration or on system startup.
5. Change the value of the `iim_wd.period` parameter to "60" and the `iim_wd.maxRetries` parameter to "2" in the `IM_RUNTIME_CONFIG/iim.conf` file.
6. Create and enable the Instant Messaging resource.
In this example, the resource group name is `IM_SVR_RS`. Provide the logical host resource name and the `HASStoragePlus` resource name. For example,

```
./scrgadm -a -j IM_SVR_RS -g IM-RG
-t SUNW.iim -x Server_root=/IM_SVR_BASE
-x Confdir_list=/IM_RUNTIME_CONFIG (ex:
/local/im/default/config )
-y Resource_dependencies=IM_HASP_RS,LOG_HOST_RS

./scrgadm -e -j IM_SVR_RS
```

7. Test the successful creation of the Instant messaging resource group by performing a failover.
`./scswitch -z -g IM-RG -h IM_NODE2`

Note

You do not need to configure the second node as the configuration is shared between all the nodes by soft links pointing to the shared location.

Installing and Configuring in a Symmetric HA Environment

This section contains instructions for configuring a symmetric high availability Instant Messaging system. To configure a symmetric high availability Instant Messaging system, perform the steps described in the following sections:

- Initial Tasks
- Installing and Configuring the First Instance of Instant Messaging
- Installing and Configuring the Second Instance of Instant Messaging

Initial Tasks

You must complete the following preparatory tasks before installing Instant Messaging on the nodes. The preparatory tasks are:

- I. Creating File Systems
- II. Installing the Instant Messaging HA Package
- III. Preparing Each Node of the Cluster

I. Creating File Systems

Instant Messaging binaries, configuration files, and runtime files reside on the CFS or on the highly available FFS. For each Instant Messaging instance, installation is needed on only one node as the binaries are shared across all the nodes.

To create file systems, perform the following steps:

1. Create four file systems by using CFS or FFS.
To create a system by using CFS, for example, the contents of the `/etc/vfstab` file should appear as follows.

```
# Cluster File System/Global File System ##
/dev/md/penguin/dsk/d500 /dev/md/penguin/rdisk/d500
  /INSTALL-ROOTIM1 ufs 2 yes logging,global
/dev/md/penguin/dsk/d400 /dev/md/penguin/rdisk/d400
  /share-disk-dirIM1 ufs 2 yes logging,global
/dev/md/polarbear/dsk/d200 /dev/md/polarbear/rdisk/d200
  /INSTALL-ROOTIM2 ufs 2 yes logging,global
/dev/md/polarbear/dsk/d300 /dev/md/polarbear/rdisk/d300
  /share-disk-dirIM2 ufs 2 yes logging,global
```

Note
The fields must be separated by tabs.

To create a system by using FFS, for example, the contents of the `/etc/vfstab` file should appear as follows.

```
# Failover File System/Local File System ##
/dev/md/penguin/dsk/d500 /dev/md/penguin/rdisk/d500
  /INSTALL-ROOTIM1 ufs 2 yes logging
/dev/md/penguin/dsk/d400 /dev/md/penguin/rdisk/d400
  /share-disk-dirIM1 ufs 2 yes logging
/dev/md/polarbear/dsk/d200 /dev/md/polarbear/rdisk/d200
  /INSTALL-ROOTIM2 ufs 2 yes logging
/dev/md/polarbear/dsk/d300 /dev/md/polarbear/rdisk/d300
  /share-disk-dirIM2 ufs 2 yes logging
```

Note
The fields must be separated by tabs.

2. Create the following mandatory directories on all the nodes of the cluster.

```
# mkdir -p /INSTALL-ROOTIM1 share-disk-dirIM1
INSTALL-ROOTIM2 share-disk-dirIM2
```

II. Installing the Instant Messaging HA Package

Install the Instant Messaging Oracle Solaris Cluster HA package in two nodes. You can use the Communication Suite 6 Update 1 installer to install the HA package.

To install the Instant Messaging Oracle Solaris Cluster HA agent, perform the following steps:

1. a. Run the Communications Suite 6 installation command.
`commpkg install`



Note

In case of Oracle Solaris 10 zone, run the above command from the global and non-global zones.

2. When prompted, select the Instant Messaging Oracle Solaris Cluster HA Agent software.
3. Type the Sun Cluster HA Agent pre-configuration command.

```
# <IM_SCHA_BASE>/bin/init-config
```



Note

In case of Oracle Solaris 10 zone, run the above command only from the global zone.

III. Preparing Each Node of the Cluster

For each node in the cluster, create the Instant Messaging runtime user and group under which the components will run. The user ID (UID) and group ID (GID) numbers must be the same on all nodes in the cluster.

- Runtime User ID: User name using which the Instant Messaging server runs. The default value is `inetuser`.
 - Runtime Group ID: Group using which the Instant Messaging server runs. The default value is `inetgroup`.
- Although the `configure` utility creates these IDs, you can create the IDs before you invoke the `configure` utility as part of the preparation of each node. Create the runtime user and group ID on a node where you might not invoke the `configure` utility, which is usually secondary node.

Make sure that the username, group name and the corresponding user ID and group ID are same in the following files on all nodes:

- `inetuser` or the name that you select in the `/etc/passwd` directory on all the nodes in the cluster
- `inetgroup` or the name that you select in the `/etc/group` directory on all the nodes in the cluster

Refer to your operating system documentation for detailed information about users and groups.

Installing and Configuring the First Instance of Instant Messaging

To install the first instance of Instant Messaging, perform the following steps:

1. Verify whether the files are mounted.
On the primary node `Node1`, type the following command:
`df -k`

The following message shows a sample output:

```
/dev/md/penguin/dsk/d500      35020572
    34738 34635629    1%  /INSTALL-ROOTIM1
/dev/md/penguin/dsk/d400      35020572
    34738 34635629    1%  /share-disk-dirIM1
```

2. Using the Communications Suite 6 installer, install Instant Messaging on the primary node.
 - a. Type the Communications Suite installer command.

```
# commpkg install
```



Note

In case of Oracle Solaris 10 zones, refer to the Communications Suite installation documentation.

- b. At the Specify Installation Directories prompt, type the installation root `INSTALL-ROOTIM1`.
3. Create a symbolic link from the Instant Messaging the `/etc/opt/SUNWiim` directory to the shared disk `IM_RUNTIME_DIR` directory on all the nodes of the cluster. For example, type the following commands on a cluster node:

```
# pwd
/etc/opt/

# ln -s /share-disk-dirIM1 SUNWiim
```

To configure Oracle Solaris Cluster on the first node by using the Oracle Solaris Cluster command-line interface, perform the following steps:

1. Register the following resource types.

```
./scrgadm -a -t SUNW.HAStoragePlus
./scrgadm -a -t SUNW.iim
```

2. Create a failover resource group.
In the following example, the resource group is `IM-RG1`, `IM_NODE1` is the primary node and `IM_NODE2` is the failover node.
`./scrgadm -a -g IM-RG1 -h IM_NODE1,IM_NODE2`
3. Create a logical hostname resource for the node.
Add the logical host name `LOG_HOST_RS` to the resource group. Instant Messaging listens on this hostname. The following example uses `LOG-HOST-IM-RS1`. Replace this value with the actual hostname.

```
./scrgadm -a -L -g IM-RG1 -l LOG-HOST-IM-RS1
./scrgadm -c -j LOG-HOST-IM-RS1 -y R_description=
    "LogicalHostname resource for LOG-HOST-IM-RS1"
```

4. Bring the resource group online.
`scswitch -Z -g IM-RG1`
5. Create a `HAStoragePlus` resource and add it to the failover resource group.
In this example, the resource is called `IM_HASP_RS1`. Replace the resource with your own resource name.

**Note**

The example is split for display purpose in this document.

```
./scrgadm -a -j IM-HASP-RS1 -g IM-RG1 -t
  SUNW.HAStoragePlus:4 -x
FilesystemMountPoints=/INSTALL-ROOTIM1,
  /share-disk-dirIM1
./scrgadm -c -j IM-HASP-RS1 -y R_description="Failover data
  service resource for SUNW.HAStoragePlus:4"
```

6. Enable the HAStoragePlus resource.

```
./scswitch -e -j IM-HASP-RS1
```

To configure the first instance of Instant Messaging, perform the following steps:

1. Run the `configure` utility on the primary node.

```
# cd /INSTALL-ROOTIM1/im
# ./configure
```

For more information about the `configure` utility, see [configure Utility](#).

2. When prompted for the Instant Messaging Server Runtime Files Directory, type `/share-disk-dirIM1` if you are using HAStoragePlus for the runtime files.
3. When prompted for the Instant Messaging host name, type the logical host. Choose to accept the logical host even if the `configure` utility cannot connect to the specified host. The logical host resource might be offline at the time when you invoke the `configure` utility.
4. Do not start Instant Messaging after configuration or on system startup.
5. Change the value of the `iim_wd.period` parameter to "60" and the `iim_wd.maxRetries` parameter to "2" in the `IM_RUNTIME_CONFIG/iim.conf` file.
6. Create and enable the Instant Messaging resource. In this example, the resource group name is `IM_SVR_RS1`. Provide the logical host resource name and the HAStoragePlus resource name.

```
./scrgadm -a -j IM_SVR_RS1 -g IM-RG1
  -t SUNW.iim -x Server_root=/INSTALL-ROOTIM1/im
  -x Confdir_list=/share-disk-dirIM1/default/config
  -y Resource_dependencies=IM-HASP-RS1,LOG-HOST-IM-RS1

./scrgadm -e -j IM_SVR_RS1
```

7. Test the successful creation of the Instant Messaging resource group by performing a failover.

```
./ scswitch -z -g IM-RG1 -h IM_NODE2
```

**Note**

You do not have to configure the second node as configuration is shared between all the nodes by soft links pointing to shared location.

Installing and Configuring the Second Instance of Instant Messaging

To install the second instance of Instant Messaging, perform the following steps:

1. Verify whether the files are mounted. On the primary node IM_NODE2, type:

```
df -k
```

The following output is displayed:

```
/dev/md/polarbear/dsk/d300 35020572
34738 34635629 1% /share-disk-dirIM2
/dev/md/polarbear/dsk/d200 35020572
34738 34635629 1% /INSTALL-ROOTIM2
```

2. Using the Communications Suite 6 installer, install Instant Messaging on the primary node.
 - a. Type the Communications Suite installation command.

```
# commpkg install
```



Note

In case of Oracle Solaris 10 zones, refer to the Communications Suite installation documentation.

- b. At the Specify Installation Directories prompt, specify the installation root
INSTALL-ROOTIM2.
3. Create a symbolic link from the Instant Messaging `/etc/opt/SUNWiim` directory to the shared disk `IM_RUNTIME_DIR` directory on this cluster node.

For example, type the following commands on all the nodes of the cluster:

```
# pwd
/etc/opt/

# ln -s /share-disk-dirIM2 SUNWiim
```

Configuring Oracle Solaris Cluster on the Second Node

To configure Oracle Solaris Cluster on the second node by using the Oracle Solaris Cluster command-line interface, perform the following steps:

1. Create a failover resource group.
In the following example, the resource group is `IM-RG2`, `IM_NODE2` is the primary node and `IM_NODE1` is the failover node.

```
./scrgadm -a -g IM-RG2 -h IM_NODE2,IM_NODE1
```
2. Create a logical hostname resource for this node.
Add the logical host name `LOG_HOST_RS` to the resource group. Instant Messaging will listen on this host name. The following example uses `LOG-HOST-IM-RS2` in the place where you will substitute in the actual hostname.

```
./scrgadm -a -L -g IM-RG2 -l LOG-HOST-IM-RS2
./scrgadm -c -j LOG-HOST-IM-RS2 -y R_description=
    "LogicalHostname resource for LOG-HOST-IM-RS2"
```

3. Bring the resource group online.

```
scswitch -Z -g IM-RG2
```
4. Create a `HASStoragePlus` resource and add it to the failover resource group.
In this example, the resource is called `IM-HASP-RS2`. Replace it by your own resource name. Note that the lines are divided and show as two lines in the example for display purposes in this document.

```
./scrgadm -a -j IM-HASP-RS2 -g IM-RG2 -t
    SUNW.HAStoragePlus:4 -x
FilesystemMountPoints=/INSTALL-ROOTIM2,
    /share-disk-dirIM2
./scrgadm -c -j IM-HASP-RS2 -y R_description="Failover data
    service resource for SUNW.HAStoragePlus:4"
```

5. Enable the HAStoragePlus resource.

```
./scswitch -e -j IM-HASP-RS2
```

To configure the second instance of Instant Messaging, perform the following steps:

1. Run the `configure` utility on the primary node.

```
# cd /INSTALL-ROOTIM2/im
# ./configure
```

For more information about the `configure` utility, see [configure Utility](#).

2. When prompted for the Instant Messaging Server Runtime Files Directory, type one of the following:
If you are using an HAStoragePlus for the runtime files, type `/share-disk-dirIM2`
3. When prompted for the Instant Messaging host name, type the logical host.
For example, accept the logical host even if the `configure` utility cannot connect to the specified host. The logical host resource might be offline when you invoke the `configure` utility.
4. Do not start Instant Messaging after configuration or on system startup.
In an HA configuration, the Instant Messaging service requires the logical host to be online for Instant Messaging to work correctly.
5. Change the value of the `iim_wd.period` parameter to "60" and the `iim_wd.maxRetries` parameter to "2" in the `IM_RUNTIME_CONFIG/iim.conf` file.
6. Create the Instant Messaging resource and enable the resource.
In this example, the resource group name is `IM_SVR_RS2`. Provide the logical host resource name, the HAStoragePlus resource name, and the port number. By default, Instant Messaging uses ports 5269, 5222 and 45222. If the first instance uses these port numbers, use different port numbers for the second instance.

```
./scrgadm -a -j IM_SVR_RS2 -g IM-RG2
    -t SUNW.iim -x Server_root=/INSTALL-ROOTIM2/im
    -x Confdir_list=/share-disk-dirIM2/default/config
    -y Resource_dependencies=IM-HASP-RS2,LOG-HOST-IM-RS2
    -x port_list="5270/tcp", "5223/tcp", "45223/tcp"
```

7. Test the successful creation of the Instant messaging resource group by performing a failover.

```
./scswitch -z -g IM-RG2 -h IM_NODE1
```



Note

You do not have to configure the second node as configuration is shared between all the nodes by soft links pointing to shared location.

Starting and Stopping the Instant Messaging HA Service

To start and stop the Instant Messaging HA service, use the Oracle Solaris Cluster `scswitch` command.



Caution

Do not use the `imadmin start`, `imadmin stop`, or `imadmin refresh` commands in a HA environment with Sun Cluster. Instead, use the Sun Cluster administrative utilities. For more information about the Oracle Solaris Cluster `scswitch` command, refer to the Oracle Solaris Cluster Reference Manual for Solaris OS.

To start the Instant Messaging HA service, type the following command:

```
# scswitch -e -j IM_SVR_RS
```

To stop the Instant Messaging HA service, type the following command:

```
# scswitch -n -j IM_SVR_RS
```

To restart the Instant Messaging HA Service, type the following command:

```
# scswitch -R -j IM_SVR_RS
```

Troubleshooting the Instant Messaging HA Configuration

To help troubleshooting error messages are stored in the error log. The logs are controlled by the `syslog` facility. For information about using the logging facility, see the [HA Related Documentation](#) and the `syslog.conf` man page.

Removing HA for Instant Messaging

In order to remove Instant Messaging from an HA environment, you need to remove the Instant Messaging cluster agent `SUNWiimsc` as described in this section.

To Remove HA for Instant Messaging Before You Begin

When you remove the `SUNWiimsc` package as described in this procedure, any customizations you made to the RTR file `SUNW.iim` are lost. If you want to restore them at a later time, you need to create a backup copy of `SUNW.iim` before removing the `SUNWiimsc` package.

1. Bring down the Instant Messaging data service:

```
scswitch -F -g im-resource-group
```

2. Disable all resources in the Instant Messaging resource group (*im-resource-group*):

```
# scswitch -n -j im-resource
# scswitch -n -j im-logical-host
# scswitch -n -j im-resource-group-store
```

3. Remove the resources from the Instant Messaging resource group:

```
# scrgadm -r -j im-resource
# scrgadm -r -j im-logical-host
# scrgadm -r -j im-resource-group-store
```

4. Remove the Instant Messaging resource group:

```
# scrgadm -r -g im-resource-group
```

5. Remove the Instant Messaging resource type:

```
# scrgadm -r -t SUNW.iim
```

6. Remove the SUNWiimsc package using the Java Enterprise System installer or manually as follows:

```
pkgrm SUNWiimsc
```

When you remove the package, any customizations you made to the RTR file are lost.

7. If you are using a shared directory for configuration files and binaries, remove any soft links created during HA configuration.

On Node 1:

```
rm /etc/opt/SUNWiim
```

On all other nodes:

```
rm /usr/cluster/lib/rgm/rtreg/SUNW.iim
```

HA Related Documentation

A list of high availability documentation is as follows:

- [Sun Java Enterprise System 5 Update 1 Technical Overview](#)
- [Sun Java Enterprise System 5 Update 1 Installation Guide for UNIX](#) describes the Java Enterprise System installer, uninstaller, and the supported installation scenarios.
- [Sun Java Enterprise System 5 Update 1 Release Notes](#) provide current information about the Sun Java Enterprise System product.
- [Sun Cluster Concepts Guide for Solaris OS](#) provides a general background about the Sun Cluster software, data services, terminology resource types, resources, and resource groups.
- [Sun Cluster Data Services Planning and Administration Guide for Solaris OS](#) provides general information about planning and administration of data services.
- [Sun Cluster System Administration Guide for Solaris OS](#) provides software procedures for administering a Sun Cluster configuration.
- [Sun Cluster Reference Manual for Solaris OS](#) describes commands and utilities existing with the Sun Cluster software, including commands found only in the SUNWscman and SUNWccon packages.
- [Sun Java System Communications Services 6 2005Q4 Deployment Planning Guide](#) provides further information about how HA is implemented in Instant Messaging.

Chapter 4. Instant Messaging Configuration File and Directory Structure Overview

Oracle Communications Instant Messaging Server Configuration File and Directory Structure Overview

This information describes the configuration files you use to administer Instant Messaging. Familiarize yourself with the locations of these files before making changes to your deployment's configuration. This information also describes the Instant Messaging server directory structure and the properties files used to store Instant Messaging operational data and configuration information.

Topics:

- [Instant Messaging Server Directory Structure](#)
- [Instant Messaging Server Configuration File](#)
- [Instant Messaging Data](#)

Instant Messaging Server Directory Structure

[Instant Messaging Server Directory Structure](#) shows the platform-specific directory structure for the Instant Messaging server.

Table 3-1 Instant Messaging Server Directories

Description	Oracle Solaris Location	Red Hat Linux Location
<p>Program Files</p> <p>These files include the native executable files, the library files in the bin or lib directory, the shell scripts in the sbin directory, the Java classes, and templates files in the lib directory.</p>	<p>Instant Messaging Installation Directory</p> <p>The default value for the Installation Directory is: <code>/opt/sun/comms/im</code></p>	<p>Instant Messaging Installation Directory</p> <p>The default value for the Installation Directory is: <code>/opt/sun/comms/im</code></p>

<p>Server Configuration files</p> <p>These files are in the Configuration Directory and include the iim.conf file and a subdirectory which contains all the server-wide access control files.</p>	<p>Instant Messaging Configuration Directory</p> <p>The default value for the Configuration Directory is: /etc/opt/SUNWiim/default/config</p> <p>For convenience, the installer creates a symbolic link from /etc/opt/SUNWiim/default/config to /opt/SUNWiim/config.</p> <p>In addition, if you create multiple instances of Instant Messaging, the name of the /default directory will vary depending on the instance. See Creating Multiple Instances from a Single Instant Messaging Installation for more information.</p>	<p>Instant Messaging Configuration Directory</p> <p>The default value for the Configuration Directory is: /etc/opt/sun/im/default/config</p> <p>For convenience, the installer creates a symbolic link from /etc/opt/sun/im/default/config to /opt/sun/im/config.</p> <p>In addition, if you create multiple instances of Instant Messaging, the name of the /default directory will vary depending on the instance. See Creating Multiple Instances from a Single Instant Messaging Installation for more information.</p>
<p>Runtime Directory Contains Instant Messaging server data.</p> <p>These files include the configurable directory for the files generated by the server at runtime. It includes the end user data in the data directory. It also contains the server, multiplexor, Calendar agent, and XMPP service log files, in the log directory.</p>	<p>Instant Messaging Runtime Directory</p> <p>The default value for the Runtime Directory is: /var/opt/SUNWiim/default</p> <p>In addition, if you create multiple instances of Instant Messaging, the name of the /default directory will vary depending on the instance. See Creating Multiple Instances from a Single Instant Messaging Installation for more information.</p>	<p>Instant Messaging Runtime Directory</p> <p>The default value for the Runtime Directory is: /var/opt/sun/im/default</p> <p>In addition, if you create multiple instances of Instant Messaging, the name of the /default directory will vary depending on the instance. See Creating Multiple Instances from a Single Instant Messaging Installation for more information.</p>
<p>Database</p> <p>If you are using a file-based property store, the database directory contains end user information such as the user and news channels directory. If you are using LDAP to store user data, the database directory is not used.</p>	<p>Instant Messaging Database Directory</p> <p>The default value for the Database Directory is: /var/opt/SUNWiim/default/db</p> <p>In addition, if you create multiple instances of Instant Messaging, the name of the /default directory will vary depending on the instance. See Creating Multiple Instances from a Single Instant Messaging Installation for more information.</p>	<p>Instant Messaging Database Directory</p> <p>The default value for the Database Directory is: /var/opt/sun/im/default/db</p> <p>In addition, if you create multiple instances of Instant Messaging, the name of the /default directory will vary depending on the instance. See Creating Multiple Instances from a Single Instant Messaging Installation for more information.</p>

<p>Instant Messenger resources.</p> <p>These files contain HTML documents and jar files used by Instant Messenger. The topmost directory contains the locale-independent resources, and the locale-specific directories contain the localized resources.</p>	<p>Instant Messaging Resource directory.</p> <p>The default value for the Resource Directory is: /opt/sun/comms/im/html</p>	<p>Instant Messaging Resource directory</p> <p>The default value for the Resource Directory is: /opt/sun/im/html</p>
--	---	--

Instant Messaging Server Configuration File

Instant Messaging stores all configuration options in the `iim.conf` file. For more information on the parameters and their values stored in this file, see [Instant Messaging Configuration Parameters in `iim.conf`](#).

Instant Messaging Data

Instant Messaging server stores the following data used by Instant Messenger in the database directory, and is indicated by the `iim.instancevardir` parameter in `iim.conf`:

- End user properties, such as contact lists, messenger settings, subscribed news channels and access control (alternatively, these properties can be stored in LDAP).
- News channel messages and access rules.
- Alert Messages that are to be delivered. These messages are delivered and removed when the recipient logs in.
- Public conferences. This does not involve instant messages which are not persistent, but only properties of the conference objects themselves, such as access rules.

Chapter 5. Enabling Single Sign-On (SSO) for Instant Messaging

Enabling Single Sign-On (SSO) for Oracle Communications Instant Messaging Server

This information describes using Access Manager to enable SSO for Instant Messaging.

Topics:

- [Enabling Single Sign-On \(SSO\) Overview](#)
- [SSO Limitations and Notices](#)
- [Configuring Instant Messaging to Support Access Manager-Based SSO and Policies](#)
- [Troubleshooting SSO for Instant Messaging](#)

Enabling Single Sign-On (SSO) Overview

Single sign-on is the ability for an end user to authenticate once (that is, log on with user ID and password) and have access to multiple applications. Access Manager is the official gateway used for SSO for Unified Communication Suite servers. That is, users must log into Access Manager to get access to other SSO configured servers.

For example, when properly configured, a user can sign in at the Access Manager login screen and have access to Instant Messenger in another window without having to sign in again. Similarly, if Calendar Server is properly configured, a user can sign in at the Access Manager login screen, then have access to Calendar in another window without having to sign in again.

Other Communications Suite servers, such as Messaging Server, provide two methods of deploying SSO. The first way is through the Access Manager, the second way is through trusted circle technology. Using a trusted circle is the legacy method of implementing SSO, and is not used by Instant Messaging. Though this method provides some features not available with Access Manager SSO, all future development will be with the Access Manager.

SSO Limitations and Notices

- The Instant Messenger session is only valid for as long as the Access Manager session is valid. If the user logs out of Access Manager the Instant Messenger session is automatically closed (single sign-off) as soon as the user sends another request to the server.
- SSO applications working together must be in the same DNS domain.
- SSO applications must have access to the Access Manager verification URL (naming service).
- Browsers must have cookies enabled.
- Single Sign On (SSO) cannot be enabled unless the policy is `identity`. When SSO is selected, set the value of the `iim.policy.modules` parameter to `identity` in the `iim.conf` file.

Configuring Instant Messaging to Support Access Manager-Based SSO and Policies

Two `iim.conf` parameters support Instant Messaging SSO.

Table 5-1 Instant Messaging Single Sign-On Parameters

Parameter	Description
<i>iim_server.usesso</i>	Determines whether or not the Instant Messaging server should depend on the SSO provider during authentication. The Access Manager Session API provides the Instant Messaging server with the ability to validate session IDs sent by the client. Possible values include: 0 - Do not use the SSO provider. 1 - Use the SSO provider first and default to LDAP if the SSO validation fails. -1 - Use only the SSO provider without attempting LDAP authentication even when SSO authentication fails. Default: 1 if you chose to leverage Access Manager for SSO when you ran the <code>configure</code> utility. Otherwise, the default value is 0.
<i>iim_server.ssoprovider</i>	Specifies the class implementing the <code>com.sun.im.provider.SSOProvider</code> interface. If <i>iim_server.usesso</i> is not equal to 0 and this option is not set, the server uses the default Access Manager-based SSO Provider that is internally defined in Instant Messaging. Typically, you will not modify this parameter. Default: None

To Enable SSO for Instant Messaging

1. Ensure that the Access Manager SDK is installed on the same host as the Instant Messaging server.
See [Sun Java Communications Suite 5 Installation Guide](#) for more information.
2. Ensure that Instant Messaging services are assigned to the organization in the Access Manager console (`amconsole`).
If you are using other Communications Suite server products in your deployment, such as Messaging Server, you may need to manually configure Access Manager-based services for Instant Messaging.
See [Adding Instant Messaging and Presence Services to a Sub-organization in Access Manager for Single Sign-On and Policy Management Support](#) for instructions.
3. Run the `configure` utility.
See [configure Utility](#) for instructions.
4. When prompted whether you want to use Access Manager for SSO, select yes.
5. Set the `iim.policy.module` parameter to `identity` in the `iim.conf` file.

```
iim.policy.module = "identity"
```
6. Restart the Instant Messaging server by typing

```
imadmin start
```

Troubleshooting SSO for Instant Messaging

If there is a problem with SSO, the first thing to do is check the `xmppd.log` server log file and the client log files for errors. Increasing the logging level may be helpful. New logging levels will only take effect after server restart.

Ensure that Instant Messaging services have been assigned to the organization and its parent organization in the Access Manager console (`amconsole`). See [Adding Instant Messaging and Presence Services to a Sub-organization in Access Manager for Single Sign-On and Policy Management Support](#) for information.

Ensure that the `iim_server.usesso` parameter is not set to 0 in `iim.conf`. See [Table 5-1 Instant Messaging Single Sign-On Parameters](#) for information on this parameter. If it is set to 0, complete the steps in [To Enable SSO for Instant Messaging](#).

If you are unable to log into Instant Messaging directly, see the `xmppd.log` file for an error similar to either of the following:

```
DEBUG xmppd [com.sun.im.service.util.Worker3] Service      \\
URL not found:session.com.iplanet.sso.SSOException: Service URL not
found:
```

```
INFO xmppd [com.sun.im.service.util.Worker 3] [Identity]  \\
Failed to create SSO token for USERNAME
```

```
INFO xmppd [org.netbeans.lib.collab.util.Worker 1] [LDAP]  \\
pops does not have required objectclass for storing to ldap
```

If any of these errors exist, perform the following steps to solve the problem:

1. Create a user through `amconsole` and add authentication, configuration, Instant Messaging, and presence services to the user.
2. Attempt to log in with the user you created.
3. Check to ensure that the `amldapuser`'s password is correctly filled in through `amconsole`.
4. Check whether the domain, for example, `o=siroe.com`, has the Authentication Configuration Service Instance.
5. Check if the Authentication Configuration Service Instance has the Authentication Module set to LDAP or Membership. The value should show a state of `REQUIRED/SUFFICIENT`. Instant Messaging only supports login with username and password. If you are using Auth-Chain, you need to disable it to use Instant Messaging.
6. In the LDAP or Authentication Module, enter the `amldapuser` password for `CORE`.
7. Select the newly created `ldapService` Authentication Configuration Service Instance under the Organization Authentication Configuration drop-down menu and the Administrator Authentication Configuration drop-down menu in the Core Authentication Module Configuration.
8. Log in again.

The `imadmin` command fails to bind and takes the Directory Manager password as input. To fix this issue, do the following:

1. Include the `iim_ldap.usergroupbindcred=password` parameter in the `iim.conf` file.
2. Type the `./imadmin assign_services` command to assign services to users in the LDAP directory.

Chapter 6. Configuring Hosted Domain Support

Setting Up and Configuring Hosted Domain Support

This information describes how to configure hosted domains for Instant Messaging.

Topics:

- [Hosted Domains Overview](#)
- [Setting Up Schema 1 and Schema 2 for Hosted Domains](#)
- [Setting Up a Hosted Domain Environment with Access Manager](#)
- [Cross Domain Searches](#)

Hosted Domains Overview

Instant Messaging server provides support for hosted domains. In a hosted domain installation, each domain shares the same instance of the Instant Messaging server that enables multiple domains to exist on a single server. Each hosted domain has a name space that can contain unique users, groups, resources, preferences, and attributes.

Starting with **Instant Messaging 8**, Access Manager is no longer required when implementing hosted domains.

Setting Up Schema 1 and Schema 2 for Hosted Domains

Instant Messaging Server supports two schema versions: Schema 1 and Schema 2. This section describes the steps to set up the schema for hosted domains.

Schema 1 Structure

The directory structure of Schema 1 includes two trees for domain management: the organization tree and the domain component (DC) tree. For example, for a xyz.abc.com domain, the tree structure is as follows:

```
A, dc tree: o=internet // dc tree root suffix
dc=com
dc=abc
dc=xyz // domain node
```

The domain should contain the following attributes:

- `objectclass=inetDomain`
- `inetDomainBaseDn=o=xyz.abc.com`
- `dc=xyz,dc=abc, dc=com`

`inetDomainBaseDn` is a mandatory attribute for the `inetDomain` object class. You should also specify the status of the `inetDomainStatus` attribute as active.

`o=xyz.abc.com, dc=xyz,dc=abc,dc=com` is the domain name of organization in the organization tree that contains the users for the domain `xyz.abc.com`.

To Configure Instant Messaging for Schema 1

1. To configure the Instant Messaging configuration for Schema 1, set the following parameters in the `iim.conf` file:

```
iim.policy.modules = "iim_ldap_schema1"  
iim_ldap.schema1.domain_config_root = "<value>"
```

`iim_ldap.schema1.domain_config_root` should be the domain component tree root suffix. For example, `o=internet`.

2. Make sure to either remove or comment out the following line:

```
iim.policy.modules = "iim_ldap"
```

3. If you want to log in by using Convergence, edit the Convergence `httpbind.conf` file to include both default domain and hosted domains.

For example:

```
default.domains=xyz.abc.com, other.hosteddomain.com
```

You should then be able to log in to Convergence as `user@hosteddomain`. The default domain user can log in with just the uid.

Schema 2 Structure

Schema 2 has only the domain component as the `config` root. Schema 2 has the following tree structure:

```
B, Organization tree: dc=xyz,dc=abc,dc=com // Base dn for users/groups  
o=xyz.abc.com  
ou=people // Users are under this node
```

To Configure Instant Messaging for Schema 2

1. To configure the Instant Messaging configuration for Schema 2, set the following parameters in the `iim.conf` file:

```
iim.policy.modules = "iim_ldap_schema2"  
iim_ldap.schema2.domain_config_root = "<value>"
```

If the default value of the `iim.policy.modules` parameter is `iim_ldap`, the users under the non-default domain cannot be searched. Users cannot log in to the Instant Messaging server. The Instant Messaging server, in this case, does not go through the domain component tree to find the value of the `inetDomainBaseDn` attribute. The server uses the value of the `iim_ldap.searchbase` attribute to search users who exists in the default domain. You can specify the default domain by using the `iim_server.domainname` attribute.

`iim_ldap.schema2.domain_filter` specifies the object class of the domain node. The default value is `inetDomain`. `iim_ldap.schema2.domain_config_root` should be the domain component tree root suffix. For example, `dc=xyz, dc=abc,dc=com`.

2. If you want to log in by using Convergence, edit the Convergence `httpbind.conf` file to include both default domain and hosted domains.
For example:

```
default.domains=xyz.abc.com, other.hosteddomain.com
```

You should then be able to log in to Convergence as `user@hosteddomain`. The default domain user can log in with just the uid.



Note

Instant Messaging does not provide a tool to create these topologies.

Setting Up a Hosted Domain Environment with Access Manager

This section describes the steps to set up a hosted domain environment with Access Manager.

Prerequisites

Make sure that you install Access Manager on the machine where you have installed the Instant Messaging server.

To set up a hosted domain environment if you have installed Access Manager, perform the following steps:

1. Log in as `admin` to Access Manager.
2. Create a new domain, for example `siroe.com`, under the top-level organization tree by clicking the New Organization button.
`dc=siroe,dc=com`
3. Assign all the services to the newly created domain.
4. Create users under this domain and assign the required services to all users.
 - a. Go to the `/opt/sun/comms/im/html/en/` directory and type the following command:
`cp im.jnlp siroe.jnlp`
 - b. Add the following parameters in the `siroe.jnlp` file:

```
href="en/siroe.jnlp"  
<argument>domain=siroe.com</argument>
```



Note

Change all the other occurrences of `im.jnlp` to `siroe.jnlp` in this file.

- c. Type the `/opt/sun/comms/im/sbin/iwadmin redeploy im` command.
 - d. Restart Web Server and the Instant Messaging server.
 - e. Start the Instant Messaging client by typing
`http://foo.sun.com:1234/im/en/siroe.jnlp` in your web browser.
5. Make sure that the following parameters exist in the `iim.conf` file.

```
iim_ldap.useidentityadmin = "true"
iim_server.usesso = "1"
iim.policy.modules = "identity"
iim.userprops.store = "ldap"
```



Note

If you want to log in as `username@domain.com`, add the

```
<property name="com.ipplanet.im.client.allowarobase"
value="true" />
```

parameter in the `im.jnlp` file and set it to `true`. The Instant Messaging client will not parse the `@` symbol while logging in.

Cross Domain Searches

Cross domain search functionality enables users in one domain to search for users and groups in other domains. To enable the cross domain search functionality for contacts and conferences, perform the following steps:

1. Open the `iim.conf` file and make sure that the following parameters exist:

```
iim_server.discofilter.principal.any = true
iim_server.discofilter.conference.any = true
iim_server.discofilter.domains.any=true
```

2. Add the following parameter in the `iim.conf` file. This parameter loads the specified domains in the server memory when the server starts.

```
iim_server.default_domains=abc.com,xyz.com,cde.com, fgh.com
```

You can separate the domain names by using a `,` (comma) or a (blank space).

Chapter 7. Administering Instant Messaging Components

Administering Oracle Communications Instant Messaging Server Components

This information explains how to administer the Instant Messaging components (server, multiplexor, Calendar agent, cluster agent, watchdog, sms-gateway, MSN gateway, AIM gateway, and Yahoo gateway) and perform other administrative tasks, such as changing configuration parameters and creating backups.

As of Instant Messaging 9.0.1.4.0, the AIM, MSN, and Yahoo gateways are deprecated and may be removed in a future release.

Topics:

- [Stopping, Starting, Refreshing, and Checking Instant Messaging Components](#)
- [Changing Instant Messaging Server and Multiplexor Configuration Parameters](#)
- [Backing Up Instant Messaging Data](#)

Stopping, Starting, Refreshing, and Checking Instant Messaging Components

The `imadmin` command enables you to:

- Start and stop all Instant Messaging components (server, multiplexor, Calendar agent, cluster agent, watchdog, sms-gateway, MSN gateway, AIM gateway, and Yahoo gateway).

As of Instant Messaging 9.0.1.4.0, the AIM, MSN, and Yahoo gateways are deprecated and may be removed in a future release.

- Start and stop an individual Instant Messaging component.
- Refresh all Instant Messaging component configurations.
- Refresh an individual Instant Messaging component.
- Check the status of Instant Messaging components.

The `imadmin` command-line utility can be executed only by `root` or a user who has administration rights to the system(s) on which the Instant Messaging server and multiplexor are running. This end user is typically the identity that the server runs as, and is designated during installation:

- On Oracle Solaris - `inetuser`
- In a deployment with Sun Java System Access Manager, if the Sun Java System Portal Server and the Instant Messaging server are installed on the same host, the user is the one who is running the Access Manager, as `root`.

The `imadmin` command-line utility is located in the following directory:
`im-svr-base/sbin`

Starting the Instant Messaging server enables Instant Messenger to connect to it. Stopping the Instant Messaging server closes all connections and disconnects all Instant Messenger clients.

Using Service Management Framework (SMF)

Instant Messaging supports the Service Management Framework (SMF) for stopping and starting Instant Messaging. On supported platforms, an SMF service is registered when you install Instant Messaging server. You can use either the `svcadm` command or the Instant Messaging `imadmin` utility to start and stop Instant Messaging processes.

To start and stop Instant Messaging by using the `svcadm` command:

```
svcadm enable svc:/application/sunim
svcadm disable svc:/application/sunim
```

To check the status of the Instant Messaging service :

```
svcs sunim
```

SMF related log messages can be found in `/var/svc/log/application-sunim:default.log/`.

Starting Instant Messaging Components

You can start all the components together or a single component separately.

Use the `imadmin` command with the `start` option to start the Instant Messaging Server, multiplexor, Calendar agent, cluster agent, watchdog, sms-gateway, MSN gateway, AIM gateway, and Yahoo gateway depending on which components are enabled.

As of Instant Messaging 9.0.1.4.0, the AIM, MSN, and Yahoo gateways are deprecated and may be removed in a future release.

To Start All Components

- At the command line, type the following:
`imadmin start`

If both server and multiplexor are enabled, this command first starts the Instant Messaging server, and then starts the multiplexor.

If the watchdog is enabled (default), this command starts the watchdog, then the watchdog reads the configuration file and starts the Instant Messaging Server and/or multiplexor as necessary.

To Start a Single Component

- At the command line, type the `imadmin start` command with an argument that designates the component as follows:
Server:
`imadmin start server`
Multiplexor:
`imadmin start multiplexor`
Calendar agent:
`imadmin start agent-calendar`
Watchdog:
`imadmin start watchdog`

SMS Gateway:

```
imadmin start sms-gateway
```

MSN Gateway:

```
imadmin start msn-gateway
```

As of Instant Messaging 9.0.1.4.0, the MSN gateway is deprecated: it may be removed in a future release.

AIM Gateway:

```
imadmin start aim-gateway
```

As of Instant Messaging 9.0.1.4.0, the AIM gateway is deprecated: it may be removed in a future release.

Yahoo Gateway:

```
imadmin start yim-gateway
```

As of Instant Messaging 9.0.1.4.0, the Yahoo gateway is deprecated: it may be removed in a future release.

Stopping Instant Messaging Components

You can stop all the components together or a single component separately.

Use the `imadmin` command with the `stop` option to stop the Instant Messaging Server, multiplexor, Calendar agent, cluster agent, watchdog, sms-gateway, MSN gateway, AIM gateway, and Yahoo gateway depending on which components are enabled.

As of Instant Messaging 9.0.1.4.0, the AIM, MSN, and Yahoo gateways are deprecated and may be removed in a future release.

To Stop All Components

- At the command line, type the following:

```
imadmin stop
```

If the watchdog is running, `imadmin` brings the watchdog down first, and then stops the server and/or the multiplexor.

This command stops the server, multiplexor, Calendar agent, cluster agent, watchdog, sms-gateway, MSN gateway, AIM gateway, and Yahoo gateway, terminates all end user connections, and disconnects any inbound and outbound servers configured.

As of Instant Messaging 9.0.1.4.0, the AIM, MSN, and Yahoo gateways are deprecated and may be removed in a future release.

To Stop a Single Component

- At the command line, type the `imadmin stop` command with an argument that designates the component as follows:
 Server:

```
imadmin stop server
```

 Multiplexor:

```
imadmin stop multiplexor
```

 Calendar agent:

```
imadmin stop agent-calendar
```

 Watchdog:

```
imadmin stop watchdog
```

Refreshing Component Configuration

Use the `imadmin` command with the `refresh` option to stop and restart an individual Instant Messaging component and refresh that component's configuration.

You can refresh all the components together or a single component separately.

Whenever you change a configuration parameter in the `iim.conf` file, you also need to refresh the configuration.

To Refresh All Components

- At the command line, type the following:

```
imadmin refresh
```

This command stops the server, multiplexor, Calendar agent, cluster agent, watchdog, sms-gateway, MSN gateway, AIM gateway, and Yahoo gateway, terminates all end user connections, and disconnects any inbound and outbound servers configured.

As of Instant Messaging 9.0.1.4.0, the AIM, MSN, and Yahoo gateways are deprecated and may be removed in a future release.

If the watchdog is running, `imadmin` brings the watchdog down first, and then stops the server and/or the multiplexor. Then starts the watchdog which reads the configuration file and starts the Instant Messaging server and/or multiplexor as necessary.

To Refresh a Single Component

- At the command line, type the `imadmin refresh` command with an argument that designates the component as follows:

```
Server:
imadmin refresh server
Multiplexor:
imadmin refresh multiplexor
Calendar agent:
imadmin refresh agent-calendar
Cluster agent:
imadmin refresh monitor
Watchdog:
imadmin refresh watchdog
```

Checking the Status of Instant Messaging Components

You can check the status of all the components together or a single component separately using the `imadmin` command with the `status` option. This command returns results in the following format:
Component [status]

For example:

```
Server           [UP]
Multiplexor      [UP]
Agent:calendar   [DOWN]
Watchdog         [UP]
```

To Check the Status of All Components

- At the command line, type the following:
`imadmin status`

This command returns the status of all enabled components.

To Check the Status of a Single Component

- At the command line, type the `imadmin status` command with an argument that designates the component as follows:
Server:
`imadmin status server`
Multiplexor:
`imadmin status multiplexor`
Calendar agent:
`imadmin status agent-calendar`
Watchdog:
`imadmin status watchdog`

Changing Instant Messaging Server and Multiplexor Configuration Parameters

Instant Messaging stores configuration parameters in the `iim.conf` file. For a complete list of configuration parameters, see [Instant Messaging Configuration Parameters in `iim.conf`](#).

To change configuration parameters, manually edit the configuration parameters and values in the `iim.conf` file, then refresh the Instant Messaging server configuration. If you change a multiplexor parameter, you only need to refresh the multiplexor as follows:

```
imadmin refresh multiplexor
```

For a complete list of parameters and their values, see [Instant Messaging Configuration Parameters in `iim.conf`](#).

To Change Configuration Parameters

1. Change to the `im-cfg-base` directory.
See [Instant Messaging Server Directory Structure](#) for instructions on locating `im-cfg-base`.
2. Edit `iim.conf` by using a text editor.
3. Save your changes.
4. Refresh the configuration by using `imadmin`.
For example: `imadmin refresh`



Note

If you change the multiplexor listen port (*iim_mux.listenport*) or the multiplexor host, update the *im.html* or the *im.jnlp* files accordingly. Failure to do so disables Instant Messenger from connecting to the server. For more information, see [Managing Instant Messenger](#).

Backing Up Instant Messaging Data

Instant Messaging does not come with any disaster recovery tools. Use your site's backup system to backup the configuration and database directories periodically. This section describes backing up Instant Messaging in the following sections:

- [Backup Information](#)
- [Performing a Backup](#)
- [Restoring Backup Information](#)

Backup Information

The Instant Messaging information that needs to be backed up are of the following types:

- Configuration Information
- Instant Messaging end user data
- Instant Messenger resources

The configuration information is stored in the configuration directory (*im-cfg-base*). The Instant Messaging data is stored in the database directory (*im-db-base*). Default paths are described in [Instant Messaging Server Directory Structure](#).

The Instant Messenger resources must be backed up if they have been customized. The location of the Instant Messenger resources are provided during installation.

Performing a Backup

While the configuration information does not change frequently, the Instant Messaging end-user data changes rapidly and to prevent any loss of end-user data you should back up the Instant Messaging end-user data on a periodic basis. You need to perform the backup before running the installation program and the uninstallation program.

To backup the end user data and the configuration information you do not have to stop the Instant Messaging server as all the disk commits by the server are automatically performed.

Restoring Backup Information

The backup of the end-user data and the configuration information needs to be restored when there is a disk failure and all the end-user data and the configuration information is lost.

To Restore End-user Data from Backup

1. Change to the *im-runtime-base* directory.
See [Instant Messaging Server Directory Structure](#) for information on locating *im-runtime-base*.
2. Stop the Instant Messaging server:
`imadmin stop`
3. Copy the backed up data to the *im-db-base* directory.
Be sure to maintain the directory structure of the backed-up data.

4. Verify the permissions and owner of the newly restored data.
The files should be owned by the Instant Messaging system user. See [Creating a UNIX System User and Group](#) for information on this user. Permissions should be set as follows:
 - a. Files: 600 (indicating read and write permissions for owner only)
 - b. Directories: 700 (indicating read, write, and execute permissions for owner only)Refer to your operating system documentation for information on changing permissions and owners.
5. Start the Instant Messaging server.
`imadmin start`

Chapter 8. Federating Deployment of Multiple Instant Messaging Servers

Federating Deployment of Multiple Oracle Communications Instant Messaging Servers

In an LDAP-only deployment, when you federate multiple Instant Messaging deployments you form a larger Instant Messaging community. End users from different servers can communicate with each other, use conference rooms on other domains, and subscribe to news channels on remote servers based on the access privileges.

For enabling communication between multiple Instant Messaging servers in your network, you need to configure your server to identify itself to the other Instant Messaging servers in the network. An Instant Messaging server identifies itself with its domain name, host and port number, server ID, and password.

In an LDAP-only deployment, the two servers should reside in different domains.

Within the server configuration, you can assign each Instant Messaging server a symbolic name, consisting of letters and digits, for example, `IMserver1`.



Note

Secure server-to-server communication using TLS. This is required to prevent third party infringement of security when data is exchanged between two servers. This precaution is extremely desirable in the case where the link between the two servers uses the public internet. Follow the instructions outlined below to configure TLS between Instant Messaging servers.



Note

You can use the server to server federation only if the servers are using the same protocol. IM servers use XMPP protocol. So, you may federate a server to server communication with GTalk or Openfire servers.

Configuring Federated Communication Between Instant Messaging Servers

This section describes how to enable federated communication between two Instant Messaging servers.

[Table 8-1](#) lists the parameters in `iim.conf` used to federate communication between two servers, and the values for these parameters for two example Instant Messaging servers; `iim.company22.com` and `iim.i-zed.com`.

For more information on the configuration parameters, see [Instant Messaging Configuration Parameters in `iim.conf`](#).

**Note**

Each Instant Messaging server is identified by its symbolic name. The symbolic name of the server is added in the `iim_server.coservers` parameter in `iim.conf`. This parameter has multiple values and each value is separated by a comma.

Table 8-1 Example Configuration Information for Two Federated Instant Messaging Servers

Parameter in <code>iim.conf</code>	Value for Server <code>iim.company22.com</code>	Value for Server <code>iim.i-zed.com</code>
<code>iim_server.serverid</code>	Iamcompany22	iami-zed
<code>iim_server.password</code>	secretforcompany22	secret4i-zed
<code>iim_server.coservers</code>	coserver1	coserver1
<code>iim_server.domainname</code>	iim.company22.com	iim.i-zed.com
<code>iim_server.coserver1.host</code>	iim.i-zed.com:5269	iim.company22.com:5269
<code>iim_server.coserver1.serverid</code>	Iami-zed	Iamcompany22
<code>iim_server.coserver1.password</code>	secret4i-zed	secretforcompany22
<code>iim_server.coserver1.domain</code>	i-zed.com	company22.com

To Federate Communication Between Two Instant Messaging Servers

1. Gather the information listed in [Table 8-1](#).
2. Change to `im-cfg-base` on the server `iim.company22.com`.
See [Instant Messaging Server Directory Structure](#) for instructions on locating `im-cfg-base`.
3. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.

**Note**

The `iim.conf` file should be owned by the Instant Messaging server account you created during installation. If the `iim.conf` file cannot be read by the Instant Messaging server account, the server and multiplexor will be unable to read the configuration. Additionally, you might lose the ability to edit `iim.conf`.

4. Modify the parameter values to match your deployment.
[Table 8-1](#) lists the parameters you need to modify. If the parameters do not exist in `iim.conf`, add them. The following example shows the section of `iim.conf` on `iim.company22.com` corresponding to the server-to-server communications that you need to modify:

```
iim_server.serverid=Iamcompany22
iim_server.password=secretforcompany22
iim_server.domainname=iim.icompany22.com
iim_server.coservers=coserver1
iim_server.coserver1.host=iim.i-zed.com:5269
iim_server.coserver1.serverid=Iami-zed
iim_server.coserver1.password=secret4i-zed
iim_server.coserver1.domain=i-zed.com
```

5. Follow steps 2 through 4 for the `iim.conf` file on server `iim.i-zed.com`.
The following example shows the section of `iim.conf` on `iim.i-zed.com` corresponding to the server-to-server communications that you need to modify:

```
iim_server.serverid=Iami-zed
iim_server.password=secret4i-zed
iim_server.domainname=iim.i-zed.com
iim_server.coservers=coserver1
iim_server.coserver1.host=iim.company22.com:5269
iim_server.coserver1.serverid=Iamcompany22
iim_server.coserver1.password=secretforcompany22
iim_server.coserver1.domain=company22.com
```

6. Save the changes and close `iim.conf`.
7. Refresh the configuration on both servers.

```
imadmin refresh server
```

Chapter 9. Optimizing an Instant Messaging Server Pool By Using the Redirect Server

Optimizing an Instant Messaging Server Pool By Using the Redirect Server

Use the redirect service that ships with Instant Messaging to balance the load between servers in a server pool (multi-node deployment). Performance is directly impacted by the amount of communication required between servers in a single deployment, so by increasing the probability that two users who will likely share presence information and messages end up on the same node, you improve performance.

This chapter contains information about using the Instant Messaging redirect server in the following sections:

- [Overview of Instant Messaging Redirect](#)
- [Configuring an Instant Messaging Server Instance as a Redirect Server](#)
- [Administering the Instant Messaging Redirect Server](#)
- [Creating and Managing the Instant Messaging Redirect Table Using the `rdadmin` Utility](#)
- [Instant Messaging Redirect Server Physical Host Monitoring](#)
- [Instant Messaging Redirect Server Best Practices and Troubleshooting](#)

Overview of Instant Messaging Redirect

The redirect server is an Instant Messaging server instance configured specifically to perform redirect tasks such as assigning connection end-points to Instant Messaging servers. Adding a redirect server to your deployment reduces the amount of communication between servers by grouping users who are likely to communicate with each other on the same host. This reduces the amount of presence notifications sent back and forth between servers in your deployment. Groups of users are determined by contact list contents. Shared entries in contact lists indicate a higher likelihood for communication.

Instant Messaging User Partitioning Algorithm

Instant Messaging determines the best division between users in your deployment and creates groups or partitions of users. The algorithm Instant Messaging uses is as follows:

1. Determine one or more sets of users, or user network, and their connections in your deployment. The redirect server then creates a table called the user-to-network map that maps each user to a user network.
2. Partition user networks that are larger than the maximum partition size along weakest ties, such that the maximum size of each weakly connected component is no larger than the configured partition size. Weak ties may be determined by a low number of connections between user networks, however, other parameters such as geographic constraints, number of connections per user network, and other constraints set by administrators may also be taken into account when partitioning user networks.
3. Distribute the sets into a specified number of partitions of roughly equal size. The redirect server first creates the network-to-partition table as part of this process and finally the user-to-partition table. These tables together make up the redirect database. The redirect database maps each user with a partition ID. You create and manage this database using the `rdadmin` command line utility.

Example 7-1 Instant Messaging Redirect Sequence of Events

This example describes the sequence of events that occur for a successful client redirect to take place.

1. Administrator runs the `rdadmin` command to generate and/or update the redirect database.
2. User connects to the redirect server and attempts to authenticate.
3. Redirect server determines the identity of the user and looks up the corresponding user ID in the redirect database.
4. If the redirect server does not find the user ID in the redirect database, the redirect server contacts the next redirect server (determined by a round-robin mechanism) to locate the redirect database that contains the user ID. If the user ID is found in the redirect database, the redirect server obtains the partition ID to which the user has been assigned.
5. Redirect server determines the node to which the user will be redirected based on the assigned partition ID.
6. Redirect server returns an error to the client that contains the node to which it is being redirected and closes the connection to the client. The redirect server uses the see-other-host stream error to return this information to the client. See [RFC 3920](#) for more information.
7. The client interprets the error and establishes a connection to the node returned with the error.
8. Redirect server continuously monitors nodes and updates its partition-to-host table as required.

About the Instant Messaging Redirect Database

The database includes only local users. Gateways, components, and remote users are not included in the redirect database.

Instant Messaging Redirect Server Overview

The redirect server is an instance of the Instant Messaging server whose sole function is to redirect client connections. The redirect server does not perform any other service to end users. Upon startup, the redirect server loads the server configuration and partitions file and creates the following data structures:

- A list of instances to which this server can redirect client connections. This is the redirect server's instance list. The instance list is built from entries in the `redirect.hosts` file.
- A table that maps partitions to physical hosts. This table is called the partition map. The redirect server builds the partition map by going through the instance list until it reaches the specified maximum number of partitions.

The redirect server uses both data structures to redirect client connections. See [Example 7-1](#) for an explanation of how the redirect server uses this information.

Instant Messaging Redirect Server and StartTLS

As much of the StartTLS negotiation as is required to establish the identity of the connecting client may take place between the client and the redirect server. The client does not need to verify credentials, instead it only requires the user ID.

Configuring an Instant Messaging Server Instance as a Redirect Server

To specify that a server instance is a redirect server, you need to provide a value for the `iim_server.redirect.provider` parameter in `iim.conf`. Once you have designated the instance as a redirect server, you will need to provide further configuration information by specifying values for additional redirect-specific parameters in `iim.conf`. [Table 7-1](#) describes the redirect configuration parameters.

Table 7-1 Redirect Server Configuration Parameters in `iim.conf`

Parameter	Default Value	Description
<code>iimserver.redirect.provider</code>	None	Comma-separated list of redirect provider names or classes that implement the <code>com.sun.im.provider.Redirector</code> interface. Any value for this parameter defines the server instance as a redirect server. Supported values include <code>db</code> , <code>roundrobin</code> , <code>regex</code> , and class names that implement the <code>com.sun.im.provider.Redirector</code> interface.
<code>iimserver.redirect.to</code>	None	Comma-separated list of nodes to which this redirect server may redirect client connections. Node names can be any alphanumeric string. This list may be a superset of the hosts defined in <code>iimserver.redirect.to.nodename.host</code> .
<code>iimserver.redirect.to.nodename.host</code>	None	Where <code>nodename</code> is the name of the node as it exists in <code>iimserver.redirect.to</code> . This attribute is required for <code>nodename</code> to be used by the redirect server.
<code>iimserver.redirect.to.nodename.usessl</code>	False	If true, then <code>nodename</code> is configured to use legacy SSL. See Overview of Using TLS and Legacy SSL in Instant Messaging for more information.
<code>iimserver.redirect.db.users</code>	<code>im-db-base/redirect.db</code>	Name and location of the redirect database.
<code>iimserver.redirect.db.partitions</code>	<code>im-cfg-base/redirect.partitions</code>	Name and location of the redirect partitions file.
<code>iimserver.redirect.db.partitionsize</code>	5000	The maximum number of users in a partition.
<code>iimserver.redirect.roundrobin.partitions</code>	<code>im-cfg-base/redirect.partitions</code>	Name and location of the redirect partitions file.
<code>iimserver.redirect.pollfrequency</code>		The interval between connections made by the redirect server to the hosts defined in the <code>redirect.hosts</code> file. The redirect server polls these hosts to determine if they are online and able to accept client connections.

To Configure an Instant Messaging Server as a Redirect Server

You cannot use versions of Instant Messenger older than 2006Q1 with the redirect server. If you use a third party client, ensure that the client supports XMPP redirection.

1. Gather the information listed in [Table 7-1](#).
2. Open `iim.conf`.
See [Instant Messaging Configuration Parameters in iim.conf](#) for instructions on locating and modifying this file.
3. Modify the parameter values to match your deployment.
[Table 7-1](#) lists the parameters for which you need to provide values. If the parameters do not exist in `iim.conf`, add them. The following example shows the section of `iim.conf` on `iim.siroe.com` corresponding to the redirect server parameters you need to modify.

```
iim_server.redirect.provider=db,roundrobin
iim_server.redirect.to=imserverA,imserverB
iim_server.redirect.to.imserverA.host=iimA.siroe.com
iim_server.redirect.to.imserverB.host=iimB.siroe.com
iim_server.redirect.to.imserverA.usessl=false
iim_server.redirect.to.imserverB.usessl=false
```

4. Save your changes and close `iim.conf`.
5. Refresh the configuration on the redirect server.

```
imadmin refresh server
```

6. Configure clients to connect to the redirect server instead of the multiplexor.

Administering the Instant Messaging Redirect Server

Information about administering the Instant Messaging redirect server is described in the following sections:

- [Stopping, Starting, Restarting, Refreshing, and Checking the Status of the Instant Messaging Redirect Server](#)
- [Instant Messaging Redirect Server Logging](#)
- [Setting the Partition Size for the Instant Messaging Redirect Server](#)
- [Specifying the List of Partitions for the Instant Messaging Redirect Server](#)

Stopping, Starting, Restarting, Refreshing, and Checking the Status of the Instant Messaging Redirect Server

The redirect server is an Instant Messaging server instance that has been configured only to redirect. Use the same procedures for stopping, starting, restarting, refreshing, and checking status that you use for a normal server instance. For example, to start the redirect server, you would type:

```
imadmin start server
```

See [Stopping, Starting, Refreshing, and Checking Instant Messaging Components](#) for more information.

Instant Messaging Redirect Server Logging

The redirect server is an Instant Messaging server instance that has been configured only to redirect. Use the same instructions and logs that you use for a normal server instance. See [Managing Logging for Instant Messaging](#) for more information.

Setting the Partition Size for the Instant Messaging Redirect Server

You can specify the maximum partition size by setting the `iim_server.redirect.db.partitionsizesize` parameter in `iim.conf`. The value for this parameter is equal to the number of users allowed per partition. The default is 5000 (users).

Specifying the List of Partitions for the Instant Messaging Redirect Server

The `redirect.partitions` file defines the primary node to which users in a particular partition will be redirected and a series of fallback nodes if desired. Each non-empty, non-commented line in the file defines the node list for a partition. Each node in the list must correspond to a node defined as a value for the `iim_server.redirect.to` parameter in `iim.conf`. If there are more partitions defined than there are lines in the `redirect.partitions` file, the unspecified partitions are handled by round-robin.

By default, the `redirect.partitions` file is stored in the following location:
<im-cfg-base>/`redirect.partitions`

Example 7-2 Redirect.partitions File Configuration

This `redirect.partitions` file example assumes the following:

- The redirect server has been configured for `db` and `roundrobin` lookups.
- Three nodes have been identified as destinations for redirected clients:
 - `imserverA`
 - `imserverB`
 - `imserverC`
- These three nodes correspond to the following hosts:
 - `iimA.siroe.com`
 - `iimB.siroe.com`
 - `iimC.siroe.com`

This is expressed in `iim.conf` as follows:

```
iim_server.redirect.provider=db,roundrobin
iim_server.redirect.to=imserverA,imserverB, imserverC
iim_server.redirect.to.imserverA.host=iimA.siroe.com
iim_server.redirect.to.imserverB.host=iimB.siroe.com
iim_server.redirect.to.imserverC.host=iimC.siroe.com
```

- There are at least two user partitions.

In this scenario, `redirect.partitions` might look as follows:

```
imserverA, imserverB, imserverC
imserverB, imserverC
```

That there are two non-empty, non-commented lines indicates that there are at least two user partitions. The first line defines the redirect behavior for partition 1. The redirect server will redirect partition 1 users first to `imserverA`. If that fails, the redirect server tries `imserverB` then `imserverC`. If no nodes are operational, the redirect server returns an error to the client.

Creating and Managing the Instant Messaging Redirect Table Using the `rdadmin` Utility

Typically, you use the `rdadmin` utility on an as-needed basis. You do not need to regenerate the table frequently as roster changes are not generally high-volume. However, you should run the utility at least once every two weeks.

To Create a New or Update an Existing Instant Messaging Redirect Database

1. Stop the redirect server:

```
imadmin stop redirect
```

2. If you are updating an existing redirect database, obtain the number of partitions previously created by `rdadmin`:
 - a. Open `rdadmin.log` in a text editor.
The `rdadmin.log` file is stored in:
`<im-runtime-base>/log`
 - b. Find the value for “NO OF PARTITIONS RUN”.
3. Ensure you have at least as many user entries as partitions.
4. Generate the new redirect database:
For example:
`rdadmin generate`
See the `rdadmin` man page for additional `rdadmin` options.
The `rdadmin` utility creates the new database and saves it as `im-db-base/redirect.new.db` unless you specify a different name.
5. If you are generating the redirect database for the first time, rename the database as `redirect.db`.
6. If you are updating an existing redirect database, replace the old redirect database with the new one:
For example:

```
rm im-db-base/redirect.db cp im-db-base/redirect.new.db  
im-db-base/redirect.db
```

7. Start the redirect server:

```
imadmin start redirect
```

Instant Messaging Redirect Server Physical Host Monitoring

The redirect server monitors the operational status of the hosts to which it redirects clients. If the redirect server determines that one of the hosts has failed, it reallocates partitions to subsequent hosts as defined in the `redirect.partitions` file. In addition, the redirect server detects when a host comes back online so that partitions can be redirected back to the host. The redirect server monitors hosts in two ways:

- Periodic polling. The redirect server establishes a connection and opens an XMPP stream at an interval specified by the `iim_server.redirect.pollfrequency` parameter in `iim.conf`.
- Client retry monitoring. The redirect server may determine that a host is nonoperational if it detects that a single client is repeatedly connecting in a short period of time.

Setting the Instant Messaging Redirect Server Host Polling Frequency

1. On the redirect server, open `iim.conf`.
See [Instant Messaging Configuration Parameters in `iim.conf`](#) for instructions on locating and modifying this file.
2. Set the `iim_server.redirect.pollfrequency` parameter.
The value is in minutes. For example:

```
iim_server.redirect.pollfrequency=200
```

3. Save and close `iim.conf`.
4. Refresh the redirect server.

```
imadmin refresh server
```

Instant Messaging Redirect Server Best Practices and Troubleshooting

Best practices for using the Instant Messaging redirect server as well as troubleshooting information are described in the following sections:

- [Redirect Server Certificates](#)
- [Instant Messaging Redirect Server Supported Clients](#)
- [Using Redirect Server and Storing User Properties in LDAP](#)
- [Determining the Partition Size for the Redirect Database](#)
- [Using a Redirect Server as a Partition Host](#)

Redirect Server Certificates

In a deployment that uses certificates for secure authentication, clients may be prompted to accept two certificates every time they connect; one for the redirect server and one for the host to which the client is redirected. To avoid this, use a trusted certificate or the same certificate on both servers.

Instant Messaging Redirect Server Supported Clients

Redirect will not work for clients that do not support RFC 3920 and the `see-other-hosts` stream error (XMPP redirect) in particular. You can use Instant Messenger 2006Q1 or later with the redirect server. If you use a third party client, ensure that the client that supports XMPP redirection.

Using Redirect Server and Storing User Properties in LDAP

If you are using LDAP to store user properties, that is the `iim.userprops.store=ldap`, you need to ensure that the values for `iim_ldap.usergroupbinddn` and `iim_ldap.usergroupbindcred` have Directory Manager level access to the directory.

Determining the Partition Size for the Redirect Database

The partition size should be as large as possible to avoid having to split user networks wherever possible. However, partitions should also not be larger than that which the smallest system can support.

Using a Redirect Server as a Partition Host

It is possible for a redirect server to also host one or more partitions. You do this by listing the redirect server instance in the `redirect.partitions` file or as a value for the `iim_server.redirect.to` parameter. However, you should not make more than one redirect server a partition host because unsynchronized `redirect.partitions` files may cause redirection loops.

Chapter 10. Scaling an Instant Messaging Deployment By Using Server Pooling

Scaling an Oracle Communications Instant Messaging Server Deployment By Using Server Pooling

Server pooling enables you to support millions of users within a single domain. Using a server pool, you can share a domain across several servers in a server pool. In addition, you can use a load balancer such as the redirect server to help manage server utilization in the pool. For information on the load balancing and the redirect server, see [Optimizing an Instant Messaging Server Pool By Using the Redirect Server](#).

This information assumes that you have already installed Instant Messaging on the hosts in your server pool. In addition, if you need AM SSO and Policy management support in a server pool deployment you need to install the Access Manager SDK on each node in the server pool, and configure the SDK to communicate with a single remote Access Manager server.

Topics:

- [Overview of Server Pooling for Instant Messaging](#)
- [Availability in an Instant Messaging Server Pool](#)
- [Configuring Server-to-Server Communication Between Instant Messaging Servers in a Server Pool](#)
- [Adding a New Node to an Existing Instant Messaging Deployment](#)
- [Securing a Multi-node Deployment](#)

Overview of Server Pooling for Instant Messaging

By creating a server pool, the number of users you can support in an Instant Messaging deployment is no longer constrained by the capacity of a single server system. Instead, you can use the resources of several systems to support the users in a single domain. In addition, 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. Deploying more than one server in a server pool creates a multi-node deployment.

You create a server pool by configuring the Instant Messaging servers to communicate over the server-to-server port and get user data from the same LDAP directory. Once you have configured the servers, you need to configure the client resources to point to the load balancer, or load director, instead of a single node's host and port.



Caution

While it is possible to use a shared file system instead of an LDAP directory to store user properties, doing so negatively impacts performance and manageability. For this reason, only LDAP storage is supported for server pools.

To ensure that all servers within a server pool have consistent data, the following information is replicated among all servers in the pool:

- Routing information for end users
- Conference membership and configuration

- Multi-party conference messages

The following information is not replicated:

- One on one chat messages
- Presence subscriptions and notifications

If you are enforcing policy through access control files in your deployment, the content of the access control files must be the same among all servers in a server pool. See [Managing Instant Messaging and Presence Policies](#) for more information.

Availability in an Instant Messaging Server Pool

If a node in a server pool goes down, all currently connected clients are disconnected and the sessions and resources become unavailable. 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. When they do so, they will not need to recreate conferences or news channels as this information is shared between servers in the pool. In addition, one-to-one chat sessions can be continued after the user is directed to another node in the pool.

Configuring Server-to-Server Communication Between Instant Messaging Servers in a Server Pool

This section describes how to enable communication between two Instant Messaging servers, or peers, in a server pool. You must configure all servers in the pool with information about all other servers in the pool.

The following table lists the parameters in `iim.conf` and their values used to set up communication for two example Instant Messaging servers in a server pool; `iimA.siroe.com` and `iimB.siroe.com`.

For more information on the configuration parameters, see [Instant Messaging Configuration Parameters in iim.conf](#).

Example Configuration Information for Two Instant Messaging Servers in a Server Pool

Parameter in <code>iim.conf</code>	Value for Server A	Value for Server B	Notes
<code>iim_server.serverid</code>	<code>iimA.siroe.com</code>	<code>iimB.siroe.com</code>	In a server pool, this ID is used to support the dialback mechanism and is not used for authentication. This value should be unique within the server pool.
<code>iim_server.password</code>	<code>secretforiimA</code>	<code>secret4iimB</code>	
<code>iim_server.coservers</code>	<code>coserver1</code>	<code>coserver1</code>	Each Instant Messaging server is identified by its symbolic name. The symbolic name of the server is added in the <code>iim_server.coservers</code> parameter in <code>iim.conf</code> . This parameter may contain multiple, comma-separated values.
<code>iim_server.domainname</code>	<code>siroe.com</code>	<code>siroe.com</code>	Peer servers within a server pool share the same default domain.
<code>iim_server.coserver1.host</code>	<code>iimB.siroe.com:5269</code>	<code>iimA.siroe.com:5269</code>	The hostname and port number of the peer server in the server pool.
<code>iim_server.coserver1.serverid</code>	<code>iimB.siroe.com</code>	<code>iimA.siroe.com</code>	The server ID (<code>iim_server.serverid</code>) of the peer server in the server pool.
<code>iim_server.coserver1.password</code>	<code>secret4iimB</code>	<code>secretforiimA</code>	The password (<code>iim_server.password</code>) of the peer server in the server pool.
<code>iim_server.coserver1.domain</code>	<code>siroe.com</code>	<code>siroe.com</code>	Peer servers within a server pool share the same default domain.



Note:

When open federation is enabled, do not use the host name as the server ID. For example, the parameter `iim_server.serverid` should not be set to `hostname`.

To Set Up Communication Between Two Instant Messaging Servers in a Server Pool

1. Gather the information listed in [Example Configuration Information for Two Instant Messaging Servers in a Server Pool](#).
2. Change to `im-cfg-base` on the server `iimA.siroe.com`.
See [Instant Messaging Server Directory Structure](#) for instructions on locating `im-cfg-base`.
3. Open `iim.conf`.

See [Instant Messaging Configuration Parameters in iim.conf](#) for instructions on locating and modifying `iim.conf`.

Note

The `iim.conf` file should be owned by the Instant Messaging server account you created during installation. If the `iim.conf` file cannot be read by the Instant Messaging server account, the server and multiplexor will be unable to read the configuration. Additionally, you might lose the ability to edit `iim.conf`.

4. Modify the parameter values to match your deployment. [Table 8-1](#) lists the parameters you need to modify. If the parameters do not exist in `iim.conf`, add them. The following example shows the section of `iim.conf` on `iimA.siroe.com` corresponding to the server-to-server communications that you need to modify:

```
iim_server.serverid=iimA.siroe.com
iim_server.password=secretforiimA
iim_server.domainname=siroe.com
iim_server.coservers=coserver1
iim_server.coserver1.host=iimB.siroe.com:5269
iim_server.coserver1.serverid=iimB.siroe.com
iim_server.coserver1.password=secret4iimB
iim_server.coserver1.domain=siroe.com
```

5. Follow steps 2 through 4 for the `iim.conf` file on server `iimB.siroe.com`. The following example shows the section of `iim.conf` on `iimB.siroe.com` corresponding to the server-to-server communications that you need to modify:

```
iim_server.serverid=iimB.siroe.com
iim_server.password=secret4iimB
iim_server.domainname=siroe.com
iim_server.coservers=coserver1
iim_server.coserver1.host=iimA.siroe.com:5269
iim_server.coserver1.serverid=iimA.siroe.com
iim_server.coserver1.password=secretforiimA
iim_server.coserver1.domain=siroe.com
```

6. Save the changes and close `iim.conf`.
7. Refresh the configuration on both servers.

```
imadmin refresh server
```

Adding a New Node to an Existing Instant Messaging Deployment

If you need to add an additional node to an existing server pool, you need to configure the new server for server-to-server communication and then add configuration information about the new server to all existing servers in the pool. In addition, you need to add configuration information about all the servers in the pool to the new node. See [To Set Up Communication Between Two Instant Messaging Servers in a Server Pool](#) for instructions.

Securing a Multi-node Deployment

When a node connects to a remote server, the node provides a dialback key. The remote server then

connects back to the node in order to verify the dialback key. In a multi-node deployment, the remote server may connect back to a different node in the pool from the node that originally sent the dialback key. The node the remote server connects to must provide the same dialback key that the original connecting node supplied. The *iim_server.dialback.key* configuration parameter defines which dialback key a node should use. The value for the dialback key is randomly generated unless you explicitly specify one. See [To Manually Define the Dialback Key for an Instant Messaging Server in a Server Pool](#) for instructions.

The `From` attribute is used by a remote server to connect back to an initiating server. Typically, a server's domain name is used as the value for the `From` attribute in server-to-server communication under Jabber. However, all servers in a server pool share the same domain name. Therefore, the domain name cannot be used as a key to locate a single server in a pool. Instead, Instant Messaging uses a server or peer identifier (*serverid*) instead of the domain name as the value for the `From` attribute.

To Manually Define the Dialback Key for an Instant Messaging Server in a Server Pool

The value for the dialback key is randomly generated unless you explicitly specify one.

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Modify the value of the *iim_server.dialback.key* parameter.
For example:

```
iim_server.dialback.key=mymultinodedialbackkey
```

3. Save the changes and close `iim.conf`.
4. Refresh the configuration on both servers.

```
imadmin refresh server
```


Chapter 11. Using Shoal for Server Pool Messaging

Using Shoal for Server Pool Messaging

Instant Messaging uses Shoal, a Java technology-based scalable and dynamic clustering framework to connect multiple servers within a server pool. For more information on Shoal, see <https://shoal.dev.java.net/>.

Topics:

- [Using Shoal for Automatic Discovery of Peer Servers in a Pool](#)
- [Using Shoal for Conferences Across Server Pools](#)
- [Using Shoal Across Subnets](#)

The feature Shoal for Automatic Discovery of Peer Servers in a Pool was introduced in **Instant Messaging 8 Update 2**. The features Shoal for Conferences Across Server Pools and Shoal Across Subnets were introduced in **Instant Messaging 8 Update 3**.

Setting Shoal Parameters

To enable Shoal, set the following parameters in the `iim.conf` file:

```
iim_server.serverid = "server1" (Make sure that this value is unique for
each server)
iim_server.password = "secret" (Make sure that this password is same
across all servers)
```

Using Shoal for Automatic Discovery of Peer Servers in a Pool

Instant Messaging enables you to use the Shoal clustering framework to automatically discover and add peer servers in a server pool. The following steps describe how to configure Shoal for the servers in a pool that belong to the same IP subnet. To configure Shoal for servers in a pool that are part of different subnets, see [Using Shoal Across Subnets](#).

To enable auto-discovery of peer servers, perform the following steps:

1. Configure a server pool containing a number of Instant Messaging servers to use the LDAP `propstore` property.
2. Set the following parameter to start auto-discovery:
`iim_server.peer.autodiscover = true`
3. Set the parameters as explained in [Setting Shoal Parameters](#).
Setting the parameters enables you to start and stop the servers as required. If you are connected to one server, you can see the presence of the server and chat with users on any other server.

Using Shoal for Conferences Across Server Pools

Instant Messaging enables the use of Shoal group messaging to broadcast conference messages across the server pool. Shoal framework can be used to send conference messages across the server pool even if you have not used Shoal for auto-discovery or across subnets. When you enable use of Shoal across server pools, all conference presence broadcasts including join and leave notifications, messages, and chat status notifications will be sent using the Shoal group messaging feature.

To enable Shoal for conferences, perform the following steps:

1. Set the parameters as explained in [Setting Shoal Parameters](#).
2. Set the following parameters:

```
iim_server.peer.conferences.usep2p = true/false
```

This parameter is used to enable or disable the use of Shoal for conference messaging. If you set the parameter to false or not set at all, the legacy server-to-server connection is used.

You can enable Shoal anytime during and after configuration. If you enable this feature after configuration, restart all the servers.



Note:

When using Shoal for peer discovery and conferences, ensure the following:

- `iim_server.password` is the same on all hosts.
- Relay is enabled for communication to work when hosts are on different subnets.

Using Shoal Across Subnets

The Shoal configuration of a server pool in a subnet cannot discover new peers that are present in different IP subnets. Shoal uses relay nodes to propagate peer information across subnets. You need to configure the IM server to start a separate process that performs the Shoal relay functionality, by providing connection details of the relays present in different subnets.

To enable Shoal across different subnets, you must start the relay server. To start the relay server, you need at least one relay server per subnet. You can configure any number of relay servers.

To start the relay server, set the following parameters:

```
relay.imadmin.enable=true  
relay.listen_address=<address of relay server> (Optional)
```

The list of relay servers is specified by using the `relay.uri_list` parameter:

```
relay.uri_list = <list of relays>
```

You specify each relay by using a URI of the form `tcp://host:port`. For example:

```
relay.uri_list = tcp://relay2.example.com:5600,  
tcp://relay3.example.com:5600
```

You can start or stop the relay process independently of the IM server. Stopping or restarting the relay

process does not affect the servers that are already in the pool.

Chapter 12. Securing Instant Messaging Using TLS and Legacy SSL

Securing Oracle Communications Instant Messaging Server Using TLS and Legacy SSL

Instant Messaging supports TLS (Transport Layer Security) and legacy SSL (Secure Sockets Layer) for secure communications. This information provides instructions on setting up security for Instant Messaging using these protocols.

Topics:

- [Overview of Using TLS and Legacy SSL in Instant Messaging](#)
- [Setting Up TLS for the Instant Messaging Server](#)
- [Activating TLS on the Instant Messaging Server](#)
- [Setting Up Legacy SSL for the Multiplexor and Instant Messenger](#)
- [Invoking the Secure Version of Instant Messenger](#)

Overview of Using TLS and Legacy SSL in Instant Messaging

Instant Messaging uses a `startTLS` extension to the Transport Layer Security (TLS) 1.0 protocol for client-to-server and server-to-server encrypted communications and for certificate-based authentication between servers. In addition, Instant Messaging supports a legacy implementation of the SSL protocol (version 3.0) for encrypted communications between Instant Messenger and the multiplexor. In the latter case, a certificate is used to validate the identity of the server to which the client connects, but certificates are not used for authentication.

Communication between multiplexor and server is over an unsecured transport. When you use TLS for client-to-server communication, the multiplexor simply passes the bytes from the client to the server and back and does not perform any encryption or decryption.

TLS is fully compatible with SSL and includes all necessary SSL functionality. TLS and SSL function as protocol layers beneath the application layers of XMPP and HTTP.



Caution

If you set up the multiplexor to only use legacy SSL, Instant Messenger will only connect to the multiplexor using SSL and will disregard any information returned from the server about TLS availability. However, if you choose to use legacy SSL with the multiplexor, all XMPP/HTTP Gateway instances should be configured to communicate directly with the server and not the multiplexor. The gateway does not support legacy SSL. Third-party clients that connect to the multiplexor over legacy SSL and then request a TLS connection are permitted to do so.

In addition, the multiplexor connects to the server over an unsecured transport. If you want to secure communications from end-to-end (client through multiplexor to server and back), use TLS instead of legacy SSL.

You must use Java 1.5 (minimum) in order to use TLS with the Instant Messaging server.

For information on TLS and StartTLS in XMPP, see Use of TLS in RFC 3920, [Extensible Messaging and Presence Protocol: Core](#). For an overview of certificates, SSL, and TLS, see [Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#). The procedures in this section assume you are using the Sun Java System Application Server to generate certificates. If you are using another web container, you will need to refer to that web container's documentation for specific instructions on generating keystores and certificates.

Setting Up TLS for the Instant Messaging Server

Enabling TLS for Instant Messaging server-to-server and client-to-server communication requires the following general steps:

1. Creating a Java keystore (JKS) and a private key using the keytool utility.
For an overview of the keytool utility, see ["Tools for Managing Security" in Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#). For instructions on generating the JKS using Sun Java System Application Server, see ["Working with Certificates and SSL" in Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#).
2. Using the private key to generate a server certificate for the Instant Messaging server. See ["Generating a Certificate Using the keytool Utility" in Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#) for instructions.
3. Getting the Instant Messaging server certificate signed by a Certificate Authority (CA). See ["Signing a Digital Certificate Using the keytool Utility" in Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#) for instructions. Replace Application Server with Instant Messaging where applicable.
4. Restart the Instant Messaging server. See [Starting Instant Messaging Components](#) for details.
5. Obtain the CA's root certificate. Contact your CA for instructions on obtaining the CA's root certificate.
6. Import the certificates into the keystore.
You import the CA root certificate and the signed server certificate into the keystore using the keytool utility as described in ["Using the keytool Utility" in Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#).
7. Activating TLS in the server by setting the appropriate parameters in the `im.conf` file. For instructions see [Activating TLS on the Instant Messaging Server](#).
8. For server-to-server communication over TLS, you need to repeat these steps for each server that will be communicating over TLS. You do not need to perform anything to configure Instant Messenger to use TLS. You also do not need to configure the multiplexor for TLS, however you must not set up the multiplexor to use legacy SSL if you intend to use TLS.
9. If you are using the XMPP/HTTP Gateway in your deployment, configure the gateway to communicate directly with the Instant Messaging server and not the multiplexor.

If you are using the Sun Java System Application Server, steps 1 through 5 are documented in ["Working with Certificates and SSL" in Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#). Step 6 is described in [Activating TLS on the Instant Messaging Server](#).

Activating TLS on the Instant Messaging Server

Before you can activate TLS on the server, you must create a JKS, obtain and install a signed server certificate, and trust the CA's certificate as described in [Setting Up TLS for the Instant Messaging Server](#). You activate TLS on the server when you want to use TLS for server-to-server and/or client-to-server communication.

[Table 12-1](#) lists the parameters in `im.conf` used to enable TLS in an Instant Messaging server. It also contains the description and the default value of these parameters.

Table 12-1 Instant Messaging Server TLS Configuration Parameters

Parameter	Default Value	Description
<i>iim_server.sslkeystore</i>	None	Contains the relative path and filename for the server's Java keystore (JKS). For example: <i>im-cfg-base</i> <i>server-keystore.jks</i>
<i>iim_server.keystorepasswordfile</i>	<i>sslpassword.conf</i>	Contains the relative path and the name of the file that contains the password for the keystore. This file should contain the following line: Internal (Software) Token: <i>password</i> Where <i>password</i> is the password protecting the keystore.
<i>iim_server.requiresssl</i>	<i>false</i>	If true, the server will terminate any connection that does not request a TLS connection after the initial stream session is set up.
<i>iim_server.trust_all_cert</i>	<i>false</i>	If this value is <i>true</i> , the server will trust all certificates, including expired and self-signed certificates, and will also add the certificate information into the log files. If <i>false</i> , the server will not log certificate information and will only trust valid certificates signed by a CA.

To Activate TLS Communication in the Instant Messaging Server

Use this procedure to configure the Instant Messaging server to use secure communication over TLS in the following ways:

- Require TLS for all client and server connections.
- Require TLS only for specific server-to-server connections.
- Allow TLS connections for clients and servers that request a secure transport after the initial communication session has been set up.
- A Combination of requiring TLS for specific server-to-server connections and allowing TLS connections for other clients and servers.

Ensure that you have created a JKS, obtained and installed a server certificate, and configured the server to trust the CA's certificate as described in [Setting Up TLS for the Instant Messaging Server](#).

For server-to-server TLS communication, you must complete this procedure on each server you want to configure to use TLS.

1. Add values for the following parameters in *iim.conf*.
If the parameters are not already present in *iim.conf*, add them.

```
iim_server.sslkeystore=/opt/sun/comms/im/config/server-keystore.jks
iim_server.keystorepasswordfile=sslpassword.conf
```

The server will now respond to a connection request from any client or another Instant Messaging server with the information that it is able to communicate over TLS. The requesting client or server then chooses whether or not to establish a secure connection over TLS.

2. If you want the server to require TLS for all connections from clients, and remote and peer servers, add the following parameter to *iim.conf*:

```
iim_server.requiresssl=true
```

If you set this parameter to `true`, the server will terminate a connection with any client or remote or peer server that does not support TLS. Use this parameter to require secure client-server communication over TLS.

See [Federating Deployment of Multiple Instant Messaging Servers](#) for more information about server-to-server communication.

3. If you want to require TLS for communication with a specific remote or peer server, add the following parameter to `iim.conf`:

```
iim_server.coserver1.requiresssl=true
```

Set this parameter for each coserver for which you want to require TLS.

If you set `iim_server.requiresssl` to `true`, the server will require a TLS connection for any server with which it communicates. In this case, you do not need to set this parameter for specific coservers.

4. (Optional) If you want the server to trust all certificates it receives, and to add certificate information to the log files, add the following parameter to `iim.conf`:

```
iim_server.trust_all_cert=true
```



Caution

You might need to use this feature to test your deployment before you go live. However, you typically should not do this on a deployed system as it presents severe security risks. If this value is `true`, the server will trust all certificates, including expired and self-signed certificates, and will also add the certificate information into the log files. If `false`, the server will not log certificate information and will only trust valid certificates signed by a CA.

5. Refresh the server configuration using `imadmin`.

```
imadmin refresh server
```

6. Verify that TLS is working properly.
You can do this a number of ways, for example by following the steps in [Invoking the Secure Version of Instant Messenger](#).

Example 12-1 TLS Configuration in `iim.conf`

The following is an example section of an `iim.conf` file with the required TLS configuration for server-to-server and client-to-server communication. Values for the parameters in this example will be different in your deployment.

```
! Server to server communication port.
iim_server.port = "5269"
! Should the server listen on the server to server
! communication port
iim_server.useport = "True"
iim_server.coservers=coserver1
iim_server.coserver1.serverid=Iamcompany22
iim_server.coserver1.password=secretforcompany22
iim_server.coserver1.host=iim.i-zed.com:5269
iim_server.serverid=Iami-zed
iim_server.password=secret4i-zed
iim_server.trust_all_cert=true
iim_server.sslkeystore=/opt/sun/comms/im/config/server-keystore.jks
iim_server.keystorepasswordfile=sslpassword.conf
```

Setting Up Legacy SSL for the Multiplexor and Instant Messenger

If you are using an Instant Messaging client that does not support TLS, you can still use SSL instead of TLS for client-to-multiplexor communication. If you configure the multiplexor to use SSL, you cannot use TLS for client-to-server communication. All communication between the multiplexor and the server will be in clear text over an unsecured transport.

If you set up legacy SSL on the multiplexor and are using the XMPP/HTTP Gateway, you must configure the gateway to communicate directly with the server, not the multiplexor. The gateway does not support legacy SSL.

Enabling SSL between the multiplexor and Instant Messenger requires the following:

1. Requesting an SSL Certificate for the Instant Messaging Multiplexor from the CA.
2. Installing the Certificate.
3. Enabling Legacy SSL Between the Multiplexor and Instant Messenger.
4. Activating TLS on the Instant Messaging Server.
5. Invoking the Secure Version of Instant Messenger.



Note:

Instant Messaging 7.3 and later versions don't support NSS-based certificates for multiplexor. SSL support for multiplexor is through JKS. For more information on how to enable SSL through JKS, see [Migrating the Multiplexor Certificate and Enabling SSL](#).

Requesting an SSL Certificate for the Instant Messaging Multiplexor from the CA

To enable SSL in the multiplexor, you need to request a certificate.

To Request a Certificate for the Instant Messaging Multiplexor

This section assumes you are requesting the certificate using either the Sun Java System Web Server or Sun Java System Application Server as your web container.

The multiplexor uses NSS for certificate management, so you can use the NSS utilities to create, manage, and use certificates and the certificate database.

1. In a web browser, type the following URL to start the web container's administration server:


```
http://hostname.domain-name:administration-port
```

A window prompting you for a user name and password appears.

2. Type the administration user name and password you specified during the Web Server or Application Server installation.
The Administration Server page appears.
3. Create a separate Web Server or Application Server instance.
For more information on installing multiple instances of the Application Server, see the [Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#). For information about installing multiple instances of Web Server, see the [Sun Java Communications Suite 5 Installation Guide](#).
4. Create a trust database to store the public and private keys, referred as the key-pair file. The key-pair file is used for SSL encryption. For information on creating a trust database, see [Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#) for Application Server and [Sun Java System Web Server 7.0 Administrator's Guide](#) for Web Server.
5. Request a certificate from the CA.
For more information on requesting a certificate, see [Sun Java System Application Server Enterprise Edition 8.2 Administration Guide](#) for Application Server and [Sun Java System Web Server 7.0 Administrator's Guide](#) for Web Server.

Installing the Certificate

After you receive the signed server certificate from your Certificate Authority, you need to install the certificate and create databases for secure communication.

To Install the Certificate for the Instant Messaging Multiplexor

1. In a web browser, type the following URL to start the administration server:

```
http://hostname.domain-name:administration-port
```

A window appears, prompting you for a user name and password.

2. Type the administration user name and password you specified during the Web Server or Application Server installation.
The Administration Server page appears.
3. Install the server certificate.
For more information on installing the certificate, see the Web Server or Application Server product documentation at <http://www.oracle.com/technetwork/indexes/documentation/index.html>
4. Change to your Web Server or Application Server's `/alias` directory.
5. Copy the database files from the `/alias` directory to the Instant Messaging server's `im-cfg-base` directory.

For example, on Solaris:

```
cp https-serverid-hostname-cert8.db
/etc/opt/SUNWiim/default/config/cert8.db

cp https-serverid-hostname-key3.db
/etc/opt/SUNWiim/default/config/key3.db

cp secmod.db /etc/opt/SUNWiim/default/config/secmod.db
```

and on Red Hat Linux:

```
cp https-serverid-hostname-cert8.db
/etc/opt/sun/im/default/config/cert8.db

cp https-serverid-hostname-key3.db
/etc/opt/sun/im/default/config/key3.db

cp secmod.db /etc/opt/sun/im/default/config/secmod.db
```



Note

You need to allow Read permission on the `cert7.db`, `key3.db`, and `secmod.db` files for the system user used by the multiplexor. In addition, if you created multiple instances of Instant Messaging, the name of the `/default` directory will vary depending on the instance.

See [Table 3-1](#) for default locations for *im-cfg-base*.

6. Change to your *im-cfg-base* on the multiplexor's host.
See [Instant Messaging Server Directory Structure](#) for information on locating *im-cfg-base*.
7. Create a file named `sslpassword.conf` using a text editor of your choice.
8. Enter the following line in `sslpassword.conf`.

```
Internal (Software) Token:password
```

Where *password* is the password you specified when you created the trust database.

9. Save and close `sslpassword.conf`.
10. Ensure that all Instant Messenger end users have Ownership and Read permission on `sslpassword.conf`.
11. Restart the multiplexor.
12. Verify that SSL is working properly.
You can do this a number of ways, for example by following the steps in [Invoking the Secure Version of Instant Messenger](#).
13. Log in to the Web Server or Application Server as an administrator.
14. Remove the server instance that you created while requesting the certificate.

Enabling Legacy SSL Between the Multiplexor and Instant Messenger

You enable SSL for client-to-multiplexor communication by modifying parameters in `iim.conf` and then connecting to the multiplexor using the secure version of the Instant Messenger client.

[Table 12-2](#) lists the parameters in `iim.conf` for enabling SSL between Instant Messenger and the multiplexor. It also lists the description and the default value of these parameters.

Table 12-2 Instant Messaging Multiplexor SSL Parameters

Parameter	Default Value	Description
iim_mux.usessl	off	If the value is set to on, the multiplexor requires an SSL handshake for each connection it accepts, before exchanging any application data.
iim_mux.secconfigdir	Solaris: /etc/opt/SUNWiim/default/config Linux: /etc/opt/sun/im/default/config	This directory contains the key and certificate databases. It usually contains the security module database. In addition, if you created multiple instances of Instant Messaging, the name of the /default directory will vary depending on the instance. See Creating Multiple Instances from a Single Instant Messaging Installation for more information.
iim_mux.keydbprefix	(Empty string)	This value should contain the key database filename prefix. The key database file name must always end with key3.db. If the Key database contains a prefix, for example This-Database-key3.db, then value of this parameter is This-Database.
iim_mux.certdbprefix	(Empty string)	This value should contain the certificate database filename prefix. The certificate database file name must always end with cert7.db. If the certificate database contains a prefix, for example Secret-stuff-cert7.db, then value of this parameter is Secret-stuff.
iim_mux.secmodfile	secmod.db	This value should contain the name of the security module file.
iim_mux.cernickname	Multiplexor-Cert	This value should contain the name of the certificate you entered while installing the certificate. The certificate name is case-sensitive.
iim_mux.keystorepasswordfile	sslpassword.conf	This value should contain the relative path and the name of the file containing the password for the key database. This file should contain the following line: Internal (Software) Token: <i>password</i> Where <i>password</i> is the password protecting the key database.

To Enable SSL Between Instant Messenger and the Multiplexor

1. Open iim.conf.

- See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Add the values from [Table 12-2](#) to the multiplexor configuration parameters in `iim.conf`.

Example 12-2 Legacy SSL Multiplexor Configuration in `iim.conf`

The following is an example of `iim.conf` with the multiplexor configuration parameters included:

```
! IIM multiplexor configuration
! =====
!
! Multiplexor specific options

! IP address and listening port for the multiplexor.
! WARNING: If this value is changed, the port value of
&rsquo;-server&rsquo;
! argument in the client's im.html and im.jnlp files should
! also be changed to match this.
iim_mux.listenport = "siroe.com:5222"

! The IM server and port the multiplexor talks to.
iim_mux.serverport = "siroe.com:45222"

! Number of instances of the multiplexor.
iim_mux.numinstances = "1"

! Maximum number of threads per instance
iim_mux.maxthreads = "10"

! Maximum number of concurrent connections per multiplexor process
iim_mux.maxsessions = "1000"

iim_mux.usessl = "on"
iim_mux.seconfigdir = "/etc/opt/SUNWiim/default/config"
iim_mux.keydbprefix = "This-Database"
iim_mux.certdbprefix = "Secret-stuff"
iim_mux.secmodfile = "secmod.db"
iim_mux.certnickname = "Multiplexor_Cert"
iim_mux.keystorepasswordfile = "sslpassword.conf"
```

Invoking the Secure Version of Instant Messenger

Instant Messenger automatically supports TLS. If you have configured the server to use TLS as described in [Activating TLS on the Instant Messaging Server](#) then the server will communicate to the client that it can support a TLS session when Instant Messenger connects to the server. Instant Messenger can then request that the connection be changed to use TLS.

You invoke the legacy SSL version of Instant Messenger by accessing `imssl.html` or `imssl.jnlp` from your web browser. These files are located under the resource directory, the base directory under which all the Instant Messenger resources are stored.

The links to these applet descriptor files can also be added to `index.html`.

Once you have configured legacy SSL for the multiplexor or TLS for the server, you can verify that the Instant Messenger client has made a secure connection.

To Verify a Secure Instant Messenger Connection

1. Log in to Instant Messenger.
If you are using legacy SSL, access `imssl.html` or `imssl.jsp` from your web browser. If you are using TLS, access the client normally. See [Invoking Instant Messenger](#) for information. Instant Messenger will always use TLS if it is available and if the multiplexor is not set up for legacy SSL.
2. On the Instant Messenger Main Window, ensure the lock icon is visible.
The lock icon appears on the bottom right corner of the Main Window when Instant Messenger is using a secured transport, either SSL or TLS.

Chapter 13. Configuring the Voice Chat

Configuring Voice Chat for Oracle Communications Instant Messaging Server

This chapter gives an overview of the voice chat feature and explains the procedure to configure this feature in Instant Messenger. This chapter contains the following sections:

- [Voice Chat Overview](#)
- [Configuring Voice Chat](#)
- [Initiating Voice Chat](#)

Voice Chat Overview

Instant Messaging provides its users the capability to talk to each other by using a PC-based audio hardware. The Instant Messaging client uses the Instant Messaging server to set up the call. The client uses the `XEP-0166-Jingle` protocol for VoIP in which the audio streams travel in peer-to-peer (p2p) signals. For more information about the `XEP-0166-Jingle` protocol, see <http://www.xmpp.org/extensions/xep-0166.html>.



Note

This feature does not support interactions with other VoIP providers.

Configuring Voice Chat

To enable this feature, select the Enable Audio option while deploying the Instant Messaging Client resources. After you select the Enable Audio option, you can chat and start the voice chat session. This feature enables only peer-to-peer chat. The voice chat option does not exist in a conference chat.



Note

Once you configure the Instant Messenger, you cannot enable or disable the Enable Audio option. To make any changes to this option, reconfigure Instant Messenger by invoking the `init-config` command.

Initiating Voice Chat

The Audio Pane is available at the bottom right of the Lower Message pane in the Instant Messaging client interface. This pane contains icons that can be used to initiate and end voice conversations with other users. The audio pane contains the following items:



Initiate a call: Click this icon to initiate an audio chat with other users.



Hang up: Click this icon to end an audio chat.

To initiate a voice chat, do the following:

1. Start the text-chat session.
2. Select the user from the participation list. The Initiate a call icon is enabled.
3. Click the Initiate a call icon.
The other user gets a request to accept the call. Once the user accepts the call, they are in connected status. The user can use the Hang up icon to end the call.

**Note**

There is no option to record or archive a voice chat conversation.

Firewall Consideration

A common issue with a client-to-client communication is the network and client-level firewalls that block incoming connections to random ports.

The voice chat feature uses an initial TCP client-to-server session to negotiate the client-to-client UDP VoIP link. The current voice chat feature does not have a method to restrict the UDP ports that are used on the clients that accept the connection.

One indication that a firewall might be blocking the establishment of the VoIP link is when the client which initiates the VoIP connection sees "Trying..." for 10 seconds and then a "Call Ended" message is displayed even though the incoming voice chat request was accepted by the other client.

To enable the Instant Messaging Client `debug` option, add the `<argument> debug=true </argument>` parameter in the `im.jnlp` file. The debug information appears in the [Java Console](#) output.

**Note**

If VoIP is enabled, the audio pane is available at the bottom right of the Lower Message pane of the chat window. If VoIP is not enabled during configuration, the audio pane is not available in the chat window.

There is currently no method to enable or disable the VoIP feature at a user or domain level.

Chapter 14. Administering Instant Messaging End Users

Administering Instant Messaging End Users

Instant Messaging does not provide bulk user provisioning tools. You need to use a directory bulk provisioning tool for provisioning multiple Instant Messaging end users. By default, Instant Messaging does not provide specific commands to add, modify, or delete Instant Messaging end users. However, you can customize Instant Messenger to allow users to add themselves to the directory.

Likewise in an LDAP-only deployment, you cannot prevent an end user from using Instant Messenger. In an LDAP-only deployment, the only way to prevent end users from using Instant Messaging is to delete them from the directory or inactivate their user accounts in the directory. Keep in mind that doing this also prevents the user from binding to the directory. In a deployment using Access Manager policy attributes, you can prevent an end user from accessing only Instant Messenger. In addition, if you deploy Instant Messaging with Access Manager, you should use the provisioning tools provided with Access Manager instead of allowing users to register themselves.

The administrator can manage Instant Messaging end users, using the Instant Messaging Administrator Access Control mechanism. For more information on Instant Messaging Administrator Access Control, see [Overview of Privacy, Security, and Site Policies](#), then the Access Manager is used for provisioning Instant Messaging end users. For more information, see the [Communications Suite Deployment Planning Guide](#).



Caution

If you deny end users the privilege to set up watches on other end users by editing the `sysWatch.acl` file, the Instant Messenger's Main window is not displayed for these end users. This effectively denies end users the ability to send instant messages. However, end users would still be able to see alerts and news channels.

Topics:

- [Disabling End User Access to Instant Messenger](#)
- [Registering New Instant Messaging Users](#)
- [Storing Instant Messaging User Properties in LDAP](#)
- [Assigning Instant Messaging and Presence Services to End Users](#)

Disabling End User Access to Instant Messenger

If you are using Instant Messaging with Access Manager, you can deny user access to Instant Messenger services as described in this section.

To Disable Instant Messaging End Users

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Modify the following values as shown:


```
iim_ldap.useidentityadmin="true"
iim_server.usesso=1The value for this parameter may also be 0
iim.policy.modules="identity"
iim.userprops.store="ldap"
```

3. Save and close `iim.conf`.
4. Refresh the Instant Messaging server.

```
imadmin refresh server
```

See [Refreshing Component Configuration](#) for more information. If you are using Instant Messaging in an HA environment, do not use `imadmin`, instead use the Oracle Solaris Cluster tools to refresh the server.

5. Use the Access Manager console (`amconsole`) to remove Instant Messaging services from the user for which you want to disable access.

Registering New Instant Messaging Users

You can customize Instant Messenger to allow new user registration. When a user registers, the Instant Messaging server uses the information provided during registration to perform an `ldapadd` operation to create a user entry in the directory.



Note

If you are using Instant Messaging with Access Manager, you should not allow users to register using this method. Instead, you should use the provisioning tools provided with Access Manager.

To allow new user registration, you need to configure the server to allow registration and then customize Instant Messenger resources by adding an argument to the `im.jnlp.template` and `im.html.template` files, running the `configure` utility, then (if necessary) redeploying the resource files.

This section describes:

- [Configuring the Instant Messaging Server to Allow New User Registration](#)
- [Customizing Instant Messenger to Allow New User Registration](#)
- [Registering as a New Instant Messaging User](#)

See [Managing Instant Messenger](#) for more information about customizing resource files.

Configuring the Instant Messaging Server to Allow New User Registration

In order to configure the Instant Messaging server to allow new user registration you need to add configuration parameters to `iim.conf`. [Table 14-1](#) lists the parameters you need to add and a brief description of each.

Table 14-1 Instant Messaging Server New User Registration Configuration Parameters

Parameter	Description
<code>iim.register.enable</code>	If <code>TRUE</code> , the server allows new Instant Messaging end users to register themselves (add themselves to the directory) using Instant Messenger.
<code>iim_ldap.register.basedn</code>	If self-registration is enabled, the value of this parameter is the DN of the location in the LDAP directory in which person entries are stored. For example: "ou=people,dc=siroe,dc=com"
<code>iim_ldap.register.domain</code>	The domain to which new users will be added. For example, <code>directory.siroe.com</code> .

To Configure the Instant Messaging Server to Allow New User Registration

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Add the configuration parameters and appropriate values as described in [Table 14-1](#).
3. Save and close `iim.conf`.
4. Refresh the server configuration using the `imadmin` command.
`imadmin refresh server`

Customizing Instant Messenger to Allow New User Registration

When you customize the resource files to allow new user registration, a new button appears on the Login dialog box. Users click this button to access the New User Registration dialog box where they can register. When a user registers, their information is added to the LDAP directory.

To Customize Instant Messenger to Allow New User Registration

1. Open the `im.jnlp.template` file in a text editor.
By default this file is stored in `im-svr-base/html`.
2. Search for the line:

```
<application-desc main-class="com.iplanet.im.client.iIM">
```

3. Add the following argument to the end of the section:

```
<argument>register=true</argument>
```

4. Save and close `im.jnlp.template`.
5. Open the `im.html.template` file in a text editor.
By default this file is stored in `im-svr-base/html`.
6. Add the register parameter to the file:

```
<PARAM NAME="register" VALUE="true">
```

7. Add the following parameter to the `EMBED` tag:

```
register=true
```

8. Save and close `im.html.template`.
9. Run the configure utility, selecting the "Messenger Resources" component only when prompted for which components you want to configure.
See [Configuring Instant Messaging After Installing or Upgrading](#) for instructions.

10. If you are using Application Server or Web Server, redeploy the resource files.
See [Redeploying Resource Files](#) for instructions.
11. Launch Instant Messenger.
The I am a New User button should appear on the Login dialog box.

Registering as a New Instant Messaging User

Once you have added the new user registration argument to the `im.jnlp` and `im.html` files and redeployed the resource files users can register themselves.

To Register as a New Instant Messaging User

1. In a web browser, go to the Instant Messaging home page.
2. Click Start or click Use Java Plug-in.
The Login dialog box appears, displaying the I am a New User button.
3. Click I am a New User.
The New User Registration dialog box appears.
4. Enter the information in the fields provided and click OK.
The information is stored in the directory.

Storing Instant Messaging User Properties in LDAP

In a deployment without Access Manager, you can choose to store user properties in LDAP instead of a file (default). You need to run the `imadmin assign_services` command in order to add required objectclasses to user entries in the directory. These objectclasses are used by Instant Messaging to store user properties in user entries.



Caution

Some user attributes may contain confidential information. Ensure that your directory access control is set up to prevent unauthorized access by non-privileged users. Refer to your directory documentation for more information.

To Store Instant Messaging User Properties in LDAP

1. In `iim.conf`, ensure that the `iim.policy.modules` parameter has a value of `iim_ldap`.
See [iim.conf File Syntax](#) for information on `iim.conf`.
2. In `iim.conf`, ensure that the `iim.userprops.store` parameter has a value of `ldap`.
3. From the command line, run `imadmin` with the `assign_services` option:

```
imadmin assign_services
```

`imadmin` checks the value of the `iim.policy.modules` parameter in `iim.conf`.

4. Enter the Bind DN and password you want `imadmin` use to bind to the directory.
The Bind DN should have sufficient credentials to modify the directory schema, for example the Directory Manager DN.
5. Enter the Base DN under which user entries are stored.
Next, `imadmin` adds `sunIMUser`, and `sunPresenceUser` objectclasses to the user entries in the organization you specified.

Assigning Instant Messaging and Presence Services to End Users

In a deployment with Access Manager, you can assign Instant Messaging and presence services to end users with the `imadmin assign_services` command. Alternatively, you can use the Access Manager console.

To Assign Instant Messaging and Presence Services to End Users

1. In `iim.conf`, ensure that the `iim.policy.modules` parameter has a value of `identity`.
See [iim.conf File Syntax](#) for information on `iim.conf`.
2. From the command line, run `imadmin` with the `assign_services` option:

```
imadmin assign_services
```

`imadmin` checks the value of the `iim.policy.modules` parameter in `iim.conf`.

3. Enter the Base DN of the organization under which user entries are stored.
This is the organization that contains the user entries managed by Access Manager.
Next, `imadmin` assigns Instant Messaging and presence services to the users in the organization you specify.

Chapter 15. Configuring Instant Messaging and Presence Service Protocol

Configuring Oracle Communications Instant Messaging Server and Presence Service Protocol

Instant Messaging and Presence Service (IMPS) protocol enables mobile instant messaging and presence services. IMPS is used to exchange messages and presence information between mobile devices, mobile services and Internet-based instant messaging services. IMPS access is provided through a web-application in the Instant Messaging server. The IMPS web-application can be deployed to a supported web container using the Instant Messaging configuration utility.

This section describes the steps to configure the Instant Messaging and Presence Service protocol. Configuring IMPS protocol includes server-side and client-side configuration.

This section consists of the following topics:

- [Server-Side Configuration](#)
- [Client-Side Configuration](#)

Server-Side Configuration

To configure Instant Messaging on the server side, perform the following steps:

1. Start the Instant Messaging configurator tool.
2. Choose to deploy IMPS and specify the web container path to deploy IMPS.
3. Go to the `im-cfg` folder and specify the values for the following parameters in the `imps.conf` file:
 - `xmppServerHost`: Name of the XMPP server host where the IMPS gateway connects.
 - `xmppPort`: XMPP port to which the IMPS gateway connects.
 - `IMPSSessionAliveTime`: Default time until which the server keeps the clients session alive if no further communication is sent from the client. If the time period and `KeepAliveTime` request value is less than the value specified for this parameter, the client is disconnected.
 - `xmppDefaultDomain`: Default domain of the users.
 - `cirPort`: CIR port information for IMPS clients to connect to.
 - `log4jInitfile`: Log4j file information.
4. In the `im-cfg` folder, edit the `imps-log4j.properties` file and specify the directory where you want to save the log file. For example, `log4j.appender.A1File = /var/opt/sun/comms/im/default/log/imps.log`.
5. To use the LDAP-based realm, specify the following parameters in the `iim.conf` file:

```
iim.policy.modules="iim_ldap"  
iim.userprops.store="ldap"  
iim_ldap.sasl.mechanism.factories="com.ipplanet.im.server.sasl.IMPSSASL"
```



Note

For IMPS 4way login, ensure that the user password is in clear text.

6. To use the identity-based realm, specify the following parameters in the `iim.conf` file:

```
iim.policy.modules="identity"  
iim.userprops.store="ldap"  
iim_ldap.sasl.mechanism.factories="com.ipplanet.im.server.sasl.IMPSSAS"
```

7. Restart the web container and the Instant Messaging server.

Client-Side Configuration

To configure Instant Messaging on the client side, perform the following steps:

1. In any device that supports IMPS, select Instant Messaging.
2. Go to Settings->Servers->New Server and type the server name that you want to use in the Server Name field.
3. Type the required access point that you want to use.
4. Type the IMPS URL in the Web Address field.
5. Type the user ID in the userid field.
6. Provide the user password in the password field.
7. Connect to the server.

Chapter 16. Configuring LDAP Failover for Instant Messaging

Configuring LDAP Failover for Oracle Communications Instant Messaging Server

LDAP failover in Instant Messaging enables you to configure the Instant Messaging server to have multiple LDAP servers as back-end storage. If one LDAP server becomes unavailable, the Instant Messaging server will be able to fail over to another LDAP server.

LDAP failover works on a multi-master replication (MMR) setup of LDAP servers. All the LDAP servers in the settings are masters and have permission to read and write data. The Instant Messaging server uses only one server at a time but fails over to another LDAP server when the current server becomes unavailable. The other LDAP server is expected to be in sync with the current server as far as data is concerned.

Setting Up LDAP Failover

To set up an LDAP failover, perform the following settings:

1. Set up the MMR with the LDAP Servers. All the LDAP servers should be master servers. That is, each server should have the permission to read and write data to all the LDAP servers.
2. Make sure that all the master servers in the setup are started and synchronized.
3. Configure the `iim.conf` file settings as follows:
 - a. Add the LDAP replicas.

```
iim_ldap.replicas=ldap1,ldap2
```

- b. Add the LDAP server name and port.

```
iim_ldap.ldap1.host=<xyz>:<389>
iim_ldap.ldap2.host=<abc>:<489>
{code:none}
where _xyz_ and _abc_ are the host names of the failover
systems, and {{389}} and {{489}} are the ports.
## Set the {{iim_ldap.debugPool}} parameter to {{true}}.
{code:none}
iim_ldap.debugPool=true
```



Note 1

Only the Instant Messaging server is replica aware. All the support tools that use the `iim.conf` file are not replica aware. For support tools to start, the default LDAP server should be up and running.

**Note 2**

The default LDAP configuration in Instant Messaging `iim.conf` file is required, even when LDAP failover is configured (CR 6994439). Example of default LDAP configuration:

```
iim_ldap.host = "<xyz>:389"  
iim_ldap.usergroupbinddn = "cn=Directory Manager"  
iim_ldap.usergroupbindcred=password
```


Chapter 17. Configuring Java Message Service Support for Calendar Server Alerts

Configuring Java Message Service Support for Calendar Server Alerts

This information describes how to configure Java Message Service (JMS) API support for Oracle Communications Calendar Server (formerly Sun Java System Calendar Server) alerts. Instant Messaging version 7.3 uses Java Message Queue as the calendar agent for calendar reminders.

Topics:

- [Configuring Instant Messaging Server](#)
- [Configuring the Calendar Server](#)
- [Configuring Instant Messenger](#)
- [Java Message Queue Commands](#)
- [Calendar Agent Information](#)

Configuring Instant Messaging Server


To configure the JMS API support for Calendar Server alerts, edit one or more of the parameters in the `iim.conf` file as shown in the following table. If these parameters do not already exist in the `iim.conf` file, add them. By default, the `iim.conf` file is present in the `/opt/sun/comms/im/config` directory.

Parameter	Value (including the quotation mark)	Description
JMS Consumers section		
<code>jms.consumers</code>	<code>"cal_reminder2"</code>	Name of the alarm. By default, <code>jms.consumers</code> is commented out in the <code>iim.conf</code> file. Make sure to uncomment this line.
<code>jms.consumer.cal_reminder2.provider</code>	<code>"jmq"</code>	Name of the provider. Java Message Queue should be mentioned as a provider. The string <code>jmq</code> is used in the agent code to instantiate the Java Message Queue specific classes.
<code>jms.consumer.cal_reminder2.type</code>	<code>"topic"</code>	The type of the alarm to set.
<code>jms.consumer.cal_reminder2.factory</code>	<code>"com.iplanet.im.server.JMSCalendarMessageListener"</code>	The name of the C++ factory.

<code>jms.consumer.cal_reminder2.destination</code>	<code>"testTopic"</code>	Destination of the alarm. The destination type is <code>topic</code> . The <code>topic</code> can be administratively created or the Java Message Queue provider can be configured to create a <code>topic</code> for publishing a message to it. When the Calendar agent starts, it tries to subscribe to the configured <code>topic</code> . If the <code>topic</code> is not already present in the Java Message Queue broker, the Calendar agent fails to subscribe to the <code>topic</code> . Therefore, it is necessary to start the Java Message Queue broker and create the <code>topic</code> before starting the Calendar agent. For the commands to start the Java Message Queue broker and create a <code>topic</code> , see Java Message Queue Commands .
JMS Providers section		
<code>jms.providers</code>	<code>"jmq"</code>	The name of the provider.
<code>jms.provider.jmq.broker</code>	<code>"yourJMQserver:port"</code>	Port number that the Java Message Queue server listens to. <code>jms.provider.jmq.broker</code> should be the fully qualified host name or IP address and port that your Java Message Queue server is listening on. For example, <code>localhost:7676</code> or <code>jmqhost.beta.comms.com</code>
<code>jms.provider.jmq.factory</code>	<code>"com.sun.messaging.TopicConnectionFactory"</code>	Name of the C++ factory.
<code>jms.provider.jmq.jmsuser</code>	<code>"guest"</code>	The user ID of the JMS user. A Java Message Queue user is created in the Java Message Queue provider user database. If access to the Java Message Queue provider or the <code>topic</code> is controlled, specify the username. The username is provided by the Calendar agent while establishing connection with the Java Message Queue provider. If this parameter is not specified, the agent tries to connect anonymously.
<code>jms.provider.jmq.jmsspwd</code>	<code>"passwd"</code>	The password of the JMS user.

The parameters in the following table are available starting with **Instant Messaging 8.0**. If upgrading from a prior version, you must set the parameters manually.

Parameter	Value	Description
<code>agent-calendar.imadmin.enable</code>	<code>"false"</code>	If set to true, you can start the agent-calendar by using the <code>imadmin</code> command.
<code>agent-calendar.iim_server.host</code>		Hostname of the Instant Messaging server with which the agent calendar communicates.
<code>agent-calendar.iim_server.port</code>		Port number of the Instant Messaging server with which the agent calendar communicates.

 **Note**

1. Configure Calendar Server to publish messages to the same topic in the `ics.conf` file for the Calendar Server.
2. GlassFish Server (formerly Application Server) uses Java Message Queue on port 7676 independently of Messaging Server and Calendar Server. A user who performs a single host installation of Calendar Server might see problems with port 7676 being occupied if GlassFish Server is already installed on that host. To avoid this problem, edit the `/etc/imq/imqbrokerd.conf` file in the Oracle Solaris default path and set the `ARGS=-port` parameter to a free port.
3. `imqbrokerd` should be up and running for Calendar server alerts to work when configured with Java Message Queue.

Configuring the Calendar Server

Make sure that you specify the `ics.conf` parameters as shown in the following table. If these parameters do not already exist in the `ics.conf` file, add them. To configure the Calendar server, perform the following steps:

1. Log in as an administrator with configuration privileges.
2. Stop Calendar Server services by using the `stop-cal` command.
3. Change to the `/etc/opt/SUNWics5/cal/config` directory and save a copy of your old `ics.conf` file.
4. Edit the `ics.conf` parameters as shown in the following table.

Parameter	Value (including the quotation mark)	Description
<code>caldb.serveralarms.dispatchtype</code>	<code>"jmq"</code>	Type of server alarm to dispatch.
<code>caldb.serveralarms.jmqTopic</code>	<code>"testTopic"</code>	Type of alarm to set.
<code>caldb.serveralarms.jmqhost</code>	<code>"<hostname>"</code>	Name of the Java Message Queue host. The Java Message Queue host should match with the broker specified in the <code>iim.conf</code> file.
<code>caldb.serveralarms.jmqLib</code>	<code>"/opt/sun/comms/calendar/SUNWics5/cal/lib/libmqcrt.so"</code>	Path to the Java Message Queue library. The path to the library on the Linux operating system is <code>/opt/sun/calendar/lib/libmqcrt.so</code> .
<code>caldb.serveralarms.jmqport</code>	<code>"<portname>"</code>	Port number to which the Java Message Queue server will listen to. The Java Message Queue host and port 7676 should match with the broker specified in the <code>iim.conf</code> file.
<code>caldb.serveralarms.jmqUser</code>	<code>"guest"</code>	Name of the Java Message Queue user to publish alarms. The Java Message Queue user to publish alarms can be same as in the <code>iim.conf</code> file.
<code>caldb.serveralarms.jmqPwd</code>	<code>"passwd"</code>	Password of the Java Message Queue user.

Configuring Instant Messenger

This section lists the changes that you should make to the `ics.conf` and `iim.conf` configuration files to configure Instant Messenger.

`ics.conf` File Configuration Settings

Ensure that you have set the following parameters in the `ics.conf` file.

Parameter	Value (including the quotation mark)	Description
<code>caldb.berkeleydb.ensmsg.advancedtopics</code>	<code>"yes"</code>	Set to <code>"yes"</code> to enable notifications for advanced topics.
<code>caldb.berkeleydb.ensmsg.createevent</code>	<code>"yes"</code>	Set to <code>"yes"</code> to enable specific action for which notification is required.

`iim.conf` File Configuration Settings

Ensure that you have set the following parameters in the `iim.conf` file.

**Note**

These parameters are added to the `iim.conf` file when you run the configurator tool to configure the Calendar agent. To manually enable the Calendar pop-up, ensure that the following parameters exist and are correctly set in the `iim.conf` file.

Parameter	Value (including the quotation mark)	Description
<code>iim_agent.enable</code>	"true"	Enables agents for Instant Messaging. Set the value to <code>iim_agent.enable="true"</code> .
<code>iim_agent.agent-calendar.enable</code>	"true"	Loads a component that enables the Calendar agent. Set the value to <code>iim_agent.agent-calendar.enable="true"</code> .
<code>agent-calendar.jid</code>	<code>calendar.siroe.com</code>	Java ID of the Calendar agent. Set the value to <code>agent-calendar.jid=calimbot.server.domain</code> .
<code>agent-calendar.password</code>		Password you want the Calendar agent to use to connect to the Instant Messaging server. Set the value to <code>agent-calendar.password=password</code> .
<code>iim_server.components</code>	<code>agent-calendar, httpbind</code>	Set the value to <code>iim_server.components=agent-calendar</code> .

Java Message Queue Commands

This section lists some of the Java Message Queue commands and the paths to specify in the Oracle Solaris operating system.

- `imqbrokerd`: Starts the Iplanet Message Queue (iMQ) broker and server.
- `imqcmd shutdown bkr`: Stops the iMQ broker and server.
- `imqcmd create dst -t t -n test`: Creates a topic named `test`.
- `imqcmd destroy dst -t t -n test`: Destroys a topic named `test`.
- `imqusermgr add -u guest2 -p passwd -g admin`: Creates a user named `guest2` with the password `passwd`. The `-g admin` option specifies the group under which a user is created. The values of this option can be `admin`, `user`, and `anonymous`. Each value has different access privileges.

The following are the paths in the Oracle Solaris operating system:

- `/var/imq/instances/imqbroker/log/` is the Java Message Queue log directory path.
- `/var/imq/instances/imqbroker/props/config.properties` contains the iMQ configuration.
- `/var/imq/instances/imqbroker/etc/accesscontrol.properties` specifies the access control.

The default user name is `guest` and the password is `guest`. The default file is `user store` and `/var/imq/instances/imqbroker/etc/passwd` is the password file for the users created.

Calendar Agent Information

Calendar agent does not directly refer to classes from the `jmq.jar` file. If `jmq.jar` is not found in the classpath, an error message is logged in the `agent-calendar.log` file.

Calendar agent continues to support the Event Notification Service (ENS) messages. By default, the ENS option is enabled. To use Java Message Queue instead of ENS, disable the ENS option by commenting or removing the corresponding parameters in the `iim.conf` and `ics.conf` files.

Note
Instant Messenger 7.3 uses Java Message Queue as the calendar agent for calendar reminders instead of ENS.

The following sample `iim.conf` file shows the configuration to use the ENS broker.

```
Calendar-IM integration Configuration
=====
Configuration to use ens broker.
JMS Consumers
jms.consumers = "cal_reminder"
jms.consumer.cal_reminder.destination = "enp:///ics/customalarm"
jms.consumer.cal_reminder.provider = "ens"
jms.consumer.cal_reminder.type = "topic"
jms.consumer.cal_reminder.param = "eventtype=calendar.alarm"
jms.consumer.cal_reminder.factory =
"com.ipplanet.im.server.JMSCalendarMessageListener"

JMS providers
jms.providers = "ens"
jms.provider.ens.broker = "yourENShost.beta.comms.com:57997"
jms.provider.ens.factory = "com.ipplanet.ens.jms.EnsTopicConnFactory"
```

Note
To activate changes you make to the `iim.conf` and `ics.conf` files, restart the Instant Messaging Server and Calendar Server.

Chapter 18. Instant Messaging Server New Features

Instant Messaging Server New Features

Features documented on this page were introduced in the following product release versions:

- Instant Messaging (Version 8)
 - SMS Gateway for Instant Messaging
 - Multiuser Chat Support for IMPS
 - Gateways for AOL and MSN
 - Revised `configure` Command

Chapter 19. SMS Gateway for Instant Messaging

SMS Gateway for Instant Messaging

This section describes the SMS (Short Message Service) gateway feature and explains the procedure to configure the SMS gateway for Instant Messaging version 8. This section contains the following topics:

- [SMS Gateway Overview](#)
- [Configuring the SMS Gateway](#)
- [Starting and Stopping the SMS Gateway](#)

SMS Gateway Overview

The SMS gateway feature enables the Instant Messaging server to deliver chat messages and alerts in the form of SMS to the Instant Messaging users who are offline. This feature provides streamlined instant messaging experience to users by forwarding messages to the users' mobile phones when they are offline. The SMS gateway uses the SMPP (short message peer-to-peer) protocol and XMPP (Extensible Messaging and Presence Protocol) for messaging services.

The following list provides a description of the SMS gateway terms:

- **SMS:** Short Message Service is a wireless messaging service that permits the transmission of a short text message from and to a digital wireless terminal.
- **SMSC:** Short Message Service Center is a network element in the mobile telephone network that delivers SMS messages to mobile devices.
- **SMPP:** Short Message Peer-to-Peer protocol is a telecommunication protocol used for exchanging SMS messages between SMS entities. For example, short message service centers.
- **XMPP:** Extensible Messaging and Presence Protocol is an open Extensible Markup Language (XML) protocol for near-real-time messaging, presence, and request-response services.

Configuring the SMS Gateway

To enable the SMS gateway feature, you must configure the Instant Messaging server and client. This section contains the following topics:

- [Server-Side Configuration](#)
- [Client-Side Settings](#)
- [Enabling and Disabling the Forward Offline Messages To The SMS Address Option](#)

Server-Side Configuration

You can configure the SMS gateway feature by manually adding the SMS gateway parameters in the `im.conf` file or by using the `configure` utility.

To Manually Configure the SMS Gateway

To manually configure the SMS gateway, add the following parameters in the `im.conf` file. By default, the `im.conf` file is stored in the `/etc/opt/SUNWiim/default/config` directory.

The following table shows the SMS gateway parameters that you need to add in the `im.conf` file:

Parameter	Default Value	Description
<code>smsgw.imadmin.enable</code>	<code>false</code>	Enables or disables the SMS gateway. If set to <code>true</code> , you can start the SMS gateway by using the <code>imadmin</code> command.
<code>smsgw.jid</code>	None.	A jabber ID (JID) to bind the SMS gateway to the Instant Messaging server. The value of this parameter should be the same as the value that you define for the <code>smpplibind.jid</code> parameter.
<code>smsgw.password</code>		Password to authenticate the SMS gateway to the Instant Messaging server. The value of this parameter should be the same as the value that you define for the <code>smpplibind.password</code> parameter.
<code>smsgw.iim_server</code>	None.	Hostname and port number of the Instant Messaging server.
<code>smsgw.sms_limit</code>	<code>-1</code>	Number of messages that can be sent per hour. The default value is <code>-1</code> and it indicates that unlimited number of SMS messages that can be sent per hour.
<code>smsgw.sms_queue_capacity</code>	512	Maximum number of messages that can be queued for SMS delivery.
<code>smsgw.im_char_limit</code>	500	Maximum number of characters that you can specify in one message. If the number of characters is greater than the specified value, the message is rejected.
<code>smpplib.smsc_ip_address</code>	None.	IP address or hostname of the SMSC.
<code>smpplib.smsc_port</code>	2775	Port number of the SMSC.
<code>smpplib.bind_id</code>	None.	Identifier used to bind the SMS gateway to the SMSC.
<code>smpplib.bind_password</code>		Password to authenticate the SMS gateway to the SMSC.
<code>smpplib.sender_id</code>	None.	Sender ID of the outgoing SMS.

The following table shows the Instant Messaging server parameters that you need to add in the `iim.conf` file.

Parameter	Default Value	Description
<code>iim_server.components</code>	None.	List of component identifiers that should have <code>smpplibind</code> . For example, <code>httpbind</code> , <code>smpplibind</code> .
<code>iim_agent.smpplibind.enable</code>	<code>false</code>	Enables the Instant Messaging server to identify the SMS gateway.
<code>smpplibind.jid</code>	None.	A jabber ID (JID) for binding the SMS gateway to the Instant Messaging server.
<code>smpplibind.password</code>		Password to authenticate the SMS gateway to the Instant Messaging server.

To Configure the SMS Gateway by using the `configure` Utility

To configure the SMS gateway by using `configure` utility, perform the following steps:

1. Install Instant Messaging version 8 by using the Communication Suite 6 Update 1 installer. For more details about the communication installer, see Sun Java Communications Suite 6 Installation Guide.
2. Invoke the `configure` utility.
`./configure`
3. Perform the following tasks in the configurator tool panel.
 - a. Choose the Enable SMS Gateway option by typing `yes`.
 - b. Choose the Enable Local Component option by typing `yes`.
If you select this option, you can administer the SMS gateway by using the `imadmin` command-line utility. For example, to start the SMS gateway, you can type `./imadmin start sms-gateway`. You can also start the gateway by typing `./imadmin start`.
4. Type the XMPP (Extensible Messaging and Presence Protocol) server hostname.
You can configure Instant Messaging and the SMS gateway on the same host or on different hosts. If you choose to configure the gateway for a remote Instant Messaging server, specify the remote server hostname. The default hostname is the name of the local host.
5. Type the port number.
The default value is the port number that you specify for the XMPP server. For example, if the XMPP server port is `5269`, type `5269`.
6. Type the bind ID of the SMSC at the ESME System Id prompt.
7. Type the SMSC bind password at the ESME System Password prompt.
8. Type the IP address or the FQHN (Fully Qualified Host Name) of the SMSC at the SMSC Host address prompt.
9. Type the SMSC port number at the SMSC port prompt. The default port number is `2775`.
10. Type the Sender ID at the SMS Sender ID prompt.
The sender ID is the ID with which you have registered to the SMSC. The SMSC always send a SMS with the sender ID that you specify here.

Client-Side Settings

The Instant Messaging server searches for the recipient phone number in the following order of precedence:

1. Phone number settings in user v-card of a third-party messaging client
2. LDAP setting in the `mobile` attribute of Directory Server
3. Phone number settings in the Instant Messaging client

If you use a third-party messaging client such as Psi, specify the phone settings in the user v-card. See the third-party messaging client documentation for the procedure about adding phone settings.

If you use Directory Server, add the recipient phone number in the LDAP `mobile` attribute. For more information about the Directory Server, refer to the Directory Server documentation at <http://www.oracle.com/technetwork/documentation/legacy-sun-identity-mgmt-193462.html>.

If you use Instant Messaging, add the recipient phone number in the Instant Messaging client.

To add the phone number, perform the following steps:

1. Go to Tools->Settings->Alerts.
2. Select the Forward Offline Alerts to the SMS Address: check box and type the phone number in the text field.
3. Click on Apply->OK->Close to save your settings.



Note

The Forward Offline Alerts to the SMS Address: option is activated in the Instant Messaging client only if the `iim_agent.smpbind.enable` parameter is set to `true` in the `iim.conf` file and if the SMS gateway is up and running. For more information about alert settings, see the Instant Messaging online help.

Enabling and Disabling the Forward Offline Messages To The SMS Address Option

After you configure the SMS gateway feature in the Instant Messaging server, you can enable or disable the Forward offline messages to the SMS address option through the Instant Messaging client.



Note

You can enable or disable the SMS gateway feature only through the Instant Messaging client.

To enable the Forward offline messages to the SMS address option, perform the following steps:

1. Log in to the Instant Messaging client.
2. Go to Tools-Settings->Alerts and check the Forward offline messages to the SMS address check box.
3. Click on Apply->OK->Close to save your settings.

If a message is sent to user who is offline, the message is forwarded as SMS to the user's mobile number.

To disable the Forward offline messages to the SMS address option, perform the following steps:

1. Log in to the Instant Messaging client.
2. Go to Tools-Settings->Alerts tab and uncheck the Forward offline messages to the SMS address check box.
3. Click on Apply->OK->Close to save your settings.

If a message is sent to user who is offline, the message is not forwarded as SMS to the user's mobile number.

Starting and Stopping the SMS Gateway

You can start and stop the SMS gateway by using the `imadmin` command-line utility. Before starting the SMS gateway, make sure that the Instant Messaging service and the SMSC service are online.

To start the SMS gateway, type the following command:

```
./imadmin start sms-gateway
```

To stop the SMS gateway, type the following command:

```
./imadmin stop sms-gateway
```

To check the status of the SMS gateway, type the following command:

```
./imadmin status sms-gateway
```

Chapter 20. Multiuser Chat Support for IMPS

Multiuser Chat Support for IMPS

This section describes the multiuser chat support for Instant Messaging Presence Service (IMPS) in mobile devices. This section contains the following topics:

- [Overview of Multiuser Chat Support for IMPS](#)
- [Multiuser Chat Feature](#)
- [Enabling the Multiuser Chat Support for IMPS](#)
- [Setting Up IMPS for Virtual Domains](#)
- [Multiuser Chat Limitations](#)

Overview of Multiuser Chat Support for IMPS

The Instant Messaging and Presence Service (IMPS) protocol enables instant messaging and presence service in mobile devices. The IMPS protocol is used to exchange messages and presence information between mobile devices, mobile phone service providers, and Internet-based instant messaging services. The Instant Messaging server provides IMPS access through a web application. The IMPS web application can be deployed to a web container by using the Instant Messaging `configure` utility.

Instant Messaging version 8 provides a multiuser chat feature for mobile devices that support IMPS. To enable the multiuser chat feature, Instant Messaging uses Extensible Messaging and Presence Protocol (XMPP). Multiple XMPP users can exchange messages in the context of a conference room or channel. In addition to the chat room features such as room topics and invitations, the XMPP protocol enables the ability to kick and ban users from a conference, name room moderators and administrators, and provide different types of membership.

Messages received from an IMPS client are converted to XMPP messages and sent to the Instant Messaging server through the multiplexor.

For more information about how to configure the IMPS, see [Configuring Instant Messaging and Presence Service Protocol](#).

Multiuser Chat Feature

The multiuser chat feature is accessible to users on the basis of their roles and access privileges. The user roles are assigned by the Instant Messaging server administrator. The administrator can use either Access Manager or the access control list (ACL) files to define roles. For more information about Access Manager, see <http://www.oracle.com/technetwork/documentation/legacy-sun-identity-mgmt-193462.html>.

IMPS provides the following of access privileges:

- **Admin:** A user with `admin` privileges can create and delete groups, add members to a group, remove members from a group, ban or kick a member from a group, retrieve a list of members in the group, retrieve a list of rejected members, change the member access, set group properties, and retrieve group properties.
- **Moderator:** A user with `moderator` privileges can add members to a group, remove members from a group, reject a member from a group, retrieve a list of joined members, retrieve a list of rejected members, and retrieve group properties.
- **Normal:** A user with `normal` privileges can join and leave a conference, retrieve a list of joined members, and retrieve the properties of the group.



Note

The chat option might differ based on the design and specification of various mobile devices. Refer to the user documentation of mobile devices for more information about chat options.

IMPS provides the following chat features:

- **Create group:** Enables users to create a group. A group is formed by two or more users to exchange information, opinions, and comment about a particular topic. The group ID of the group that is created should be unique.
- **Delete group:** Enables users to delete a group.
- **Join and leave a group:** Enables users to join or leave a group.
- **Search for a group:** Enables users to search for a group.
- **Add group members:** Enables users to add members to a group.
- **Remove group members:** Enables users to remove members from a group.
- **Authorize access request of users:** Enables users to authorize members of a group.
- **Rejecting users from a group:** Enables users to reject or kick members from a group. This action removes a user temporarily from a group. The user can re-enter the room at a later time.
- **Subscribe to group changes:** Enables users to subscribe to a group and learn the changes that are made to group.

Enabling the Multiuser Chat Support for IMPS

To enable the multiuser chat for IMPS feature, select the `Deploy IM IMPS Gateway` option while configuring the Instant Messaging server. If you set the `Deploy IM IMPS Gateway` option to `yes`, the `iim_ldap.sasl.mechanism.factories` and `iim_ldap.userpasswordattr` parameters are added to the `iim.conf` file. By default, the `iim.conf` file is stored in the `/etc/opt/SUNWiim/default/config` directory.

Setting Up IMPS for Virtual Domains

If the Instant Messaging server has multiple domains, you can access the users of other domains.

For example, assume that a server has `abc.com` and `xyz.net` as virtual domains. To set up the IMPS for the virtual domains, perform the following steps:

1. Make a copy of the `/etc/opt/SUNWiim/default/config/imps.conf` file and name the copied file as `/etc/opt/SUNWiim/default/config/imps_xyz.conf`.
2. Edit the `imps_xyz.conf` file.
 - a. Change the value of the `xmppDefaultDomain` parameter to `xyz.com`.
 - b. Change the value of the `cirPort` parameter to any unused or existing port number.
3. Generate the IMPS WAR file by typing the following command:

```
/opt/SUNWiim/sbin/iwadmin generate imps -d /tmp/xyz.war
```
4. Extract the IMPS WAR file and edit the `web.xml` file in `xyz.war`.
5. Change the value of the `imps.config.file` parameter to `/etc/opt/SUNWiim/default/config/imps_xyz.conf`.
6. Regenerate the IMPS WAR file and deploy the WAR in the application server with a different

context path. For example, `/xyz`.

You can now access the virtual domain in the device by using the URL `http://host:port/xyz`.

Multiuser Chat Limitations

The multiuser chat for IMPS feature has the following limitations:

- Only users with `admin` or `moderator` privileges can get group properties in a XMPP server. If a user with `normal` privilege tries to get group properties from a mobile device, an error occurs. For example, *Insufficient user rights*. According to the IMPS specification, every user can get group properties.
- The IMPS four-way login, IMPS4WAY, is not exposed when the realm is `Identity`. The login works only when the realm is `LDAP`.
- Non-default domain users are not supported in the `iim_ldap` policy. The support is available only in a `iim_ldap_schema1` setup.

Chapter 21. Gateways for AIM, MSN, and Yahoo

Support for AIM, MSN, and Yahoo Gateways

As of Instant Messaging 9.0.1.4.0, the AIM, MSN, and Yahoo gateways are deprecated and may be removed in a future release.

This information describes the procedure to enable and configure gateways with the Instant Messaging software.

Topics:

- [Overview of Gateways](#)
- [Enabling the AIM Gateway in Instant Messaging](#)
- [Enabling the MSN Gateway in Instant Messaging](#)
- [Enabling the Yahoo Gateway in Instant Messaging](#)
- [Accessing Gateways](#)
- [Gateway Limitations](#)

Overview of Gateways

Instant Messaging version 8 co-packages the gateways for AIM (AOL Instant Messaging), MSN, and Yahoo messaging clients. The gateways enable Instant Messaging users to communicate with their contacts on AOL, MSN, and Yahoo. Co-packaging the gateways enables easy installation and configuration of the gateways with Instant Messaging. You can configure the gateway by using the `configure` utility. This section describes the steps to enable gateways with Instant Messaging.

System Requirements for Gateways

When you install Instant Messaging version 8 and above through the Communication Suite installer, the external gateways PyAIMt for AIM, PyMSNt for MSN, and PyYIMt for Yahoo are installed with the Instant Messaging software. The installer also installs the gateway dependencies such as Twisted, Zope interface, pyOpenSSL, and pycrypto.

The following table lists the platform support for gateways for different versions of Instant Messaging:

IM Version	Oracle Solaris Version	Red Hat Version
Instant Messaging 8	Oracle Solaris 9 and 10	Red Hat 4 and 5
Instant Messaging 8 Update 1	Oracle Solaris 9 and 10	Red Hat 4 and Red Hat 5
Instant Messaging 8 Update 2	Oracle Solaris 10	Red Hat 4 and Red Hat 5 Update 3
Instant Messaging 8 Update 3	Oracle Solaris 10	Red Hat 5 Update 3

Make sure that the following software exist in the machine where you install the gateways.

Oracle Solaris 9

- Python version 2.4.3

- OpenSSL version 0.9.7a
- GCC (GNU Compiler Collection) libgcc-3.3

Oracle Solaris 10, Red Hat 4 and Red Hat 5 OS

The required versions of Python, OpenSSL, and libgcc come inbuilt with the Oracle Solaris 10, Red Hat 4 and Red Hat 5 operating systems. You do not have to install them separately.

Python 2.4.x comes inbuilt with Oracle Solaris 10, Red Hat4, and Red Hat5 OS. This is the supported and certified python version for gateways on the above platforms.

By default, the gateways pick up the dependencies from their default installation path. If you install the dependencies in non-default locations, specify the path in the `gateways.conf` configuration file. The `gateways.conf` file is present in the `im-base-dir/gateways.conf` directory.

Enabling the AIM Gateway in Instant Messaging

As of Instant Messaging 9.0.1.4.0, the AIM gateway is deprecated and may be removed in a future release.

To enable the AIM gateway in Instant Messaging, perform the following steps:

1. Install Instant Messaging version 8 by using the Communication Suite 6 Update 1 installer. For more details about the communication installer, see Sun Java Communications Suite 6 Installation Guide.
2. Invoke the `configure` utility.


```
./configure
```
3. Perform the following tasks in the configurator tool panel:
 - a. Select the Enable AOL Gateway option by typing `yes` in the command line.
 - b. Select the Enable Local Component option by typing `yes` in the command line.
 If you select this option, you can administer the AIM gateway by using the `imadmin` command-line utility. For example, to start the AIM gateway, you can type `./imadmin start aim-gateway`. You can also start the gateway by typing `./imadmin start`.
4. Type the XMPP (Extensible Messaging and Presence Protocol) server hostname. You can configure Instant Messaging and the AIM gateway on the same host or on different hosts. If you choose to configure the gateway for a remote Instant Messaging server, specify the remote server hostname. The default hostname is the name of the local host.
5. Type the port number. The default value is the port number that you specify for the XMPP server. For example, if the XMPP server port is 5269, type 5269.

Configuring the AIM Gateway in Instant Messaging

When you enable the AIM gateway in Instant Messaging by using the `configure` utility, gateway-related changes are made to the `iim.conf` and `pyaimt.xml` configuration files.

Instant Messaging Configuration Changes

The following parameters are added to the `iim.conf` file. By default, the `iim.conf` file is located in the `im-config-dir` directory.


```

! AOL gateway Integration
!-----
iim_agent.aim_gateway.enable="false"
aim_gateway.jid=aim.${domainname}
aim_gateway.password=<password>
aim_gateway.imadmin.enable="false"

```

AIM Gateway Configuration Changes

The following parameters are added to the `pyaimt.xml` file. By default, the `pyaimt.xml` file is located in the `im-inst-dir/config` directory.

```

<jid> aim.${domainname} </jid>
<confjid>muc.${domainname}</confjid>
<spooldir>${instanceVardir}</spooldir>
<pid>${instanceVardir}/log/PyAIMt.pid</pid>
<mainServer>${imserver_hostname}</mainServer>
<mainServer>${imserver_hostname}</mainServer>
<port>${imserver_port}</port>
<secret><password></secret>
<debugFile>${instanceVardir}/log/pyaimt.log</debugFile>

```

The generated log file for the AIM gateway is `pyaimt.log` and is stored in the `im-instvar-dir/log` directory.

Enabling the MSN Gateway in Instant Messaging

As of Instant Messaging 9.0.1.4.0, the MSN gateway is deprecated and may be removed in a future release.

To enable the MSN gateway in Instant Messaging, perform the following steps:

1. Install Instant Messaging version 8 by using the Communication Suite 6 Update 1 installer. For more details about the communication installer, see Sun Java Communications Suite 6 Installation Guide.
2. Invoke the `configure` utility.


```
./configure
```
3. Perform the following tasks in the configurator tool panel:
 - a. Select the Enable MSN Gateway option by typing `yes` in the command line.
 - b. Select the Enable Local Component option by typing `yes` in the command line.
 If you select this option, you can administer the MSN gateway by using the `imadmin` command-line utility. For example, to start the MSN gateway, you can type `./imadmin start msn-gateway`. You can also start the gateway by typing `./imadmin start`.
4. Type the XMPP server hostname.
 You can configure Instant Messenger and the MSN gateway on the same host or on different hosts. If you choose to configure the gateway for a remote Instant Messaging server, specify the remote server hostname. The default hostname is the name of the local host.
5. Type the port number.
 The default value is the port number that you specify for the XMPP server. For example, if the XMPP server port is 5269, type 5269.

Configuring the MSN Gateway in Instant Messaging

When you enable the MSN gateway in Instant Messaging by using the `configure` utility, gateway-related changes are made to the `iim.conf` and `pysmnt.xml` configuration files.

Instant Messaging Configuration

The following parameters are added to the `iim.conf` file. By default, the `iim.conf` file is located in the `im-config-dir` directory.

```
! MSN gateway Integration
!-----
iim_agent.msn_gateway.enable="false"
msn_gateway.jid=msn.${domainname}
msn_gateway.password=<password>
msn_gateway.imadmin.enable="false"
```

MSN Gateway Configuration

The following parameters are added to the `pysmnt.xml` file. By default, the `pysmnt.xml` file is located in the `im-inst-dir/config` directory.

```
<jid> msn.${domainname}</jid>
<spooldir>${instanceVardir}</spooldir>
<pid>${instanceVardir}/log/PyMSNt.pid</pid>
<mainServer>${imserver_hostname}</mainServer>
<port> ${imserver_port}</port>
<secret><password></secret>
<debugFile>${instanceVardir}/log/pysmnt.log</debugFile>
<host>FQHN</host>
```

The generated log file for the MSN gateway is `pysmnt.log` and is stored in the `im-instvar-dir/log` directory.

Enabling the Yahoo Gateway in Instant Messaging

This feature is introduced in Instant Messaging 8 Update 1.

As of Instant Messaging 9.0.1.4.0, the Yahoo gateway is deprecated and may be removed in a future release.

To enable the Yahoo gateway in Instant Messaging, perform the following steps:

1. Install Instant Messaging 8 Update 1 by using the Communication Suite 6 Update 2 installer. For more details about the communication installer, see Communications Suite 6 Installation Guide.
2. Invoke the `configure` utility.
`./configure`
3. Perform the following tasks in the configurator tool panel:
 - a. Select the Enable Yahoo Gateway option by typing `yes` in the command line.
 - b. Select the Enable Local Component option by typing `yes` in the command line.

If you select this option, you can administer the Yahoo gateway by using the `imadmin` command-line utility. For example, to start the Yahoo gateway, type `./imadmin start yim-gateway`. You can also start the gateway by typing `./imadmin start`.

4. Type the XMPP server hostname.
You can configure Instant Messenger and the Yahoo gateway on the same host or on different hosts. If you choose to configure the gateway for a remote Instant Messaging server, specify the remote server hostname. The default hostname is the name of the local host.
5. Type the port number.
The default value is the port number that you specify for the XMPP server. For example, if the XMPP server port is 5269, type 5269.

Configuring the Yahoo Gateway in Instant Messaging

When you enable the Yahoo gateway in Instant Messaging by invoking the `configure` utility, gateway-related changes are made to the `iim.conf` and `pyyimt.xml` configuration files. You can modify these configuration files if required.

Instant Messaging Configuration

The following parameters are added to the `iim.conf` file. By default, the `iim.conf` file is located in the `im-config-dir` directory.

```
! Yahoo gateway Integration
!-----
iim_agent.yim_gateway.enable="false"
yim_gateway.jid=yim.${domainname}
yim_gateway.password=<password>
yim_gateway.imadmin.enable="false"
```

Yahoo Gateway Configuration

The following parameters are added to the `pyyimt.xml` file. By default, the `pyyimt.xml` file is located in the `im-inst-dir/config` directory.

```
<jid>yim.${domainName}</jid>
<confjid>muc.${domainName}</confjid>
<host>${FQHN}</host>
<spoolFile>${instanceVarDir}/yahouser.dbm</spoolFile>
<pid>${instanceVarDir}/log/PyYIMt.pid</pid>
<mainServer>${FQHN}</mainServer>
<mainServerJID>${FQHN}</mainServerJID>
<port>${S2SPortNumber}</port>
<secret><password></secret>
<debugFile>${instanceVarDir}/log/PyYIMt.log</debugFile>
```

The generated log file for the Yahoo gateway is `pyyimt.log` and is stored in the `im-instvar-dir/log` directory.

Accessing Gateways

To access gateways through the Instant Messaging client, perform the following steps:

1. Start the Instant Messaging client.

2. Go to File -> Add Services.
A list of transports that you have configured such as AIM Transport, MSN Transport, and Yahoo Transport is displayed.
3. Select the required transport from the Add Services list and type the user name and password.
For example, type the AIM user name and password to connect to the AIM transport.

After you log in, your contacts on the external network are added to the existing contacts list in the Instant Messaging client.

Gateway Limitations

This section lists the limitations of gateways.

- The MSN gateway fails to start on the Red Hat Linux 4 64-bit operating system. To resolve this issue, perform the following steps:
 - a. Build and install pyOpenSSL-0.7 on the Red Hat Linux 4 64-bit OS that has the MSN gateway installed.
 - b. Replace the content of the MSN gateway `OpenSSL` directory with the content of the `pyOpenSSL` `OpenSSL` directory that you built in Step a, by typing the following command:

```
#cp -r /pyopenssl/lib/python2.3/site-packages/OpenSSL/  
/opt/sun/comms/im/lib/gateway/lib/python2.4/site-packages/OpenSSL
```

- c. Restart the MSN gateway.
- In MSN gateway, the roster subscription is sent each time the user registers and logs in to a particular service.
 - In AIM gateway, roster subscription does not work with the Instant Messaging client.
 - The AIM gateway log file is not generated. To generate the log file, perform the following steps:
 - a. Go to the `IMinstalledDir/lib/gateway/pyaim-t-0.8a/src/` directory.
 - b. Edit the `config.py` file.

Change

```
debugLevel = 0 # 0->None, 1->Traceback, 2->WARN,ERROR,  
3->INFO,WARN,ERROR  
debugFile = ""
```

to

```
debugLevel = 3 # 0->None, 1->Traceback, 2->WARN,ERROR,  
3->INFO,WARN,ERROR  
debugFile = "${instanceVarDir}/default/log/PyAIMt.log"
```

- Every time a user connects to AIM, MSN, or Yahoo Transport, a dialog box to Accept or Deny the contact in the buddy list appears.
- Chat messages from MSN users are shown as messages from Instant Messaging user in the Instant Messaging client chat window.
- Logs are not getting generated in Yahoo Gateway.
- The Yahoo Gateway Roster list always displays the status of buddies as offline in Yahoo Transport.

Workaround: You can do either of the following:

1. Login to Instant Messaging client.
2. Add Yahoo Transport and login as a Yahoo user.
3. Log off and then Login to Yahoo Transport service again.

OR

Log off and Login to Yahoo Transport the first time you add Yahoo Transport service.

Chapter 22. Configure Command

configure Utility in Instant Messaging Version 8

This section describes modifications made to the `configure` utility in Instant Messaging version 8.

The `configure` utility enables you to configure the Instant Messaging server as per your deployment specifications. The `configure` utility is revised in the Instant Messaging version 8 release to include new options and remove few options. By default, the `configure` utility is stored in the `<install-directory>/sbin` directory.



Note

Instant Messaging version 8 release does not support Graphical User Interface (GUI) based configuration.

This section has the following topics:

- [Command-Line Syntax](#)
- [Command-Line Options](#)
- [Sample Configuration Using the `configure` Utility](#)

Command-Line Syntax

The `configure` command has the following syntax:

```
./configure --nodisplay --silent --savestate --state --verbose --no  
--novalidate --debug --help
```

Command-Line Options

The `configure` command has the following options:

`--nodisplay`

Required if the `--silent` option is not used. Optional if the `--silent` option is used. Use this option to configure the Instant Messaging server in the command-line mode.

`--help`

Optional. Displays the help content for this command.

`--verbose`

Optional. Prints information messages to the standard output.

`--savestate <statefilename>`

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 <statefilename>`

Required if the `--nodisplay` option is not used. Use this option to run the `configure` command in the

silent mode. Specify the name and path of the state file with this option. If you are configuring the 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 <statefilename>`

Optional. During configuration, the `configure` utility provides default values for configuration. You can either use the default values or specify your own value. If you use this option, the `configure` utility 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` utility does not validate the inputs that you provide during configuration.

`--debug`

Optional. Use this option to view the debug messages on your terminal.

Note

- The `--id`, `--noconsole`, and `--loglevel` options are removed in Instant Messaging version 8.
- The `configure` utility ignores any incorrect or invalid command-line options and proceeds with the configuration process by using the valid options.

Examples of the `configure` Options

The following examples show the different scenarios in which you can use the `configure` command.

To configure through the Command-Line Interface (CLI) mode and save the inputs that you provide in the state file, type the following command:

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

To configure through the CLI mode and use the values from the state file, type the following command:

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

To configure through the silent mode and use the values from the state file, type the following command:

```
./configure --silent <stateFileName>
```

To configure through the CLI mode and use the values from the state file, type the following command. The command saves a state file. It does not do the actual configuration as the `--no` option is used.

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

Sample Configuration Using the configure Utility

The following table lists a sample configuration using default values for all options.

Category and Options	Sample or Default value	Your value
Component Selection		
<pre>Select all components you wish to configure. 1. [X] Server components 2. [X] Client components</pre>	1, 2	
User Management Options		
<pre>Use Access Manager for Single-Sign-On [no]</pre>	no	
<pre>Use Access Manager for Policy [no]</pre>	no	
<pre>Instant Messaging user properties can be maintained using one of the following storage systems: 1. On the file system 2. In the directory Enter the number corresponding to your choice: [1]</pre>	1	
Service Runtime Options		
<pre>Runtime User ID : [inetuser]</pre>	inetuser	
<pre>Runtime Group ID: [inetgroup]</pre>	inetgroup	
<pre>Runtime Directory [/var/opt/SUNWiim]</pre>	/var/opt/SUNWiim	
Network Access Points		
<pre>Domain Name</pre>	foo.sun.com	

XMPP Port [5222]	5222	
Multiplexed XMPP Port [45222]	45222	
XMPP Server Port [5269]	5269	
Disable Server (enable only multiplexor) [no]	no	
LDAP Configuration		
LDAP Host Name [imhost.india.sun.com]	imhost.india.sun.com	
LDAP Port Number [389]	389	
Base DN [dc=india,dc=sun,dc=com]		
Base DN	cn=Directory Manager	
Base Password		
Are you sure you want to use this host for LDAP connections? 1. Choose New 2. Accept Enter the number corresponding to your choice: [1]	1	
Mail Server Options		
Enable Email Integration [yes]	yes	

SMTP Server [imhost]	imhost	
Enable Email Archiving [yes]	yes	
Messenger Resources Download Configuration		
Deploy Messenger Resources [yes]	yes	
Codebase [http://imhost:80/im]	http://imhost:80/im	
Enable Audio? [no]	no	
Webcontainer Path []	Webcontainer base directory	
Web Administration URL []	https://machinename:port	
Web Administrator User Id [admin]	admin	
Web Administrator Password		
HTTP Gateway Deployment Configuration		
Deploy IM HTTP Gateway [yes]	yes	
Context Root [http://imhost:80/httpbind]	http://imhost:80/httpbind	
IMPS Gateway Deployment Configuration		
Deploy IM IMPS Gateway [yes]	yes	

Context Root [http://imhost:80/httpbind]	http://imhost:80/httpbind	
Calendar Agent configuration		
Enable Calendar Agent [no]	no	
Enable local component [no]	no	
SMS Gateway Configuration		
Enable SMS Gateway [no]	no	
Enable local component [no]	no	
MSN Gateway Configuration		
Enable MSN Gateway [no]	no	
Enable local component [no]	no	
AIM Gateway Configuration		
Enable AIM Gateway [no]	no	
Enable local component [no]	no	
Instant Messaging Services Startup Services Startup Configuration		
Start Services After Successful Configuration [yes]	yes	

Start Services When System starts [yes]	yes	
---	-----	--

Chapter 23. Managing Archiving for Instant Messaging

Managing Archiving for Oracle Communications Instant Messaging Server

This chapter explains how to configure and manage email, Portal, and custom archiving for Instant Messaging in the following sections:

- [Archiving Overview](#)
- [Enabling and Disabling Archiving for Instant Messaging](#)
- [Managing the Instant Messaging Email Archive](#)
- [Managing the Instant Messaging Portal Archive](#)
- [Using a Custom Archive Provider](#)

Archiving Overview

You can archive instant messages in the following ways:

- Using the Portal Server Search-based Archive. This method captures instant messages and archives these messages in a Portal Server Search database. End users can query and retrieve archived messages using the Search page on the Portal Server desktop.
- Using the Email Archive. When using this method, chat and conference participants receive emails containing the contents of the Instant Messaging sessions in which they participated. End users can use any email client to search and manage instant messages.
- Using a Custom Archive. You can choose to use either the Instant Messaging archive providers, or create your own custom archive provider. Instant Messaging provides the APIs and SPIs that can be used to write custom archive providers. For more information on Instant Messaging APIs, see [Instant Messaging APIs](#). Regardless of which type of archive provider you choose to use, you need to enable the archive provider in `iim.conf`.

You can configure Instant Messaging to use one or more archive methods at the same time.

Enabling and Disabling Archiving for Instant Messaging

Regardless of whether you choose to use portal, email, a custom archive, or any combination of archives, you enable the archiving capability in Instant Messaging the same way as described in this section. Disabling archiving as described in this section disables all archives.

To Enable Instant Messaging Archiving

After you enable archiving for Instant Messaging, you need to enable the archive provider for the type of archive you want to use as described in the following sections:

- [To Enable the Instant Messaging Email Archive](#)
- [To Enable the Instant Messaging Portal Archive Provider](#)
- [To Enable a Custom Archive Provider](#)

1. Open `iim.conf`.

See [iim.conf File Syntax](#) for information.

2. Add the following line to `iim.conf` if it does not already exist.

```
iim_server.msg_archive = true
```

3. Save and close `iim.conf`.
4. Refresh the server.

```
imadmin refresh server
```

To Disable Instant Messaging Archiving

This procedure disables all archiving for Instant Messaging. If you want to disable only email archiving, Portal archiving, or a custom archive you have configured, see one of the following sections:

- [To Disable the Instant Messaging Email Archive Provider](#)
- [To Disable the Portal Archive Provider](#)
- [To Disable a Custom Archive Provider](#)

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for information.
2. Set the `iim_server.msg_archive` parameter to `false`.

```
iim_server.msg_archive = false
```

3. Save and close `iim.conf`.
4. Refresh the server.

```
imadmin refresh server
```

Managing the Instant Messaging Email Archive

You can use Instant Messaging to archive poll, chat, conference, news channel, and alert content and email that content to end-users and administrators. You can use any email client to search and manage the archived content. This section describes the Instant Messaging email archive in the following sections:

- [Enabling and Disabling the Instant Messaging Email Archive Provider](#)
- [Configuring Email Archive Settings](#)
- [Email Header Format](#)

The Instant Messaging server caches archived records until they are emailed. If you enable email archiving, the memory requirements for the server increase. See the [Communications Suite Deployment Planning Guide](#) for information on performance tuning.

Enabling and Disabling the Instant Messaging Email Archive Provider

You enable or disable the email archive provider by modifying a parameter value in `iim.conf`.

To Enable the Instant Messaging Email Archive

Ensure that you have enabled archiving for Instant Messaging as described in [To Enable Instant Messaging Archiving](#).

1. Open the `iim.conf` file.
See [iim.conf File Syntax](#) for more information.
2. Add the following line to the `iim.conf` file if it does not already exist.

```
iim_server.msg_archive.provider =  
com.ipplanet.im.server.EmailIMArchive
```

The `iim_server.msg_archive.provider` parameter contains a comma-separated list of archive providers. If you want to enable the Portal archive in addition to the email archive for example, the parameter and its value should be entered as follows:

```
iim_server.msg_archive.provider =  
com.ipplanet.im.server.IMPSArchive, \  
com.ipplanet.im.server.EmailIMArchive
```

3. Save and close `iim.conf`.
4. Refresh the Instant Messaging server configuration.

```
imadmin refresh
```

To Disable the Instant Messaging Email Archive Provider

1. Open the `iim.conf` file.
See [iim.conf File Syntax](#) for more information.
2. Delete the `com.ipplanet.im.server.EmailIMArchive` value from the `iim_server.msg_archive.provider` parameter.
3. Save and close `iim.conf`.
4. Refresh the Instant Messaging server configuration.

```
imadmin refresh
```

Configuring Email Archive Settings

You can configure which administrators will receive emails 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 to use the extended RFC 822 header. Doing so allows mail clients to filter messages based on the header content.



Note

If you run `configure` after modifying these parameters for the email archive, any values you input will be overwritten.

[Table 18-1](#) describes the configuration parameters you use to define which administrators will receive email archives, as well as whether or not to use the extended RFC 822 header, and the content of that header.

Table 18-1 Email Archive Configuration Parameters

Parameter	Default Value	Description
<i>iim_arch.admin.email</i>	Empty String	Comma-separated list of administrator email addresses.
<i>iim_arch.alert.admin.email</i>	None	Comma-separated list of administrator email addresses to which all archived alert messages will be sent. This parameter overrides <i>iim_arch.admin.email</i> for alert messages.
<i>iim_arch.chat.admin.email</i>	None	Comma-separated list of administrator email addresses to which all archived chat messages will be sent. This parameter overrides <i>iim_arch.admin.email</i> for chat messages.
<i>iim_arch.conference.admin.email</i>	None	Comma-separated list of administrator email addresses to which all archived conference messages will be sent. This parameter overrides <i>iim_arch.admin.email</i> for conference messages.
<i>iim_arch.poll.admin.email</i>	None	Comma-separated list of administrator email addresses to which all archived poll messages will be sent. This parameter overrides <i>iim_arch.admin.email</i> for poll messages.
<i>iim_arch.news.admin.email</i>	None	Comma-separated list of administrator email addresses to which all archived news messages will be sent. This parameter overrides <i>iim_arch.admin.email</i> for news messages.
<i>iim_arch.email.archiveheader.name</i>	None	Name of the extended RFC 822 header.
<i>iim_arch.email.archiveheader.value</i>	all	Value corresponding to the header name for <i>iim_arch.email.archiveheader.name</i> .

To Configure Administrator Recipients and the RFC 822 Header Format for the Instant Messaging Email Archive

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for more information.
2. Add the parameters in [Table 18-1](#) and appropriate values to `iim.conf`.
3. Refresh the server.

```
imadmin refresh
```

Email Header Format

The RFC 822 header content for email messages containing various types of archived Instant Messaging content is described in the following sections:

- [RFC 822 Email Archive Header Fields for One to One Chat](#)
- [RFC 822 Email Archive Header Fields for Private Conferences](#)
- [RFC 822 Email Archive Header Fields for Public Conferences](#)

- RFC 822 Email Archive Header Fields for Poll Questions with Replies
- RFC 822 Email Archive Header Fields for Poll Replies Only
- RFC 822 Email Archive Header Fields for Alerts
- RFC 822 Email Archive Header Fields for News Channel Posts

RFC 822 Email Archive Header Fields for One to One Chat

From:	Chat session initiator.
To:	Receiver and any administrators configured in iim.conf. See Table 18-1 for more information.
Subject:	First useful message over 50 characters in length.
Date:	Creation date of the email message by the archive provider.
Reply-to:	Not used.
Message-ID	Generated by the email archive provider based on the message thread.

RFC 822 Email Archive Header Fields for Private Conferences

From:	Chat session initiator.
To:	Other participants and any administrators configured in iim.conf. See Table 18-1 for more information.
Cc:	Chat session initiator.
Subject:	If a subject is set for the conference, the conference subject is used. If no subject is set, first useful message over 50 characters in length is used.
Date:	Creation date of the email message by the archive provider.
Reply-to:	Not used.
X-XMPP-Message-ID	Generated by the email archive provider based on the conference ID.

RFC 822 Email Archive Header Fields for Public Conferences

From:	Room owner in archive data.
To:	Associated mailing list, users with explicit access to the conference room, and any administrators configured in iim.conf. See Table 18-1 for more information.
Cc:	Not used.
Subject:	[Conference name] subject.
Date:	Creation date of the email message by the archive provider.
Reply-to:	Not used.
X-XMPP-Message-ID	Generated by the email archive provider based on the conference ID.

RFC 822 Email Archive Header Fields for Poll Questions with Replies

```
From:          Poll sender.
To:           Poll sender and any administrators configured
              in iim.conf. See Table 18-1 for more information.
Cc:          Not used.
Subject:      Poll question.
Date:        Creation date of the email message by the archive
              provider.
Reply-to:     Not used.
X-XMPP-Message-ID  Generated by the email archive provider.
```

RFC 822 Email Archive Header Fields for Poll Replies Only

```
From:          Poll sender.
To:           Poll recipients and any administrators configured in
              iim.conf. See Table 18-1 for more information.
Cc:          Poll sender.
Subject:      Poll question.
Date:        Creation date of the email message by the archive
              provider.
Reply-to:     Not used.
X-XMPP-Message-ID  Generated by the email archive provider.
```

RFC 822 Email Archive Header Fields for Alerts

```
From:          Alert sender.
To:           Alert recipient and any administrators configured
              in iim.conf. See Table 18-1 for more information.
Cc:          Not used.
Subject:      Alert subject.
Date:        Creation date of the email message by the archive
              provider.
Reply-to:     Not used.
X-XMPP-Message-ID  Generated by the email archive provider.
```

RFC 822 Email Archive Header Fields for News Channel Posts

```
From:          News channel post sender.
To:           Mailing list associated with the news channel
              and any administrators configured in iim.conf.
              See Table 18-1 for more information.
Cc:          Not used.
Subject:      News channel post subject.
Date:        Creation date of the email message by the archive
              provider.
Reply-to:     Not used.
X-XMPP-Message-ID  Generated by the email archive provider.
```

Managing the Instant Messaging Portal Archive

The following topics describe using the Instant Messaging Portal Archive:

- [Instant Messaging Portal Archive Overview](#)
- [Enabling and Disabling the Portal Archive Provider](#)
- [Configuring the Instant Messaging Portal Archive Provider](#)
- [Managing Archived Data in the Portal Server Search Database](#)
- [Changing the Display of Archived Data](#)
- [Sample Deployment Scenario for Archive Provider](#)

Instant Messaging Portal Archive Overview

Features of the Instant Messaging Portal Archive Provider include the following:

- It captures all the Instant Messaging traffic passing through the server.
- The archived data can be stored under separate categories in the Portal Server Search. Storing the data as separate categories helps in simplifying the search and retrieval of the archived data. The search can be performed using the Portal Server desktop.
- The security feature of Portal Server Search can be used to provide an access control list. The archive provider provides security features by which only a set of administrative users can be allowed to access the archived data.
- The data can be managed using the Portal Server Search database management tools.

All instant messages are divided into the following categories for the purpose of archiving:

Chat - All messages in the private conference rooms.

Conference - All messages in the public conference rooms.

Alerts - All alert messages.

Poll - All poll messages.

News - All messages posted in the news channels.

The Instant Messaging Portal Archive contains the following components:

Archive and Retrieval Component - Portal Server Search component, also known as the Archive and Retrieval component, is used to store archived Instant Messages. The Instant Messaging archive data is indexed and can be stored in the Portal Server Search database. You can also assign categories to the archive data. For example, you can store alert messages under the Alert category. Storing data in separate categories helps to simplify search operations and enables quick retrieval of archived data.

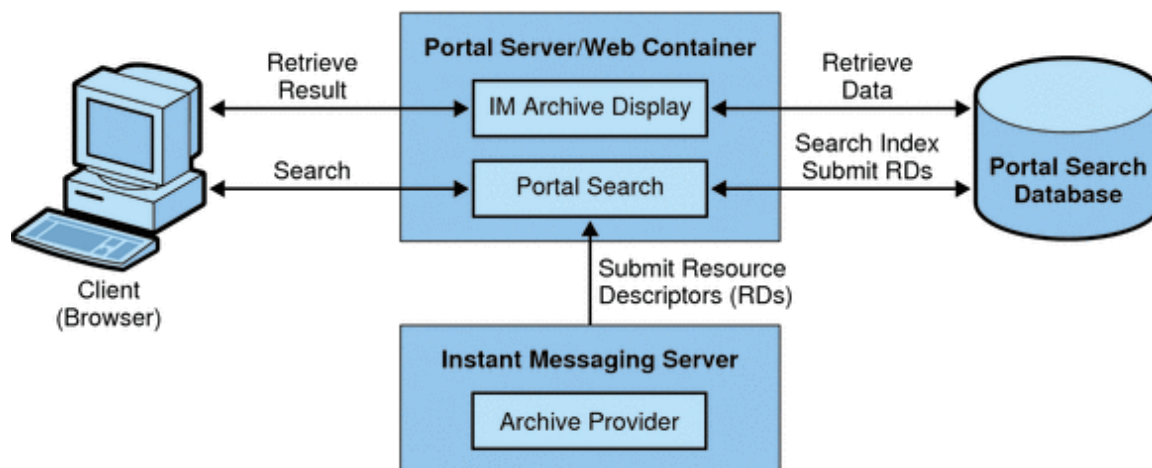
Instant Messaging Archive Search or Display Servlet - When the end user performs a search operation for documents matching certain criteria, the Portal Server Search fetches pages matching this criteria. These pages can be remote web pages or Instant Messaging archive data, also referred as Instant Messaging resource descriptors.

- For remote web pages, the URL of the pages matching the criteria is listed in the Search Results List. When the end user clicks the URL of a web page in the Search Results List, the browser fetches this page from the remote web container.
- For Instant Messaging Resource Descriptors, the archive data is stored in the Portal Server Search database and is not available as downloadable documents from the web container. When the end user clicks the URL of the Instant Messaging resource descriptors to view the archive data, the Instant Messaging Archive Search or Display servlet is invoked. The Instant Messaging Archive Search servlet retrieves the information from the Portal Server Search database and generates a text or HTML response containing the Instant Messaging Archive data.

Instant Messaging Archive Provider - This component is invoked by the Instant Messaging server whenever instant messages are to be archived. The Instant Messaging Archive Provider builds the Summary Object Interchange Format (SOIF) compliant Resource Descriptors (RD) based on the data

provided by the Instant Messaging server. The Archive Provider uses Portal Server Search APIs to send these Resource Descriptors to the Portal Server Search database, and maintains a buffer of the records to be submitted to the Portal ServerSearch database to reduce the performance hit.

Figure 18-1 Instant Messaging Portal Archive components.



Enabling and Disabling the Portal Archive Provider

You enable the Instant Messaging Archive Provider or your custom archive provider by modifying parameters in `iim.conf`.

To Enable the Instant Messaging Portal Archive Provider

Ensure that you have enabled archiving for Instant Messaging as described in [To Enable Instant Messaging Archiving](#).

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Add a line to `iim.conf` for the type of archive provider you want to enable.
For a custom archive provider, add the following line:

```
iim_server.msg_archive.provider = provider-name
```

To use the Portal Server Search-based Archive Provider, replace *provider-name* with the following:

```
com.iplanet.im.server.IMPSArchive
```

The `iim_server.msg_archive.provider` parameter contains a comma-separated list of archive providers. If you want to enable the Portal archive in addition to the email archive for example, the parameter and its value should be entered as follows:

```
iim_server.msg_archive.provider =
com.iplanet.im.server.IMPSArchive, \
com.iplanet.im.server.EmailIMArchive
```

3. If you are running Sun Java System Portal Server 7 2006Q1 or later, provide a value for the

following parameter:

```
iim_arch.portal.search="_Portal Server Search URL_"
```

Where *Portal Server Search URL* is the Search URL for the Portal Server. For example:

```
iim_arch.portal.search="http://portal.siroe.com:8080/search1/search"
```

4. Save and close `iim.conf`.
5. Refresh the Instant Messaging server configuration.

```
imadmin refresh
```

6. Log in to `psconsole` as `amadmin`.
For instructions, refer to the Portal Server documentation.
7. Select Manage Channels and Containers.
8. Select the portal and organization that will host the search function.
9. Select IMChannel from the DP XML Tree View.
10. Enter the search server URL as the value for “searchServer”.
For example:

```
http://portal.siroe.com:8080/search1/search
```

11. Save properties.

To Disable the Portal Archive Provider

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for more information.
2. Delete the `com.iplanet.im.server.IMPSIMArchive` value from the `iim_server.msg_archive.provider` parameter.
3. Save and close `iim.conf`.
4. Refresh the Instant Messaging server configuration.

```
imadmin refresh
```

Configuring the Instant Messaging Portal Archive Provider

The Instant Messaging Archive Provider stores the archived messages as resource descriptors (RD) in the Portal Server Search database. The archive provider uses the following fields of the Portal Server Search schema:

Title - This field contains the names of the public conference rooms for Conference category, names of the participants in a chat session for the Chat category, subject of the Alert messages, and the names of the News Channels for alerts and news categories. The title field will contain “Poll from *Sender*” for the poll category, where *Sender* represents the display name of the sender of the poll.

Keyword - For conference and chat categories, this field contains a list of all the participants in the conference room. For a public conference room, it also contains the name of the conference room. For the Alert category, it contains the display names of the sender and the recipients. For the News category, it contains the name of the channel. For the Polls category, it contains the list of sender and recipients. For all categories, in addition to the above values this field also contains a unique ID for the categories.

Table 18-2 shows the unique ID and gives a description for each category in the archive provider.

Table 18-2 Unique ID and Description for Archive Provider Categories

Category	Unique ID
ConferenceChat	<i>RoomName-StartTime</i> Where, <i>RoomName</i> - Name of the public or private conference room <i>StartTime</i> - Timestamp of the creation of RD
Alert	<i>Alert-messageID</i> Where, <i>messageID</i> - Message ID of the message which will be archived. Message ID has importance when the RD contains only one message. For example, News message and Alert message.
Poll	<i>Poll-pollID</i>
News	<i>TopicName-messageID</i>

ReadACL - For the Conference and News categories, the value for this field is taken from the access control files of the respective conference rooms and news channels. For the Chat category, this field contains the DN of the participants. For the Alert category, this field contains the sender's DN and the recipient's DN. For the Poll category, the archive provides a new access control file.

The search access to the RDs is controlled by the value in the ReadACL field. If the document level security is enabled, the end user has access to the search results only if the ReadACL field has the end user's DN.

Description - This field contains the archived message without the HTML formatting.

Full-Text - This field contains the HTML formatted archived messages.

Classification - This field contains the category of the archived message.

To Configure the Archive Provider

1. Open `iim.conf`.
See [Instant Messaging Configuration Parameters in `iim.conf`](#) for instructions on locating and modifying `iim.conf`.
2. Add or edit the archive provider configuration parameters as desired.
See [Table A-8](#) for a list of parameters you can modify.
3. Save and close `iim.conf`.
4. Refresh the Instant Messaging server.

To Store Archived Messages in a Non-default Database

Use this procedure to configure Instant Messaging to store archived messages in a database other than the default.

1. Open `iim.conf`.
See [Instant Messaging Configuration Parameters in `iim.conf`](#) for instructions on locating and modifying `iim.conf`.
2. For the default archive provider, add the following line:

```
iim_arch.portal.search.database = database-name
```

where *database-name* is the name of your non-default database.

3. Save and close `iim.conf`.
4. Modify the Portal Server Search Channel.
Change the Portal Server Search Channel to add an option for searching the data in another database. See the Sun Java System Portal Server Desktop Customization Guide for more information.
5. Change to the `IMProvider` directory.
For example:

```
cd /etc/opt/SUNWps/desktop/default_locale/IMProvider/
```

Where *locale* is the locale of the language used in your deployment. For example, `default_ja` or `en_US`. Also, if you created multiple instances of Instant Messaging, the name of the `/default` directory will vary depending on the instance.

6. Create a back up of the `IMArchiveDisplay.jsp` file.
7. Open the `IMArchiveDisplay.jsp` file.
8. Search through the `IMArchiveDisplay.jsp` file and locate the following two lines of code:

```
<search:setQuery query = "<%= scope %>" />
<search:setRDType rdmType = "rd-request" />
```

9. Between the two lines of code shown in the previous step, add the following line of code:

```
<search:setDatabase database = "database-name" />
```

After you add the new line of code, that section of code should look as follows:

```
<search:setQuery query = "<%= scope %>" />
<search:setDatabase database = "_database-name_" />
<search:setRDType rdmType = "rd-request" />
```

where *database-name* is the name of the non-default database.

10. Replace the virtual search server with the physical server hostname.
11. Save and close `IMArchiveDisplay.jsp`.

Managing Archived Data in the Portal Server Search Database



Note

These instructions are Solaris-specific.

The Instant Messaging data is archived in the form of Resource Descriptors (RDs) in the Portal Server Search database. The individual entries in the Portal Server Search database are called resource descriptors (RDs). An RD is a specific set of information about a single resource. The fields of each RD are determined by the Portal Server Search database schema.

To manage the archived data, you need to manage the Resource Descriptors (RDs) in the Portal Server Search database. This section explains some of the frequently performed Portal Server Search database maintenance tasks.

For more information on managing data in the Portal Server Search database, see the Sun Java System Portal Server Administration Guide.

rdmgr Command

The `rdmgr` command is the main command used to work with the Search service. It gives the administrator two types of subcommands: one that is used to work with resource descriptors (RDs), and another used for database maintenance. The `rdmgr` command is normally run in a search-enabled Portal Server instance directory.

To Invoke the rdmgr Command

1. Change to the `https-<servername>` directory.
`cd /var/opt/SUNWps/https-servername`
Where *servername* is the name of the Portal Server
2. Type the following at the command-line:
`run-cs-cli portal-svr-base/SUNWps/bin/rdmgr options`
where *portal-svr-base* is the directory in which Portal Server is installed.
For more information on `rdmgr` command, see Command-Line Utilities in Sun Java System Portal Server Administration Guide.

Searching Resource Descriptors

Running `rdmgr` command with the argument value `-Q` generates a list of resource descriptors (RDs) that refines the search operation.

For example:

- To search for resource descriptors (RDs) containing the text `testing`, type:
`run-cs-cli portal-svr-base/SUNWps/bin/rdmgr -Q testing`
- To search for resource descriptors (RDs) belonging to a particular category, type the following command. Enter the command as a single line:
`run-cs-cli portal-svr-base/SUNWps/bin/rdmgr -Q "classification=Archive:Chat:January"`

Deleting Resource Descriptors

The following are the examples for deleting resource descriptors (RDs) from the Portal Server Search database:

To delete all resource descriptors (RD) containing the text `testing`, type:

```
run-cs-cli portal-svr-base/SUNWps/bin/rdmgr -d -Q testing
```

To delete all resource descriptors (RD) from a category `Archive:Chat:January`, type the following command. Enter the command as a single line:

```
run-cs-cli portal-svr-base/SUNWps/bin/rdmgr -d -Q "classification=Archive:Chat:January"
```

Changing the Display of Archived Data

The data that is archived is deployed using the `IMArchiveDisplay.jsp` file. The

IMArchiveDisplay.jsp file is installed in the folder /etc/opt/SUNWps/desktop/default/IMProvider by default. You can modify this file to change the style and the resource strings of the archived data.

For example, you can replace the default system message displayed when an end user joins the room as described in the following steps.

Similarly, the resource strings for the other keys and the style for displaying the key information can also be modified.

If you change the attribute name of Title and Full-Text in the default schema of the Portal Server Search is changed, then these changes should also be reflected in the IMArchiveDisplay.jsp file.

To Modify the Default System Message

1. Edit IMArchiveDisplay.jsp.
2. Search for the following the code lines in IMArchiveDisplay.jsp:

```
....  
ht.put("has_joined_the_room", "&lt;span class='user'> {0} &lt;/span>  
&lt;span class='headervalue'> has joined the room.&lt;/span>");  
....
```

3. Replace the headervalue with the desired text.
For example:

```
....  
ht.put("has_joined_the_room", "&lt;span class='user'> {0} &lt;/span>  
&lt;span class='headervalue'> has entered the room.&lt;/span>");  
....
```

Sample Deployment Scenario for Archive Provider

This sample deployment scenario explains how to archive the related Instant Messaging data collectively.

Example 18-1 Archiving Related Instant Messaging Data Collectively

Create separate categories for each type of data. For example, in the Archive category where all the archived Instant Messaging data are stored, create a subcategory called "Chat" for storing chat messages. You can also create subcategories for archiving data based on time. For example, to archive chat data for the month of December 2002 the subcategory will be:

```
Archive:Chat:2002:12
```

To Archive All Instant Messaging Chat Data Based on Time

1. Change to the `_im-cfg-base_` directory.
See [Instant Messaging Server Directory Structure](#) for information on locating `im-cfg-base`.
2. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
3. Add the following value for `iim_arch.chat.categoryname`:
`iim_arch.chat.categoryname = Archive:Chat:%Y:%M`
The archive provider automatically assigns the current year for `%Y` and current month for `%M`.

These values are taken from the system date and time.

To Archive and Back up Instant Messaging Chat Data for the Month of December 2005 to the Subcategory

1. Type the following:

```
rdmgr -Q "classification=Archive:Chat:2005:12" > archive.soif
```

2. Copy the `archive.soif` file to your backup system.

To Remove Archived Instant Messaging Chat Data for the Month of December 2005 from the Portal Server Search Database

1. Type the following at the command line:

```
rdmgr -d "classification=Archive:Chat:2005:12"
```

Using a Custom Archive Provider

In addition to the Portal and email archives, you can choose to use a custom archive provider.

To Enable a Custom Archive Provider

Ensure that you have enabled archiving for Instant Messaging as described in [To Enable Instant Messaging Archiving](#).

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Add a line to `iim.conf` for the type of archive provider you want to enable.
For a custom archive provider, add the following line:

```
iim_server.msg_archive.provider = provider-name
```

To use the Portal Server Search-based Archive Provider, replace *provider-name* with the following:

```
com.iplanet.im.server.IMPSArchive
```

The `iim_server.msg_archive.provider` parameter contains a comma-separated list of archive providers. If you want to enable the Portal archive in addition to the email archive for example, the parameter and its value should be entered as follows:

```
iim_server.msg_archive.provider =  
com.iplanet.im.server.IMPSArchive, \  
com.iplanet.im.server.EmailIMArchive
```

3. Save and close `iim.conf`.
4. Refresh the Instant Messaging server configuration.

```
imadmin refresh
```

To Disable a Custom Archive Provider

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for more information.
2. Delete only the value for the custom archive provider from the `iim_server.msg_archive.provider` parameter.
3. Save and close `iim.conf`.
4. Refresh the Instant Messaging server configuration.

```
imadmin refresh
```

Chapter 24. Migrating the Property Store From File to LDAP

Migrating the Property Store

This chapter describes the steps to migrate a file-based property store to an LDAP-based property store by using the `migratepropstore` command. This chapter contains the following topics:

- [Migrating the Property Store From File to LDAP](#)
- [migratepropstore Command](#)

Migrating the Property Store From File to LDAP

From a previous or current version of Instant Messaging, user data can be migrated between a file-based property store to LDAP-based property store. To migrate user data, perform the following steps from the working setup:

1. Stop the Instant Messaging server.
2. Upgrade to the latest version.

Note

This step is not required if you are using Instant Messaging version 7.3.

3. Run the Configurator tool with the same property store location. For example, to migrate data from LDAP, run the Configurator tool with the LDAP option.

Note

This step is required only if you have upgraded Instant Messaging and want to migrate data.

4. Stop the Instant Messaging server again.
5. Type the `migratepropstore` command on the command prompt.

```
<install-directory>/imadmin migratepropstore -v | --verbose -f  
|--force  
-l | --log <log-file> -e | --error_log <error-log-file> -s |  
--source <source-property-store>  
-d | --destination <destination-property-store> -m | --mconfig  
<migrator-config-file>
```

To get the property source from the `iim.conf` file, add the `-c` option.

```
<install-directory>/imadmin migratepropstore -v -f -l logfile -e  
errorlog  
-s ldap -d file -c /opt/sun/comms/im/config/iim.conf  
-m /opt/sun/comms/im/sbin/mconfig.conf
```

6. Change to the new property store in the `iim.conf` file.
7. Restart the Instant Messaging server and log in to the Instant Messaging client.

**Note**

You can selectively migrate errors, provided you have saved the errors in an error log file. See `./migratepropstore --help` for more details.

migratepropstore Command

This section describes the `migratepropstore` command. The `migratepropstore` command has the following syntax:

```
<install-directory>/imadmin migratepropstore -v | --verbose -f  
|--force  
-l | --log <log-file> -e | --error_log <error-log-file> -s | --source  
<source-property-store> -d | --destination <destination-property-store>  
  
-m | --mconfig <migrator-config-file>
```

Example:

```
/opt/sun/im/sbin/imadmin migratepropstore --verbose --force --log log1  
--error_log errorlog1 --source ldap --destination file  
--mconfig mconfig.conf
```

Command-line Options

`-h | --help`

Optional. Displays help content for this command.

`-v | --verbose`

Optional. Prints information messages to the standard output.

`-f | --force`

Optional. Forces the command to continue even in case of severe failures.

`-l | --log`

Optional. Name of the log file to record the progress of migration.

`-e | --error_log`

Optional. Name of the log file to record migration errors. The error logs recorded in this log file enable you to selectively migrate only those files that failed in a previous migration. Every time you use this command, save the log file with a different name. Maintaining different log files ensure that the log file is not overwritten every time you use this command.

`-s | --source`

Mandatory. Specifies the source of the property store type. The value is `file` or `ldap`.

`-d | --destination`

Mandatory. Specifies the source of the property store type. The value is `file` or `ldap`.

`-c | --config`

Optional. Specifies the path of the `im.config` file.

`-m | --mconfig`

Mandatory for identity realm and optional for LDAP realm. Specifies the path of the migrator config file. If you are using the Identity realm, specify the LDAP `binddn` details here.

Chapter 25. Migrating the Multiplexor Certificate and Enabling SSL

Migrating the Multiplexor Certificate and Enabling SSL

Instant Messaging enables migration of private key and Multiplexor certificate from the Network Security Services (NSS) database to Java keystore (JKS). This chapter describes the `migratecert` command and explains the steps to enable SSL on the Instant Messaging client side.

To migrate the certificate using the `migratecert` command, perform the following steps:

1. Install and configure Instant Messenger.
2. Install the NSS-based certificate for Instant Messaging Multiplexor to communicate in the SSL mode and add the following parameters in the `iim.conf` file:

```
iim_mux.usessl=<on/off>
iim_mux.seconfigdir=
iim_mux.keydbprefix=
iim_mux.certdbprefix=
```

The following code shows a sample `iim.config` file:

```
iim_mux.usessl=on
iim_mux.seconfigdir=/opt/sun/comms/im/config/certs/
iim_mux.keydbprefix=https-test.siroe.com-test-
iim_mux.certdbprefix=https-test.siroe.com-test-
```

3. Add the following parameters in the `iim.conf` file to specify the destination where the certificate should be migrated.

```
iim_mux.sslkeystore=<keystorefilename>
iim_mux.keystorepasswordfile=<passwordfilename>
```

The following code shows a sample `iim.config` file:

```
iim_mux.sslkeystore=/opt/sun/comms/im/config/india_sun_com.jks
iim_mux.keystorepasswordfile=/opt/sun/comms/im/config/sslpassword.conf
```



Note

Make sure that the `iim_mux.sslkeystore` parameter is specified as a full path. Multiplexor does not read the certificate if the parameter path is not complete.

4. Type the `/opt/sun/comms/im/sbin/imadmin migratecert` command.
The certificate is migrated to the keystore that you specified in step 3 from the source that you specified in step 2.
5. Edit the `iim.conf` file to uncomment the following parameters:

```
!iim_mux.seconfigdir=  
!iim_mux.keydbprefix=  
!iim_mux.certdbprefix=
```

To enable SSL on the client side, do the following:

1. In the `im.jnlp` file, modify the `usessl` element, which is the child element of `<application-desc ... >`.
`<argument>usessl=true</argument>`
2. In the `im.html` file, modify the `param` element, which is child element of `<OBJECT ... >`.
`<PARAM NAME="usessl" VALUE="true">`
3. Restart the Instant Messaging server and client.
The Instant Messaging client can now communicate with the Multiplexor in the SSL mode.

Chapter 26. Managing Instant Messaging's LDAP Access Configuration

Managing Oracle Communications Instant Messaging Server's LDAP Access Configuration

This information describes how Instant Messaging uses LDAP in deployments with and without Access Manager in the following sections:

- [Overview of How Instant Messaging Uses LDAP](#)
- [Searching the Directory Anonymously](#)
- [Configuring Instant Messaging to Use LDAP Groups](#)

Overview of How Instant Messaging Uses LDAP

All deployments of Instant Messaging require a directory server. In a deployment without Access Manager, the Instant Messaging server uses the directory server to perform end-user authentication and to search for end users.

In a deployment with Portal Server, the Instant Messaging server uses the directory used by Portal Server. When installed in an Access Manager deployment environment, the Instant Messaging server uses the directory used by the Access Manager to search for end users, and not for end-user authentication. In an Access Manager deployment, Access Manager performs the authentication.

If you use an LDAP directory to maintain your user namespace, the default 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.
- Instant Messenger user ID attribute of an end user is provided by the `uid` attribute (from `inetOrgPerson` objectclass).
- 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).

You can change these default settings by editing the `iim.conf` file. See [iim.conf File Syntax](#).



Caution

Some user attributes may contain confidential information. Ensure that your directory access control is set up to prevent unauthorized access by non-privileged users. Refer to your directory documentation for more information.

Searching the Directory Anonymously

Instant Messaging needs to be able to search the directory to function correctly. If your directory is configured to be searchable by anonymous users, Instant Messaging has the capability to search the

directory. If the directory is not readable or searchable by anonymous users, you must take additional steps to configure `iim.conf` with the credentials of a user ID that has at least read access to the directory. These credentials consist of:

- A distinguished name (dn)
- The password of the above (dn)

To Enable the Server to Conduct Directory Searches as a Specific End User

1. Identify values for the following parameters in `iim.conf`:

`iim_ldap.usergroupbinddn` - Specifies the distinguished name (dn) to use to bind to the directory for searches.

`iim_ldap.usergroupbindcred` - Specifies the password to use with the distinguished name (dn).

For example:

```
iim_ldap.usergroupbinddn="cn=iim_server,o=i-zed.com"
```

```
iim_ldap.usergroupbindcred=secret
```

Note

You do not have to use administrator-level credentials with write level access, as all that is necessary is read access to the domain tree. Thus, if there is an LDAP user with read level access, use its credentials instead. This is a safer alternative as it does not force you to disseminate the administrator-level credentials.

See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.

2. In a deployment with Sun Java System Access Manager, if the directory is not searchable by anonymous users:

Set the `iim_ldap.useidentityadmin` configuration parameter to `true`. Also, you can delete or comment out the following configuration parameters:

```
iim_ldap.usergroupbinddn
```

```
iim_ldap.usergroupbindcred
```

3. Edit `iim.conf`.

See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.

If the `iim_ldap.usergroupbinddn` and `iim_ldap.usergroupbindcred` parameters do not appear in `iim.conf`, you can add them anywhere in the file.

Configuring Instant Messaging to Use LDAP Groups

You can configure Instant Messaging so that end users can send a message to an LDAP group, which can be either dynamic or static.

- LDAP dynamic group: Membership, rather than being maintained explicitly in a list, is determined by search criteria using an LDAP URL. Dynamic groups use the `groupOfURLs` object class and the `memberURL` attribute to define LDAP URLs with the criteria (search base, scope, and filter) to be used for determining members of the group.
- LDAP static group: A static group is one whose entry contains a membership list of explicit DNs. You can define a static group by using the `groupOfUniqueNames` object class and by explicitly specifying the member DNs by using the `uniqueMember` attribute.

To enable end users to view dynamic and static LDAP groups in search results and add them to their instant messaging client contact list, you need to include `groupOfURLs`, and `groupOfUniqueNames` objects in search results. The following section describes how to configure Instant Messaging to do this.

Topics in this section:

- [To Configure Instant Messaging to Use LDAP Groups](#)
- [To Use Group Messaging](#)

In the Directory Server and some other LDAP servers, dynamic groups filter end users based on their DN and include them in a single group. The dynamic groups are defined in Directory Server by the `groupOfUrls` object class.

To enable end users to view the dynamic groups in search results and add them to their contact list, you need to include `groupOfUrls` objects in search results.

To Configure Instant Messaging to Use LDAP Groups

The ability to perform group messaging was introduced in **Instant Messaging 8 Update 3 Patch 5**.

1. If you have not already done so, create the LDAP group to be used for group messaging. See [Managing Groups](#).
2. Edit the `iim.conf` file to set the `iim_server.group.servicename` parameter, if the service name for group messaging is to be changed from the default name `groups`.
For example:

```
iim_server.group.servicename = mygroups
```

See [iim.conf File Syntax](#) for instructions on locating and modifying the `iim.conf` file.

3. Add the appropriate content to the `iim.conf` file, depending on if you want to search for dynamic or static groups.
 - To search for dynamic groups, add the following two lines:

```
iim_ldap.usergroupbynamefilter="( | (& ((objectclass=groupofurls)
(cn={0})) (&(objectclass=inetorgperson)(cn={0}))) )"

iim_ldap.groupclass=groupofurls
```

- To search for static groups, set the following two parameters:

```
iim_ldap.usergroupbynamefilter="( | (&(objectclass=groupofunique
(cn={0})) (&(objectclass=inetorgperson)(cn={0}))) )"

iim_ldap.groupclass=groupofuniquenames
```



Note

Static groups can also be inherited from `groupofnames` object class, and their members listed using `member` attribute. However, the search filters for static groups must be modified accordingly. By default, the `member` attribute is not used as the membership attribute of a static group. Hence, the parameter must be set to `iim_ldap.groupmemberattr=member` to use `member` attribute. Do not include line breaks within a single line. The attribute and object class names are configurable. By default, the `memberOfUrls` attribute is used as the membership attribute of a dynamic group. If you want to use an attribute name other than `memberOfUrls`, set the `iim_ldap.groupmemberurlattr` option to the attribute name you want to use.

- To search for both dynamic and static groups, set the following parameters:

```
iim_ldap.usergroupbynamefilter="( | (& ( | (objectclass=groupofuniquenames) (cn={0})) (& (objectclass=inetorgperson) (cn={0}))) ) ) "
```

```
iim_ldap.groupclass=groupofuniquenames,groupofurls
```

4. Save the changes to the `iim.conf` file.
5. To send a message to a group, see the next procedure, [To Use Group Messaging](#).

To Use Group Messaging

1. In the client's chat window, type the group's full Jabber ID in the form `groupName@group.domainname` in the To tab.
For example: `testGroup@mygroups.example.com`
2. Type the message and click send.

Chapter 27. Managing Instant Messaging and Presence Policies

Managing Oracle Communications Instant Messaging Server and Presence Policies

Instant Messaging provides various functional features such as chat, conferencing, polls, presence access, etc. A policy describes a set of access control privileges that can be associated with these features. In turn, end users and groups can be assigned to policies according to the needs of an organization.

This chapter describes how to define and use policies to manage the access that end users and administrators have to the Instant Messaging server features and privileges:

- [Overview of Privacy, Security, and Site Policies](#)
- [Methods for Controlling End User and Administrator Privileges](#)
- [Managing Policies Using Access Control Files](#)
- [Managing Policies using Access Manager](#)

Overview of Privacy, Security, and Site Policies

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

Site Policies

Site policies specify end-user access to specific functionality in Instant Messaging. Site policies specify the ability to:

- Access the presence status of other end users
- Send alerts to other end users
- Save properties on the server
- Create and manage conference rooms
- Create and manage news channels

The Instant Messaging administrator has access to all Instant Messaging features. The administrator has `MANAGE` access to all conference rooms and news channels, can view presence information of any end user, and can view and modify properties such as Contact Lists and Instant Messenger Settings of any end user. The site policy settings have no impact on the administrator's privileges.

By default, the end user is provided with the privileges to access the presence status of other end users, send alerts to end users, and save properties to the server. In most of the deployments, the default values are not changed. These default values need to be changed when Instant Messaging is used exclusively for the pop-up functionality.

When Instant Messaging is used exclusively for the pop-up functionality, the end user will not be provided with the access privileges to presence information, chat, and news features.

**Note**

Although certain privileges can be set globally, the administrator can also define exceptions for these privileges. For example, the administrator can deny certain default privileges to select end users, roles, or groups.

Conference Room and News Channel Access Controls

End users can have the following access privileges on Conference rooms and News channels:

- **MANAGE** - full access, which includes the ability to set the conference room or the news channel privilege for other end users.
- **WRITE** - privilege to add contents to the conference room or the news channel.
- **READ** - privilege to read the conference room or the news channel contents.
- **NONE** - no access privileges.

End users with the **MANAGE** privilege can set the default privilege level for all the other end users. These end users can also define the exception rules to grant an access level that is different from the default access level permission given to specific end users or groups.

**Note**

Setting the **WRITE** privilege, also grants the end users the **READ** privilege.

User Privacy

End users can specify whether other end users can see their presence. By default, all end users can access the presence information of all other end users. End users can also set exceptions for denying this access to certain end user and groups.

If an end user has denied other end users from accessing the end user's presence status, then that end user's availability status appears as offline in other end user's contact lists. No alerts or chat invitations can be sent to an end user whose presence status is offline.

User privacy can be configured using the User Settings window in the Instant Messenger. For more information on configuring user privacy, see Instant Messenger Online Help.

[Top](#)

Methods for Controlling 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. There are two methods of policy management available: through access control files or through Access Manager.

- [Managing Policies Using Access Control Files](#) - The access control file method for managing policies allows you to adjust end-user privileges in the following areas: news channel management, conference room management, the ability to change preferences in the User Settings dialog, and ability to send alerts. It also allows specific end users to be assigned as system administrators.
- [Managing Policies using Access Manager](#) - This method gives you control of the same privileges

available with the access control file method; however, it additionally allows more fine-tuned control over various features, such as the ability to receive alerts, send polls, receive polls, etc. For a complete list, see [Table 17-3](#). Furthermore, managing policies using Access Manager gives you finer-tuned control over privileges.

Two types of policies exist, Instant Messaging policies and Presence policies. The Instant Messaging policies govern general Instant Messaging features, such as the ability to send or receive alerts, the ability to manage public conferences and news channels, and the ability to send files. Presence policies govern the control end users have over changing their online status, and in allowing or preventing others from seeing their online or presence information.

If your deployment does not include Access Manager, you must use the access control file method to manage policies. If you are using Access Manager with the Instant Messaging server, and you have installed the Instant Messaging and Presence services components, you can use either policy management method. Managing policies using Access Manager is a more comprehensive method. One advantage of this method is that it allows you to store all end-user information in the directory.

Setting the Policy Management Method

When you choose which method to use to manage policies, you must also choose where they will be stored. Select the method for managing policies by editing the `iim.conf` file and setting the `iim.policy.modules` parameter to either `identity` for the Access Manager method or `iim_ldap` for the access control file method, which is also the default method.

Follow these steps to set which method you want to use to manage policies.

To Set the Policy Management Method

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Edit the `iim.policy.modules` parameter by setting it to one of the following:
 - `iim_ldap` (default, the access control file method)
 - `identity` (the Access Manager method)If you choose `identity`, you can run `imadmin assign_services` to assign Instant Messaging and presence services to existing users.
3. Edit the `iim.userprops.store` parameter and set it to either:
 - `ldap` (To store user properties in LDAP.)
 - `file` (Default, to store user properties in files.)If you choose `ldap`, you can run `imadmin assign_services` to add the required objectclasses that store user properties to user entries in the directory.
4. Save and close `iim.conf`.
5. Refresh the configuration.

Policy Configuration Parameters

[Table 17-1](#) lists and describes the parameters available in `iim.conf` that relate to the increased role that Access Manager can play in Instant Messaging deployments.

Table 17-1 Parameters Related to Access Manager in `iim.conf`

Parameter Name	Use	Values
<i>iim.policy.modules</i>	Indicates if Access Manager or the directory is used for policy storage.	<i>iim_ldap</i> (default) <i>identity</i>
<i>iim.userprops.store</i>	Indicates if the user properties are in a user properties file or stored in LDAP. Only significant when the service definitions for the Presence and Instant Messaging services have been installed.	<i>file</i> (Default if you chose not to use Access Manager for policy when you ran the <i>configure</i> utility.) <i>ldap</i> (Default if you chose to use Access Manager for policy when you ran the <i>configure</i> utility.)

[Top](#)

Managing Policies Using Access Control Files

By editing access control files you control the following end-user privileges:

- Access to the presence status of the other end users
- Send alerts to other end users
- Save properties on the server
- Create new conference rooms
- Create new news channels

By default, end users are provided the privileges to access the presence status of other end users, send alerts to end users, and save properties to the server. For most deployments, default values do not need to be changed.

Although certain privileges can be set globally, the administrator can also define exceptions for these privileges. For example, the administrator can deny certain default privileges to select end users or groups.

In addition, if you are enforcing policy through access control files in your deployment, those files must be the same for all servers in a server pool.

[Table 17-2](#) lists the global access control files for Instant Messaging and the privileges these files provide end users.

Table 17-2 Access Control Files

ACL File	Privileges
<code>sysSaveUserSettings.acl</code>	Defines who can and cannot change their own preferences. Users who do not have this privilege cannot add contacts, create conferences, etc.
<code>sysTopicsAdd.acl</code>	Defines who can and cannot create News channels.
<code>sysRoomsAdd.acl</code>	Defines who can and cannot create Conference rooms.
<code>sysSendAlerts.acl</code>	Defines who can and cannot send alerts. Disabling <code>sysSendAlerts</code> also disables polls.
<code>sysWatch.acl</code>	Defines who can and cannot watch changes of other end users. The Instant Messenger window is displayed for end users who do not have this privilege allowing “conference and news channel subscription and non-subscription” only.
<code>sysAdmin.acl</code>	Reserved for administrators only. This file sets administrative privileges to all Instant Messaging features for all end users. This privilege overrides all the other privileges and gives the administrator the ability to create and manage conference rooms and news channels as well as access to end user presence information, settings, and properties.

To Change End-user Privileges in Access Control Files

1. Change to the `im-cfg-base/acls` directory.
See [Instant Messaging Server Directory Structure](#) for information on locating `im-cfg-base`.
2. Edit the appropriate access control file.
For example:

```
vi sysTopicsAdd.acl
```

See [Table 17-2](#) for a list of access control files.

3. Save the changes.
4. End users need to refresh the Instant Messenger window to see the changes.

Using Access Control Files in a Server Pool

If you are enforcing policy through access control files in your deployment, the content of the files must be the same for all servers in a server pool. To ensure this, copy the files from one server to each of the other nodes in the pool. See [Access Control File Location](#) for information on finding these files.

Access Control File Location

The location of the access control files is `im-cfg-base/acls`. Where `im-cfg-base` is the configuration directory. See [Instant Messaging Server Directory Structure](#) for information about the default location of the configuration directory.

Access Control File Format

The access control file contains a series of entries that define the privileges. Each entry starts with a tag as follows:

- d: - default
- u: - user
- g: - group

The tag is followed by a colon (:). In case of the default tag it is followed by `true` or `false`.

End-user and group tags are followed by the end-user or group name.

Multiple end users and groups are specified by having multiple end users (u) and groups (g) in lines.

The d: tag must be the last entry in an access control file. The server ignores all entries after a d: tag. If the d: tag is `true`, all other entries in the file are redundant and are ignored. You cannot set the d: tag as `true` in an access control file and selectively disallow end users that privilege. If default is set to `false`, only the end users and groups specified in the file will have that particular privilege.

The following are the default d: tag entries in the ACL files for a new installation:

- `sysAdmin.acl` - Contains `d:false`
- `sysTopicsAdd.acl` - Contains `d:true`
- `sysRoomsAdd.acl` - Contains `d:true`
- `sysSaveUserSettings.acl` - Contains `d:true`
- `sysSendAlerts.acl` - Contains `d:true`
- `sysWatch.acl` - Contains `d:true`



Caution

The format and also the existence of all the access control files might change in future releases of the product.



Note

Disabling `sysSendAlerts` also disables polls.

Example 17-1 `sysTopicsAdd.acl` File

In the following example, the d: tag entry for `sysTopicsAdd.acl` file is `false`. Therefore, the Add and the Delete news channels privileges are available to the end users and groups that appear before the d: entry, namely `user1`, `user2`, and the `sales` group.

```
# Example sysTopicsAdd.acl file
u:user1
u:user2
g:cn=sales,ou=groups,o=siroe
d:False
```

[Top](#)

Managing Policies using Access Manager

The Instant Messaging and Presence services in Access Manager provide another way to control end user and administrator privileges. Each service has three types of attributes: dynamic, user, and policy. A policy attribute is the type of attribute used to set privileges.

Policy attributes become a part of the rules when rules are added to a policy created in Access Manager

to allow or deny administrator and end-user involvement in various Instant Messaging features, such as receiving poll messages from others.

When Instant Messaging server is installed with Access Manager, several example policies and roles are created. See the Access Manager Getting Started Guide and the Access Manager Administration Guide for more information about policies and roles.

You can create new policies and assign those policies to a role, group, organization, or end user as needed to match your site's needs.

When the Instant Messaging service or the Presence service are assigned to end users, they receive the dynamic and user attributes applied to them. The dynamic attributes can be assigned to an Access Manager configured role or organization.

When a role is assigned to an end user or an end user is created in an organization, the dynamic attributes become a characteristic of the end user. The user attributes are assigned directly to each end user, they are not inherited from a role or an organization and, typically, are different for each end user. When an end users logs on, they get all the attributes that are applicable to them depending upon which roles are assigned to them and how the policies are applied.

Dynamic, user or policy attributes are associated with end users after assigning the Presence and Instant Messaging Services to these end users.

Instant Messaging Service Attributes

Table 17-3 lists the policy, dynamic, and user attributes for each service.

Table 17-3 Access Manager Attributes for Instant Messaging

Service	Policy Attribute	Dynamic Attributes	User Attributes
sunIM	sunIMAllowChat sunIMAllowChatInvite sunIMAllowForumAccess sunIMAllowForumManage sunIMAllowForumModerate sunIMAllowAlertsAccess sunIMAllowAlertsSend sunIMAllowNewsAccess sunIMAllowNewsManage sunIMAllowFileTransfer sunIMAllowContactListManage sunIMAllowUserSettings sunIMAllowPollingAccess sunIMAllowPollingSend	sunIMProperties sunIMRoster sunIMConferenceRoster sunIMNewsRoster sunIMPrivateSettings	sunIMUserProperties sunIMUserRoster sunIMUserConferenceRoster sunIMUserNewsRoster sunIMUserPrivateSettings
sunPresence	sunPresenceAllowAccess sunPresenceAllowPublish sunPresenceAllowManage	sunPresenceDevices sunPresencePrivacy	sunPresenceEntityDevices sunPresenceUserPrivacy

For each attribute in the preceding table, a corresponding label appears in the Access Manager admin console. Table 17-4 lists and describes the policy attributes and Table 17-5 lists and describes the dynamic and user attributes.

Table 17-4 Access Manager Policy Attributes for Instant Messaging

Policy Attribute	Admin Console Label	Attribute Description
sunIMAllowChat	Ability to Chat	End users can be invited to join chat room and access normal chat functionality
sunIMAllowChatInvite	Ability to Invite others to Chat	End users can invite others to chat
sunIMAllowForumAccess	Ability to Join Conference Rooms	A conference tab shows up in Instant Messenger, allowing end users to join conference rooms
sunIMAllowForumManage	Ability to Manage Conference Rooms	End users are able to create, delete, and manage conference rooms
sunIMAllowForumModerate	Ability to Moderate Conference Rooms	End users can be conference moderators
sunIMAllowAlertsAccess	Ability to Receive Alerts	End users can receive alerts from others
sunIMAllowAlertsSend	Ability to Send Alerts	End users can send alerts to others
sunIMAllowNewsAccess	Ability to Read News	A News button is displayed in Instant Messenger that enables end users to list news channels in order to receive and send news messages
sunIMAllowNewsManage	Ability to Manage News Channels	End users can manage news channels and create, delete, and assign privileges to news channels
sunIMAllowFileTransfer	Ability to Exchange Files	End users can add attachments to alert, chat, and news messages
sunIMAllowContactListManage	Ability to Manage one's Contact List	End users can manage their own contact lists; they can add and delete users or groups to and from the list; they can rename the folder in their contact list
sunIMAllowUserSettings	Ability to Manage Messenger	A Settings button is displayed in Instant Messenger that enables end users to change their own Instant Messenger settings
sunIMAllowPollingAccess	Ability to Receive Polls	End users can receive poll messages from others, and they can respond to polls
sunIMAllowPollingSend	Ability to Send Polls	A Poll button is displayed in Instant Messenger that enables end users to send poll messages to others and to receive the responses

sunPresence AllowAccess	Ability to Access other's Presence	End users can watch the presence status of others. The contact list, in addition to showing the contact, reflects contacts' presence status changes by changing the status icon
sunPresence AllowPublish	Ability to Publish Presence	End users can click to select their status (online, offline, busy, etc.) for others to watch
sunPresence AllowManage	Ability to Manage Presence Access	An Access tab is displayed in Instant Messenger settings that allows end users to set up their own default presence access, presence permitted, or presence denied list

Modifying Attributes Directly

An end user can log into the Access Manager admin console and view the values of attributes in the Instant Messaging and Presence service attributes. If the attributes have been defined as modifiable, end users can alter them. By default no attributes in the Instant Messaging service are modifiable, nor is it recommended that end users be allowed to modify them. However, from the standpoint of system administration, manipulating attributes directly can be useful.

For example, since roles do not affect some system attributes, such as setting conference subscriptions, system administrators might want to modify the values of these attributes by copying them from another end user (such as from a conference roster) or modifying them directly. These attributes are listed in [Table 17-5](#).

User attributes can be set by end users through the Access Manager admin console. Dynamic attributes are set by the administrator. A value set for a dynamic attribute overrides or is combined with the corresponding user attribute value.

The nature of corresponding dynamic and user attributes influences how conflicting and complementing information is resolved. For example, Conference Subscriptions from two sources (dynamic and user) complement each other, so the subscriptions are merged. Neither attribute overrides the other.

Table 17-5 Access Manager User and Dynamic Attributes for Instant Messaging

Admin Console Label	User Attribute	Dynamic Attribute	Attribute Description	Conflict Resolution
Messenger Settings	sunIMUser Properties	sunIM Properties	Contains all the properties for Instant Messenger and corresponds to the <code>user.properties</code> file in the file-based user properties storage	Merge. Unless a particular property has a value from both the user and dynamic attribute, then the dynamic attribute overrides.
Subscriptions	sunIM UserRoster	sunIMRoster	Contains subscription information (user contact list roster)	Merge. If a Jabber identifier is present in both the user and dynamic attribute, then the nickname will be taken from the user attribute, the group will be a union of all groups from both user and dynamic attributes, the subscription value will be the highest value from the user and dynamic value.
Conference Subscriptions	sunIMUser ConferenceRoster	sunIMConference Roster	Contains conference room subscription information	Merge. Dynamic and user subscriptions are merged, and duplicates are removed.
News Channel Subscriptions	sunIMUser NewsRoster	sunIM NewsRoster	Contains news channel subscription information	Merge. Dynamic and user subscriptions are merged and duplicates are removed.
Presence Agents	sunPresence EntityDevices	sunPresence Devices	Not used in this release (for future use)	The dynamic information is used.
Privacy	sunPresence UserPrivacy	sunPresence Privacy	Corresponds to the privacy setting in Instant Messenger	Merge. the dynamic value is used if there is a conflict.
Instant Messenger Preferences	sunIMUser PrivateSettings	sunIM PrivateSettings	Store private preferences here that are not stored in Messenger Settings	Merge.

Predefined Instant Messaging and Presence Policies

Table 17-6 lists and describes the seven example policies and roles that are created in Access Manager when the Instant Messaging service component is installed. You can add end users to different roles according to the access control you want to give them.

A typical site might want to assign the role IM Regular User (a role that receives the default Instant Messaging and Presence access) to end users who simply use Instant Messenger, but have no responsibilities in administering Instant Messaging policies. The same site might assign the role of IM Administrator (a role associated with the ability to administer Instant Messaging and Presence services) to particular end users with full responsibilities in administering Instant Messaging policies. [Table 17-7](#) lists the default assignment of privileges amongst the policy attributes. If an action is not selected in a rule, the values `allow` and `deny` are not relevant as the policy then does not affect that attribute.

Table 17-6 Default Policies and Roles for Access Manager

Policy	Role to Which the Policy Applies	Service to Which the Policy Applies	Policy Description
Default Instant Messaging and presence access	IM Regular User	sunIM, sunPresence	The default access that a regular Instant Messaging end user should have.
Ability to administer Instant Messaging and Presence Service	IM Administrator	sunIM, sunPresence	The access that an Instant Messaging Administrator has, which is access to all Instant Messaging features.
Ability to manage Instant Messaging news channels	IM News Administrator	sunIM	End users can manage news channels by creating, deleting, etc.
Ability to manage Instant Messaging conference rooms	IM Conference Rooms Administrator	sunIM	End users can manage conference rooms by creating, deleting, etc.
Ability to change own Instant Messaging user settings	IM Allow User Settings Role	sunIM	End users can edit settings modifying values in the Settings dialog box in Instant Messenger.
Ability to send Instant Messaging alerts	IM Allow Send Alerts Role	sunIM	End users can send alerts in Instant Messenger.
Ability to watch changes on other Instant Messaging end users	IM Allow Watch Changes Role	sunIM	End users can access the presence status of other Instant Messaging end users.

Table 17-7 Default Policy Assignments

	Policy						
Attribute	Default access	Can administer Instant Messaging and Presence Service	Can manage news channels	Can manage conference rooms	Can change own end-user settings	Can send alerts	Can watch changes to other users
sunIM AllowChat	allow	allow					

sunIM AllowChat Invite	allow	allow					
sunIM AllowForum Access	allow	allow		allow			
sunIM AllowForum Manage	deny	allow		allow			
sunIM AllowForum Moderate	deny	allow		allow			
sunIM AllowAlerts Access	allow	allow				allow	
sunIM AllowAlerts Send	allow	allow				allow	
sunIM AllowNews Access	allow	allow	allow				
sunIM AllowNews Manage	deny	allow	allow				
sunIM AllowFile Transfer	allow	allow					
sunIM AllowContact ListManage	allow	allow					
sunIM AllowUser Settings	allow	allow			allow		
sunIM AllowPolling Access	allow	allow					
sunIM AllowPolling Send	allow	allow					
sunPresence AllowManage	allow	allow					
sunPresence AllowAccess	allow	allow					allow
sunPresence AllowPublish	allow	allow					

Creating New Instant Messaging Policies

You can create new policies to fit the specific needs of your site.

To Create a New Policy

1. Log in to the Access Manager admin console at `http://<hostname>:<port>/<amconsole>`.
For example: <http://imserver.company22.example.com:80/amconsole>
2. Select the Identity Management tab.
3. Select Policies in the View drop down list in the navigation pane (the lower-left frame).
4. Click New.
The New Policy page appears in the data pane (the lower-right frame).
5. Select Normal for the Type of Policy.
6. Enter a policy description in the Name field.
For example: `Ability to Perform IM Task`.
7. Click Create.
Access Manager admin console displays the name of the new policy in the policy list in the navigation pane and brings up the Edit page for your new policy.
8. On the Edit page, select Rules in the View drop down list.
The Rule Name Service Resource panel appears inside the Edit page.
9. Click Add.
The Add Rule page appears.
10. Select the Service that applies.
You can select either Instant Messaging Service or Presence Service.
Each service enables you to allow or deny end users the ability to perform specific actions. For example, Ability to Chat is an action specific to the Instant Messaging service while Ability to Access other's Presence is an action specific to the Presence service.
11. Enter a description for a rule in the Rule Name field.
For example: `Rule 1`
12. Enter the appropriate Resource Name.
Enter either:
`IMResource` for Instant Messaging Service
or
`PresenceResource` for Presence Service
13. Select the Actions that you want to apply.
14. Select the Value for each action.
You can select either Allow or Deny.
15. Click Create.
The proposed rule is displayed in the list of saved rules for that policy.
16. Click Save.
The proposed rule becomes a saved rule.
17. Repeat steps 9-16 for any additional rules that you want to apply to that policy.

Assigning Policies to a Role, Group, Organization, or User

You can assign policies to a role, group, organization, or user. This includes the default policies or policies that were created after Instant Messaging was installed.

To Assign a Policy

1. Log in to the Access Manager admin console at `http://<hostname>:<port>/<amconsole>`.
For example: <http://imserver.company22.example.com:80/amconsole>
2. Select the Identity Management tab.
3. Select Policies in the View drop down list in the navigation pane (the lower-left frame).
4. Click the arrow next to the name of the policy you want to assign.
The Edit page for that policy appears in the data pane (the lower-right frame).
5. On the Edit page, select Subjects in the View drop down list.
6. Click Add.

The Add Subject page appears, which lists the possible subject types.

Access Manager Roles

LDAP Groups

LDAP Roles

LDAP Users

Organization

7. Select the subject type that matches the policy.
For example, Organization.
8. Click Next.
9. In the Name field, enter a description of the subject.
10. (Optional) Select the Exclusive check box.
The Exclusive check box is not selected as the default setting, which means that the policy applies to all members of the subject.
Selecting the Exclusive check box applies the policy to everyone who is not a member of the subject.
11. In the Available field, search for entries that you want to add to your subject.
 - a. Type a search for the entries you want to search for.
The default search is (*), which displays all the subjects for that subject type.
 - b. Click search.
 - c. Highlight entries in the Available text box that you want to add to the Selected text box.
 - d. Click Add or Add All, whichever applies.
 - e. Repeat steps a-d until you have added all the names you want to the Selected text box.
12. Click Create.
The proposed subject appears in the list of proposed subjects for that policy.
13. Click Save.
The proposed subject becomes a saved subject.
14. Repeat steps 6-13 for any additional subjects that you want to add to the policy.

Creating New Suborganizations Using Access Manager

The ability to create suborganizations using Access Manager enables organizationally separate populations to be created within the Instant Messaging server. Each suborganization can be mapped to a different DNS domain. End users in one suborganization are completely isolated from those in another. The following procedure describes minimal steps to create a new suborganization for Instant Messaging.

To Create a New Suborganization

1. Log in to the Access Manager admin console at <http://<hostname>:<port>/<amconsole>>
For example: <http://imserver.company22.example.com:80/amconsole>
2. Select the Identity Management tab.
3. Create a new organization.
 - a. Select Organizations in the View drop down list in the navigation pane (the lower-left frame).
 - b. Click New.
The New Organization page appears in the data pane (the lower-right frame).
 - c. Enter a suborganization name.
For example: `sub1`
 - d. Enter a domain name.
For example: `sub1.company22.example.com`
 - e. Click Create.
4. Register services for the newly created suborganization:
 - a. Click the name for the new suborganization in the navigation pane.
For example, click `sub1`. Ensure that you click the name, not the property arrow at the right.
 - b. Select Services from the View drop down list in the navigation pane.
 - c. Click Register.
The Register Services page appears in the data pane.
 - d. Select the following services under the Authentication heading
Core
LDAP

- e. Select the following services under the Instant Messaging Configuration heading:
Instant Messaging Service
Presence Service
- f. Click Register.
 The newly selected services for this suborganization appear in the navigation pane.
- 5. Create service templates for the newly selected services.
 - a. In the navigation pane, click the property arrow for a service, starting with the Core service.
 The Create Service Template page appears in the data pane.
 - b. In the data pane, click Create.
 A page displaying a list of template options for the service you have selected appears.
 You should click Create for each service even when you do not want to modify the template options.
 - c. Modify the options for the service template of each service as follows
Core : Generally, no options need to be modified.
LDAP : Add the prefix of the new suborganization to the DN to Start User Search field.
 After adding the prefix, the final DN should be in this format:
`o=sub1,dc=company22,dc=example,dc=com`
 Enter the LDAP password in the Password for Root User Bind and Password for Root User Bind (confirm) fields.
Instant Messaging Service : Generally, no options need to be modified.
 - d. Click Save.
 - e. Repeat steps a-d until you have created service templates for each service.

Assigning Roles to End Users in New Suborganizations

After new end users have been created in a suborganization they need to be assigned roles. Roles can be inherited from the parent organization.

To Assign Roles to End Users in a New Suborganization

1. Log in to the Access Manager admin console at `http://<hostname>:<port>/<amconsole>`
 For example: <http://imserver.company22.example.com:80/amconsole>
2. Select the Identity Management tab.
3. Select Roles in the View drop down list in the navigation pane (the lower-left frame).
4. Click on the property arrow to the right of the role you wish to assign.
 A page for that role appears in the data pane (the lower-right frame).
5. Select Users from the View drop down list in the data pane.
6. Click Add.
 The Add Users page appears.
7. Enter a matching pattern to identify users.
 For example, in the `UserId` field an asterisk, (*), lists all users.
8. Click Filter.
 The Select User page appears.
9. On the Select User page, check the Show Parentage Path check box and click Refresh.
 The parentage path is displayed.
10. Select the users to be assigned to this role.
11. Click Submit.

Chapter 28. Managing Instant Messenger

Managing Instant Messenger

This information describes how to customize and administer Instant Messenger.

Topics:

- [Configuring Instant Messenger](#)
- [Invoking Instant Messenger](#)
- [Changing the Codebase](#)
- [Changing the Web Container Port](#)
- [Customizing Instant Messenger](#)
- [Modifying How Client Users Search for Contacts](#)
- [Administering Conference Rooms and News Channels](#)
- [Modifying Instant Messenger Proxy Settings](#)
- [Controlling the Exposed Messenger Feature Set](#)
- [Instant Messenger Data Stored in the End User's System](#)
- [Redeploying Resource Files](#)

Configuring Instant Messenger

There are two ways to invoke and run Instant Messenger:

Using Java Web Start - In this configuration, Instant Messenger is launched as an application from the Java Web Start. The browser is no longer necessary once Instant Messenger is launched.

Using the Java Plug-in - In this configuration, Instant Messenger is run as a Java applet. To keep the Instant Messenger session active, the browser window from which the applet was launched must remain open and cannot be used to locate any other URL. In addition, the Java plug-in does not allow desktop integration so the Desktop Integration Settings option will not be available from the Settings dialog box.

For more information on how to configure the Java software that enables Instant Messenger, see [Setting up and Launching Instant Messenger](#).

Invoking Instant Messenger

You can invoke Instant Messenger from several locations:

- The `index.html` file that provides you the options to launch both the Java Web Start and Java Plug-in versions of Instant Messenger. This file also contains links to Instant Messenger documentation.
- A web page you have designed with a link to Instant Messenger.
- A direct URL for either the `im.html` or `im.jnlp` files.
- From the command-line.
- Using a desktop shortcut.

Invoking Instant Messenger is described in the following sections:

- [To Invoke Instant Messenger By Using a Direct URL](#)
- [To Invoke Instant Messenger From the Command-Line \(Solaris OS Only\)](#)
- [To Invoke Instant Messenger By Using a Desktop Shortcut](#)

To Invoke Instant Messenger By Using a Direct URL

1. Type the following URL in your web browser to invoke Instant Messenger:

```
http://<webserver>:<webserverport>/<path>/<filename>
```

In this URL:

<i>webserver</i>	Specifies the name of the web container on which you have installed the Instant Messenger resources.
<i>webserverport</i>	(Optional) Specifies the web container port. The default value is 80.
<i>path</i>	(Optional) Specifies the directory where the client files are installed. If the default is selected during the installation, then no subdirectory is required to store the client files.
<i>filename</i>	Specifies the Instant Messenger file to use: <i>index.html</i> - This file is provided with the product. The file contains links to <i>im.jnlp</i> and <i>im.html</i> which launch the Java Web Start and Java Plug-in versions of Instant Messenger respectively. <i>im.jnlp</i> - The <i>.jnlp</i> file to launch only the Java Web Start version of Instant Messenger. <i>im.html</i> - The web page to launch only the Java Plug-in version of Instant Messenger.

To Invoke Instant Messenger From the Command-Line (Solaris OS Only)

1. Type the following at the command-line:

```
javaws_cmd <URL>
```

See [To Invoke Instant Messenger By Using a Direct URL](#) for information about constructing the URL.

To Invoke Instant Messenger By Using a Desktop Shortcut

1. Create and use a desktop shortcut to invoke Instant Messenger.
 - Create a shortcut using Java Web Start.
 - Create a shortcut manually and set the target value as follows:

```
javaws_cmd <jnlp-URL>
```

where *jnlp-URL* is the URL to the *im.jnlp* file.

Changing the Codebase

The *codebase* is the URL from which Instant Messenger accesses resources, including the start page for initial downloads of the Instant Messaging client. This URL is defined during postinstallation configuration when the resource files are deployed by the configure utility. If you change any portion of the URL used to access Instant Messenger resources including the web container port number you need to update the codebase.

To change the codebase after you have deployed the resource files, you need to:

- Modify the template files to point to the new URL. See [To Change the Codebase in the Resource Templates](#).
- Run the `configure` utility, selecting the "Messenger Resources" component only when prompted for which components you want to configure. See [Configuring Instant Messaging After Installing or Upgrading](#) for instructions.
- Redeploy the resource files. See [Redeploying Resource Files](#) for instructions.

To Change the Codebase in the Resource Templates

- Edit each of the template files in the `im-svr-base/html` directory with the new URL. Template files are named `*.template`. See [Instant Messenger Resource Files](#) for a complete list of template files.

Changing the Web Container Port

If you change any portion of the URL used to access Instant Messenger resources including the web container port number you need to update the codebase. See [Changing the Codebase](#) for instructions.

Customizing Instant Messenger

Instant Messenger is customizable. HTML and JNLP files can be customized to suit an organization's specific needs. If you want to customize the resource files for your deployment, you should run the `configure` utility (if you haven't already done so after installing), customize the files, then redeploy the resource files. You need to run the `configure` utility first because `configure` creates some of the files that you can customize. (See [Redeploying Resource Files](#) for redeployment instructions.)

You can customize Instant Messenger to meet your requirements in the following ways:

- [Instant Messenger Resource Files](#)
- [Customizing the `index.html` and `im.html` Files](#)
- [Launching Instant Messenger Using Access Manager SSO](#)
- [Customizing the Application \(Java Web Start\)](#)
- [Rebranding Instant Messenger](#)
- [Customizing User Name and Group Name Display](#)

This section describes the Instant Messaging server files you can modify to customize Instant Messenger. The files that you can customize are all located in the resource directory `im-svr-base/html` directory. See [Table 3-1](#) for information on default directory locations.

Instant Messenger Resource Files

The Instant Messenger resource files are located within a directory referred to as the resource directory or `im-svr-base/html`.

[Table 15-1](#) contains the list of Instant Messenger files in the resource directory (`im-svr-base/html`). It also contains the description and customization information for these files. Within the resource directory, the `/locale` subdirectory is represented generically in a directory path as `lang`, but specifically as abbreviations of languages, such as `en_US`, `jp`, and `fr_FR`.

Table 15-1 Instant Messenger Resource Files in `im-svr-base/html`

File	Description	Customizable?
------	-------------	---------------

lang/im.html	The initial page that launches the Java Plug-in version of Instant Messenger.	Yes
im.html.template	The template version of im.html.	No, This file is used by the installation program to generate the im.html file.
imdesktop.jar	A client .jar file, downloaded by im.html or im.jnlp files.	No
lang/im.jnlp	The .jnlp file used to launch Java Web Start version of Instant Messenger.	Yes
im.jnlp.template	The template version of im.jnlp.	No
imjni.jar	A client .jar file, downloaded by im.html or im.jnlp.	No
messenger.jar	The main client .jar file, downloaded by im.html or im.jnlp.	No
icalendar.jar	The icalendar parser used to process calendar reminders.	No
imnet.jar	A client .jar file, downloaded by im.html or im.jnlp.	No
lang/imbrand.jar	This file contains customizable properties, stylesheets, images, backgrounds, and audio files.	Yes
lang/imssl.html	The Initial page that launches Java Plug-in version of Instant Messenger. It is used for running legacy SSL between the client and the multiplexor. Do not use this file for secure communication between the client and server over TLS.	Yes
lang/imssl.jnlp	This file launches Java Web Start version of Instant Messenger. This file is used for running SSL between the client and the multiplexor.	Yes
jnlpLaunch.jsp	If an end user is already logged into Access Manager, this file can be used to allow single sign-on and to launch Instant Messenger using Java Web Start.	Yes
pluginLaunch.jsp	If an end user is already logged into Access Manager, this file can be used to allow single sign-on and to launch Instant Messenger using Java Plug-in.	Yes
index.html	The splash page for an LDAP deployment. It contains links to im.html and im.jnlp, as well as documentation links to windows.htm, solaris.htm, and quickref.htm. You can customize this page for your site's requirements.	Yes
index.html.template	The template version of index.html.	No
lang/imhelp/SunONE.jpg	The image used by quickref.htm, solaris.htm, and windows.htm.	Can be replaced, but not modified.

quickref.html solaris.html windows.html	Located in lang/imhelp/, these files provide documentation on getting started with Instant Messenger.	Yes
lang/imhelp	Instant Messenger Online Help directory.	No
imwebex.jar		
msginstall.jar		

Customizing the index.html and im.html Files

If you are using Instant Messenger in a deployment without Access Manager, you can modify the *static* portion of the `index.html` and `im.html` files to produce a fully customized user interface. These HTML files contain both text and markups describing how the text is formatted and handled. Markup is implemented through a set of tags, which specify formats for headers, indents, font size, and font style.

Some of the page elements that can be modified are:

- Images
- Banner
- Text on screen including title and field labels
- Background schemes

You can launch the Instant Messenger applet and the Java Web Start application from `index.html`. If you are running the Instant Messenger applet, modify the `im.html` file. The `im.html` file is called by `index.html`, and invokes the Instant Messenger applet. The `im.html` file is generated when you run the configure utility and contains an applet argument that points to the multiplexor.



Note

The argument "`<PARAM NAME="server" VALUE="servername">`" represents the Instant Messaging multiplexor and its port in the `im.html` file. If you change the `iim_mux.listenport` parameter's default value, you need to change the `servername` value to `host.domain:port`.

Launching Instant Messenger Using Access Manager SSO

To launch the Instant Messenger client by using single sign-on (SSO) with Access Manager, use `IMLaunch.jsp`. This file is in the resource directory.

Access Manager and Instant Messenger must be configured to use the same web container to enable SSO.

To launch Instant Messenger enter the following in a web browser:

```
codebase/IMLaunch.jsp?server=multiplexor-hostname:multiplexor-port
```

or

```
codebase/IMLaunch.jsp?server=www.example.com:5222
```

Where:

- *codebase* is the codebase from which the Instant Messenger resources are downloaded. For example, [http://www.example.com].
- *multiplexor-hostname* is the host name of the multiplexor. For example, [http://www.company22.com].
- *multiplexor-port* is the port number on which the multiplexor listens for incoming client requests. For example, 5222.
- *IMLaunch.jsp* is used for launching Instant Messenger through either Java Web Start or Java Plug-in.

Customizing the Application (Java Web Start)

If you are running Instant Messenger using Java Web Start, you can modify the `im.jnlp`, `imres.jnlp`, and `imres.jar` files to customize the user interface. The following are modifications that can be made to these files: `imbrand.jar` - This file contains the image and audio files, and the properties that can be customized. You need Java Developers Kit 1.3 (JDK) to extract the contents from the `imres.jar` file using the `jar` command. For more information on `imbrand.jar` contents, see [Contents of imbrand.jar](#).

Use the following command to extract `imbrand.jar`:

```
jar xvf imbrand.jar
```

This command creates a directory tree where the resource files are copied. This directory structure has to be maintained when you modify the individual files in the `.jar` file.

You can substitute your version of `.gif` files or `.wav` files, without changing the file names and then place the changed files back to the directory using the following `jar` command:

```
jar -uf imbrand.jar com/Sun/im/client/images/*.gif
```

This command updates the `imbrand.jar` file with the modified `.gif` files. The same is possible with the audio files (`.wav` files).

- `im.jnlp` - this file invokes the Java Web Start version of the Instant Messenger application. You can modify the codebase, title, vendor, and descriptions in the file. [Example 15-1](#) shows a sample `im.jnlp` file with the HTML code that can be customized in bold typeface.

Example 15-1 Sample `im.jnlp` File

```

<?xml version="1.0" encoding="utf-8"?>
<!-- Instant Messenger -->
<jnlp
  spec="1.0+"
  codebase="http://im.i-zed.com:80/im"
  href="en/im.jnlp">
  <information>
    <title>Instant Messaging</title>
    <vendor>I-Zed.com</vendor>
    <homepage href="http://www.I-zed.com/" />
    <description>I-Zed's Instant Messenger</description>
    <description kind="short">Instant Messenger</description>
    <icon href="CompanyLogo.gif" />
    <offline-allowed/>
  </information>
  <security>
    <all-permissions/>
  </security>
  <resources>
    <j2se version="1.3+">
      <resources>
        <jar href="en/imres.jar" />
        <jar href="en/imbrand.jar" />
      </resources>
    </j2se>
    <jar href="messenger.jar" />
    <jar href="imdesktop.jar" />
    <jar href="imnet.jar" />
    <jar href="icalendar.jar" />
    <nativelib href="imjni.jar" />
  </resources>
  <application-desc main-class="com.iplanet.im.client.iIM">
    <argument>server=im.i-zed.com:45222</argument>
    <argument>help_codebase=http://im.i-zed.com:80/im/en</argument>
  </application-desc>
</jnlp>

```



Note

In the `im.jnlp` file, the argument `<argument>servername</argument>` represents the Instant Messaging multiplexor host and port. If you change the default value of the `iim_mux.listenport` parameter, you need to change the `servername` value to `host.domain:port`.

Contents of `imbrand.jar`

The tables in this section list the files in the `imbrand.jar` file and provide a description of each file wherever possible. The `imbrand.jar` file also contains the image and audio files you can use to re-brand Instant Messenger. This section contains the following tables:

- [Table 15-2](#) - configuration files used to configure Instant Messenger.
- [Table 15-3](#) - emoticons available for use during chat sessions.
- [Table 15-4](#) - icons used by the application on Windows.

- [Table 15-5](#) - icons used by the application on all platforms.
- [Table 15-6](#) - icons used in the toolbar.
- [Table 15-7](#) - icons used in the contact list.
- [Table 15-8](#) - icons used to describe presence information in the contact list.
- [Table 15-9](#) - icons used to describe presence information in the status bar.
- [Table 15-10](#) - available backgrounds.
- [Table 15-11](#) - sounds used to indicate alerts and status or configuration changes.

Table 15-2 Configuration Files

File	Description
brand.properties	
chat-styles.css	
bgstyles.properties	Background configuration file, used to extend the background set

Table 15-3 Emoticons

File Name	Description
emo_alarm.png	Shows alarm emotion graphically
emo_angel.png	Shows angelic emotion graphically
emo_angry.png	Shows angry emotion graphically
emo_balloons.png	Graphic depiction of a bunch of balloons
emo_beermug.png	Graphic depiction of a mug of beer
emo_cake.png	Graphic depiction of a birthday cake
emo_calendar.png	Graphic depiction of a calendar
emo_canworms.png	Graphic depiction of a can of worms
emo_clown.png	Graphic depiction of a clown's head
emo_cool.png	Shows cool emotion graphically
emo_dead.png	Indicates dead graphically
emo_devil.png	Shows devilish emotion graphically
emo_dont-tell.png	Indicates a request for secrecy graphically
emo_embarrassed.png	Shows embarrassed emotion graphically
emo_exclamation.png	Graphic depiction of an exclamation point
emo_flower.png	Graphic depiction of a flower
emo_ghost.png	Graphic depiction of a ghost
emo_goldstar.png	Graphic depiction of a gold star
emo_grin.png	Shows a grin graphically
emo_kiss.png	Shows a kiss graphically
emo_laughing.png	Show laugh emotion graphically
emo_lifepreserver.png	Graphic depiction of a life preserver

emo_lightning.png	Graphic depiction of a thunder cloud and lightning bolt
emo_lovestruck.png	An emoticon used to show love emotion graphically
emo_martini.png	Graphic depiction of a martini glass
emo_money.png	Graphic depiction of stacks of coins
emo_musicnote.png	Graphic depiction of a musical note
emo_nerd.png	Graphic depiction of a nerd
emo_nottalking.png	Shows a turned-away countenance graphically
emo_phone.png	Graphic depiction of a phone receiver
emo_present.png	Graphic depiction of a wrapped gift
emo_psychoknife.png	Graphic depiction of a knife
emo_rathole.png	Graphic depiction of a rat hole
emo_sad.png	Shows sad emotion graphically
emo_sick.png	Shows illness graphically
emo_sleep.png	Shows sleepiness graphically
emo_smiley.png	Shows a smile graphically
emo_straightfaced.png	Graphic depiction of a straight-faced person
emo_sunshining.png	Graphic depiction of a sun
emo_surprised.png	Shows surprise graphically
emo_tongue-out.png	Graphic depiction of a person sticking out his tongue
emo_violin.png	Graphic depiction of a violin
emo_whatever.png	Shows indifference or disdain graphically

Table 15-4 Application Icons - Windows

File Name	Description
im_app_icon_16.png	Title bar icon for Windows
im_app_icon_24.png	Title bar icon for Windows
tray_icon.ico	System tray icon for Windows

Table 15-5 Application Icons - All Platforms

File Name	Description
logo_login_footer.png	Logo displayed at the bottom of the Login dialog box
logo_register.png	Logo displayed on the Register dialog box
logo_sun.png	Sun logo displayed on the Login dialog box

Table 15-6 Toolbar Icons

File Name	Description
tb_addcontacts.png	Graphic for the Add Contacts button
tb_alert.png	Graphic for the Send Alert button
tb_chat.png	Graphic for the Chat With Users button
tb_conf.png	Graphic for the Add Conferences button

Table 15-7 Contact List Icons

File Name	Description
cl_folder_closed.png	Shows a closed folder graphically
cl_folder_open.png	Shows an open folder graphically

Table 15-8 Presence Icons - Contact List

File Name	Description
cl_activeconf.png	Icon displayed to indicate an active conference that appears in the Contact List
cl_away.png	Icon for away status that appears in the Contact List
cl_dnd.png	
cl_idle.png	Icon displayed to show idle status that appears in the Contact List
cl_inactiveconf.png	Icon displayed to indicate an inactive conference that appears in the Contact List
cl_offline.png	Icon for offline status that appears in the Contact List
cl_online.png	Icon for online status that appears in the Contact List
cl_pending.png	Icon that indicates pending status that appears in the Contact List

Table 15-9 Presence Icons - Status Bar

File Name	Description
sb_away.png	Icon for away status that appears in the Status Bar
sb_dnd.png	
sb_idle.png	Icon for idle status that appears in the Status Bar
sb_offline.png	Icon for offline status that appears in the Status Bar
sb_online.png	Icon for online status that appears in the Status Bar

Table 15-10 Backgrounds and Background Swatches for the Palette

bgplt_tex_blue.gif {bgplt_tex_brown.gif bgplt_tex_bubble_blue.gif bgplt_tex_bubble_brown.gif bgplt_tex_bubble_green.gif bgplt_tex_bubble_grey.gif bgplt_tex_bubble_orange.gif bgplt_tex_bubble_purple.gif bgplt_tex_bubble_ruby.gif bgplt_tex_crackle_blue.gif bgplt_tex_crackle_green1.gif bgplt_tex_crackle_grey.gif bgplt_tex_crackle_olive.gif bgplt_tex_crackle_orange.gif bgplt_tex_crackle_purple.gif bgplt_tex_crackle_ruby.gif bgplt_tex_gradation_blue.gif bgplt_tex_gradation_brown.gif bgplt_tex_gradation_green.gif bgplt_tex_gradation_grey.gif bgplt_tex_gradation_orange.gif bgplt_tex_gradation_purple.gif bgplt_tex_gradation_ruby.gif bgplt_tex_green.gif bgplt_tex_orange.gif bgplt_tex_pink.gif bgplt_tex_purple.gif bgplt_tex_weave_blue.gif bgplt_tex_weave_brown.gif bgplt_tex_weave_green.gif bgplt_tex_weave_grey.gif bgplt_tex_weave_orange.gif	bgplt_tex_weave_purple.gif bgplt_tex_weave_ruby.gif bgplt_tex_white.gif bg_tex_bubble_blue.gif bg_tex_bubble_brown.gif bg_tex_bubble_green.gif bg_tex_bubble_grey.gif bg_tex_bubble_orange.gif bg_tex_bubble_purple.gif bg_tex_bubble_ruby.gif bg_tex_crackle_blue.gif bg_tex_crackle_green1.gif bg_tex_crackle_grey.gif bg_tex_crackle_olive.gif bg_tex_crackle_orange.gif bg_tex_crackle_purple.gif bg_tex_crackle_ruby.gif bg_tex_gradation_blue.gif bg_tex_gradation_brown.gif bg_tex_gradation_green.gif bg_tex_gradation_grey.gif bg_tex_gradation_orange.gif bg_tex_gradation_purple.gif bg_tex_gradation_ruby.gif bg_tex_weave_blue.gif bg_tex_weave_brown.gif bg_tex_weave_green.gif bg_tex_weave_grey.gif bg_tex_weave_orange.gif bg_tex_weave_purple.gif bg_tex_weave_ruby.gif
---	---

Table 15-11 Sounds

File Name	Description
alert.wav	Alert sound
alerttpc.wav	Alert sound
away.wav	Sound used when you change your status to away
receive.wav	Sound used when you receive a message
send.wav	Sound used when you send a message
soundoff.wav	Sound used when you turn the sound off
soundon.wav	Sound used when you turn the sound on

Rebranding Instant Messenger

The `imbrand.jar` file contains all images and the properties that control the look and feel of Instant Messenger. You can customize the appearance of Instant Messenger by modifying the images and the

properties in `imbrand.jar`.

To Rebrand Instant Messenger

1. Copy `imbrand.jar` file to a working directory.

For example:

```
cp im-svr-base/html/lang/imbrand.jar working-directory
```

2. Change to the working directory.

```
cd working-directory
```

3. Extract the `imbrand.jar` file.

```
jar xf imbrand.jar
```

This command creates a directory tree where the resource files are copied. This directory structure has to be maintained when you modify the individual files in the `imbrand.jar` file. Alternatively, you can extract a single file included in `imbrand.jar` and put it under the directory structure you specify. For example, to extract only `brand.properties`, use the following command:

```
jar xf imbrand.jar com/sun/im/desktop/brand/brand.properties
```

4. Update `imbrand.jar` with the modified `.gif`, `.wav`, and `.properties` files.

You can update all the files in `imbrand.jar` as follows:

```
jar cf imbrand.jar .
```

To update `imbrand.jar` with a single modified file, use the following command:

```
jar uf imbrand.jar com/sun/im/desktop/brand/filename
```

Where *filename* is the name of the file included in `imbrand.jar`, for example, `brand.properties`.

5. Copy `imbrand.jar` to the resource directory.

For example:

```
cp imbrand.jar im-svr-base/html/lang/ .
```



Note

If you support multiple locales in your deployment, follow the procedure for rebranding Instant Messenger for every supported locale.

Customizing User Name and Group Name Display

You can customize how Instant Messenger displays contact and group names by changing the attribute used to display contact names. By default, the Instant Messenger uses the attribute `cn` to represent a user's display name. In your deployment, you may prefer to use `uid` or some other attribute instead of `cn`.

Contact names appear as *First Name, Last Name*. For example, `Frank Smith, Mary Jones`, and so on. When two end users have the same first name and last name, it is impossible to know which end user has to be added to the contact list. You can customize Instant Messenger to display more information in the search results for the user search, and to display additional information in the Contact tooltip to help distinguish between contacts. For example, you can display the phone number of the Contact when the mouse is placed over the Contact.

To Change the Attribute Used to Display a User's Name

1. Open `iim.conf`.

See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.

2. Specify the attribute you want to use to display usernames as the value for `iim_ldap.userdisplay`.

For example, to use the `nickname` attribute, set the `iim_ldap.userdisplay` attribute as follows:

```
iim_ldap.userdisplay=nickname
```

3. Save and close the file.

To Change the Attribute Used to Display a Group's Name

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Specify the attribute you want to use to display usernames as the value for `iim_ldap.groupdisplay`.
For example, to use the `uid` attribute, set the `iim_ldap.groupdisplay` attribute as follows:
`iim_ldap.groupdisplay=uid`
3. Save and close the file.

To Customize User Name Display in Search Results

1. Extract the files from `imbrand.jar`.
See [Table 15-1](#) for default locations for `imbrand.jar`
2. Change to the following directory:
`com/sun/im/client/`
3. Open `brand.properties`.
4. Add the `dialogs.searchresults.format` attribute to the file.
5. Add the attributes you want to include in search results in the following format:

```
{attr:attribute-name}
```

Where *attribute-name* is the name of the LDAP attribute.
For example, to include the *title* attribute, add the following line:

```
dialogs.searchresults.format=({attr:title})
```

6. Save your changes and close the file.
7. Repackage `imbrand.jar`.
8. Add the user attributes to `iim.conf`.
Specify the attributes as values for the `iim_ldap.userattributes` parameter. Separate multiple attributes with a comma, for example:
`iim_ldap.userattributes=title,department,telephonenumber`
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.

To Customize Tooltip Contents

1. Extract the files from `imbrand.jar`.
See [Table 15-1](#) for default locations for `imbrand.jar`
2. Change to the following directory:
`com/sun/im/client/`
3. Open `brand.properties`.
4. Add the `contact.tooltip.format.html` attribute to the file.
5. Specify the attribute you want to display in the tooltip as the value for `contact.tooltip.format.html`.
For example, if you want to display the telephone number and email address of the contact, you would enter:

```
contact.tooltip.format.html=mailto:{attr:mail}  
tel:{attr:telephonenumber}
```

For more information on customizing the contents of `imbrand.jar` file, see [Customizing the Application \(Java Web Start\)](#).

6. Save your changes and close the file.

7. Repackage `imbrand.jar`.

Modifying How Client Users Search for Contacts

By default the `commonname` or `cn` LDAP attribute is used as a search attribute for users. You can configure Instant Messaging to allow users to search on additional attributes. In addition, if your directory is indexed to allow the use of wildcards, you can configure the Instant Messaging server to allow wildcards in searches for contact names.

To Allow Users to Search on Custom Attributes

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Modify the `iim_ldap.usergroupbynamefilter` attribute.
This parameter specifies the LDAP search string used when searching for users or groups. Provide the attribute value in standard LDAP filter syntax. You can modify it to allow more complex searches. See your Directory Server documentation for more information on modifying search strings.
3. Save and close the file.

To Allow Wildcards in Searches

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Set the `iim_ldap.allowwildcardinuid` parameter to `True`.
This parameter determines if the use of wildcards should be enabled for User IDs while doing a search. Most directory installations have User IDs indexed for exact searches only, so the default value is `False`.
3. Ensure that User IDs are indexed for substring search in your directory.
Setting the `iim_ldap.allowwildcardinuid` parameter to `True` can impact performance unless User IDs are indexed for substring search in your directory. See your directory server documentation for instructions on indexing.

Administering Conference Rooms and News Channels

The administrator can create conference rooms and news channels for end users. However, with the proper privileges, end users can do this also. For more information about adding policies to give end users access to create conference rooms and news channels, see [17 Managing Instant Messaging and Presence Policies](#). End users who create a conference room or a news channel by default have Manage access, enabling them to administer the conference room or the news channel.

Listed below are tasks that you can perform in Instant Messenger to administer the conference rooms and the news channels. For more information on performing these tasks, see the Online Help.

- Administering conference rooms
- Administering and managing news channels
- Assigning conference room access levels to end users
- Assigning news channel access levels to end users
- Assigning end users to conference rooms
- Assigning end users to news channels (subscribing)
- Creating new conference rooms
- Creating new news channels
- Configuring end user settings
- Deleting conference rooms
- Deleting messages from news channels
- Deleting news channels

- Posting messages in news channels
- Removing end users from conference rooms
- Removing end users from news channels

Modifying Instant Messenger Proxy Settings

Instant Messaging messages can contain embedded URLs. For example, <http://stocks.yahoo.com?id=sunw>. If you are using proxy servers, you need to resolve such embedded URLs by modifying the Instant Messenger proxy settings in the Java Web Start configuration.

This is likely to happen if your organization has a firewall, and you need to go through the proxy server before connecting your client hosts to internet, and if Java Web Start has not been configured with the right proxy settings.

Java Web Start can automatically configure the proxy settings by querying the system or the default browser. However, it is not possible for the Java Web Start to automatically configure these settings if the proxy settings are configured using a JavaScript file.

To Set Proxy Settings Manually for a Single Instant Messenger Client Using Java Web Start

Completing this procedure saves proxy preferences in the user's `messenger.properties` file. If you also configure the `im.jnlp` file to use a proxy, and the proxy differs from that in the user's preferences, the user's preferences are used.

1. Invoke Java Web Start.
2. From the File menu, choose Preferences.
3. Select Manual option in the Preferences dialog.
4. Enter the following details:
 - HTTP Proxy - Enter the Name or the IP address of the proxy server.
 - HTTP Port - Enter the port number of the proxy server.
 - No Proxy_Hosts - Enter the name of any domain that you can connect directly, bypassing the proxy server. Use commas to separate multiple host names.
5. Click OK to save the proxy settings.

To Configure Proxy Settings for all Instant Messaging Client Connections in `im.jnlp`

If the proxy you set in `im.jnlp` differs from that in the user's preferences file (`/usr_home/.sunmsgr/messenger.properties`), the user's preferences are used.

1. Open the `im.jnlp` resource file in a text editor.
2. Specify the proxy server by adding the following argument:

```
<argument>proxy=_proxy-host:proxy-port_</argument>
```

Where *proxy-host* is the fully-qualified domain name of the proxy server and *proxy-port* is the port number on which the proxy server listens for incoming requests. For example, `myproxy.siroe.com:8080`.

3. Specify the proxy type by adding the following argument:

```
<argument>proxy_type=_type_
```

Where *type* can be one of `http`, `https`, or `socks`.

Controlling the Exposed Messenger Feature Set

You can control the exposed feature set of Instant Messenger by configuring the Instant Messaging applet parameters in the applet descriptor files.

Table 15-12 shows the Instant Messenger applet parameters in the applet descriptor files. It also contains the description and the default values of these parameters.

Table 15-12 Instant Messenger Applet Parameters

Parameter	Default Value	Description
<i>server</i>	127.0.0.1	The Instant Messaging server host and port.
<i>debug</i>	FALSE	If this parameter is set to true, the applet records all the task performed on java console.
<i>uid</i>		This parameter is used for SSO.
<i>token</i>		This parameter contains the SSO token and is used for auto-login.
<i>secure</i>	FALSE	Indicates to the Instant Messenger that it is run in SRA mode. It displays a security indicator.
<i>usesssl</i>	FALSE	Tells Instant Messenger to use legacy SSL when connecting to multiplexor.
<i>allow_alert_only</i>	FALSE	Tells Instant Messenger to let end user display neither the contact list nor the news channel. This parameter is used in CHAT and POPUP flavors.
<i>allow_attachments</i>	TRUE	Allows file attachment and transfer.
<i>enable_moderator</i>	TRUE	If set to true, enables the moderated conference feature.
<i>messenger_bean</i>		This parameter contains a list of messenger beans to be used. You can enter multiple factory class names with each separated by a comma.
<i>domain</i>	null	This parameter is used in multidomain Access Manager deployments. The value of this parameter should be the logical domain name of the organization in which this end user is present.
<i>gateway_url</i>	null	This parameter contains the URL of the gateway component of portal SRA.

Instant Messenger Data Stored in the End User's System

Instant Messenger caches a limited amount of information on the end user's system for auto-login. This information is located at:

```
home-directory/.sunmsgr
```

home-directory is the end user's home directory. The home directory of the end user can be obtained from the *user.home* parameter in the Java system property.

Table 15-13 shows the directories and files containing the cached data. It also contains the description of the files and the directories.

Table 15-13 Cached Data Directory and Files

File/Directory Name	Type	Description
.sunmsgr/messenger.properties	file	The file containing the auto-logon properties
.sunmsgr/user-domain	directory	Directory containing data specific to a particular log-in name, domain name combination.
home-directory/.sunmsgr/user-domain/messenger.properties	file	This file contains auto-logon options specific to particular <i>user-domain</i> . This file is not used.
home-directory/.sunmsgr/user-domain/messages/	directory	This directory contains cached messages. This directory is not used.

Table 15-14 shows the auto-logon properties for Instant Messaging. It also contains the description and the default values of these properties.

Table 15-14 Auto-logon Properties

Parameter	Default Value	Description
<i>client.password.encoded</i>	false	Determines whether or not the user password is encoded (for use with SSO). If the value for this parameter is <code>true</code> , the encoded password is stored as the value for the <i>net.password</i> parameter.
<i>net.nms</i>	127.0.0.1	Instant Messaging server host name and port.
<i>net.nmsn</i> (Where the trailing <i>n</i> is a digit used to distinguish one entry from another)		The secondary servers' host names and port numbers.
<i>net.user</i>		The default user id.
<i>net.password</i>		The encoded user password that enables auto-logon.

Redeploying Resource Files

If you are using Application Server or Web Server, and you make changes to the resource files after you run the `configure` utility as a result of site changes or customization, you need to redeploy the files to the web container. You may also need to redeploy the resource files after upgrading Instant Messaging.

To Redeploy Resource Files to Application Server or Web Server

1. Run the `iwadmin` command.

```
im-svr-base/html/iwadmin
```

Where *im-svr-base* is the directory in which you installed Instant Messaging.

Running `iwadmin` updates the Instant Messenger `.jar` files. However, `iwadmin` does not update or reinitialize the Instant Messenger download page. See the documentation for your web container for additional information. Also see the `iwadmin man` page for additional configuration options.

2. (Optional) After upgrading, if you want to reinitialize the Instant Messenger download page, run the `configure` utility again.

Reinitializing the download page overwrites any customizations you have made. If you choose not to reinitialize the download page, be aware that the product version on the download page and the product version in the Instant Messenger `.jar` files may differ.

See [Configuring Instant Messaging After Installation](#) for more information.

Chapter 29. Managing Logging for Instant Messaging

Managing Logging for Oracle Communications Instant Messaging Server

Instant Messaging creates log files that record events, related status of various software components, system errors, and other aspects of the server, multiplexor, Calendar agent, watchdog, and Instant Messenger. By examining the log files, you can monitor many aspects of the server's operation. This section provides information about logging in the following topics:

- [Instant Messaging Logging Overview](#)
- [Instant Messaging Log File Location](#)
- [Instant Messaging Component Logging Levels](#)
- [Managing Instant Messaging Logging Using Log4j](#)
- [Configuring Logging for Instant Messaging Components Using `iim.conf` Parameters](#)
- [Administering Logging for Instant Messenger](#)

For information on logging for the XMPP/HTTP Gateway, see [Managing Logging for the XMPP/HTTP Gateway](#). In addition, you can collect logging data for Instant Messenger on demand. See [Administering Logging for Instant Messenger](#) for more information.

Instant Messaging Logging Overview

Instant Messaging provides two ways to generate log files; using log4j, or without log4j by specifying parameters in `iim.conf`. Log4j style logging is available for all server instances including redirect servers, Calendar agent, watchdog, and the XMPP/HTTP Gateway, but not the multiplexor.

For information on logging for the XMPP/HTTP Gateway, see [Managing Logging for the XMPP/HTTP Gateway](#). For information on setting up logging for Instant Messenger see [Administering Logging for Instant Messenger](#).



Note

The `iim.conf` parameter-based logging mechanism may be deprecated in a future release. Use log4j wherever possible.

You can configure the level of logging for the Instant Messaging server, multiplexor, Calendar agent, watchdog, and XMPP/HTTP Gateway. In addition, using log4j, you can configure Instant Messaging to generate a separate log file for XMPP traffic only.

If you are not using log4j style logging, as part of regular system maintenance, you need to periodically review and trim the log files from occupying more disk space. The server does not perform this action.

For more information about log4j, see the <http://logging.apache.org>.

Instant Messaging Log File Location

You specify the location of the log files when you run the `configure` utility after installing Instant Messaging. Typically, log files are stored in `iim-runtime-base/log`. See [Instant Messaging Server](#)

[Directory Structure](#) for information on locating *im-runtime-base*.

If you are using log4j for log file generation in your deployment, the logger will also use the directory you specify during configuration as the base directory in which to store log4j logs.

Instant Messaging Component Logging Levels

The level or priority of maintaining the error log defines how detailed, or verbose, the log should be. A higher priority level implies less details as only events of high priority (high severity) are recorded in the log file. In contrast a lower priority level implies greater details as more events are recorded in the log file.

Regardless of whether you are using log4j or parameter-based logging, you can set the logging level separately for each component.

[Table 13-1](#) describes the logging levels for the components. These logging levels are a subset of the levels defined by the UNIX `syslog` facility.

Table 13-1 Logging Levels for Instant Messaging Components

Level	Description
FATAL	This priority level records minimum logging details in the log file. A log record is added to the log file whenever a severe problem or critical condition occurs. If a FATAL problem occurs, the application might stop functioning.
ERROR	A log record is added to the log file whenever a recoverable software error condition occurs or a network failure is detected. For example, when the server fails to connect to a client or to another server.
WARNING	A log record is added to the log file whenever a user error is detected. For example, when the server cannot understand the communication sent by the client.
INFO	A log record is added to the log file whenever a significant action takes place. For example, when an end user successfully logs in or logs out.
DEBUG	The tasks are recorded in the log file. This information is useful for debugging purposes only. Each event with individual steps within each process or task are written to the log file, to help the end user identify the problems while debugging the application.

When you select a particular logging level, events corresponding to that level and to all higher and less verbose levels are logged.

INFO is the default level for the server. ERROR is the default level for the multiplexor, Calendar agent, and watchdog log files.



Note

If you are not using log4j, and you specify `DEBUG` as the logging level, your log files will occupy more disk space. Monitor and trim your log files to prevent them from occupying more disk space.

Managing Instant Messaging Logging Using Log4j

When you install Instant Messaging, a template file (`log4j.conf.template`) for the log4j configuration file is installed into the *im-svr-base/lib* directory. When you run the `configure` utility after installation,

the template is used to create the log4j configuration file (`log4j.conf`) in the *im-cfg-base* directory. This configuration file is used to determine where to store log files generated by log4j, the logging level to use for various components, the output syntax, and to determine what log files to generate.

This section describes using the log4j logger to generate log files for Instant Messaging in the following sections:

- Instant Messaging Log4j Configuration File (`log4j.conf`) Location
- Instant Messaging Log4j Log File Syntax
- Log4j Log Levels for Instant Messaging Components
- To Specify the Location of the Log4j Configuration File (`Log4j.conf`)
- To Enable or Disable Log4j Logging for an Instant Messaging Component
- To Set Log4j Log Levels for Instant Messaging
- To Specify the Maximum Log4j Log File Size for Instant Messaging Components

The logging levels described in [Instant Messaging Component Logging Levels](#) are used by the log4j logger.

For more information about log4j, and instructions on configuring aspects of log files, such as size, number of backups, etc., see the <http://logging.apache.org>.

Instant Messaging Log4j Configuration File (`log4j.conf`) Location

You can change the location of the log4j configuration file, `log4j.conf`, by modifying the *im.log4j.config* parameter in `im.conf`. If you do not specify a value for this parameter, the logger will look in *im-cfg-base*. If the logger does not find the log4j configuration file in that directory, it uses the logging parameters in `im.conf` to generate non-log4j style logs.

See [Instant Messaging Server Directory Structure](#) for information on locating *im-cfg-base*.

Instant Messaging Log4j Log File Syntax

The configure utility generates the log4j configuration file (`log4j.conf`) based on the content of the log4j configuration file template (`log4j.conf.template`). [Example 13-1](#) shows the log4j template. In this template:

- `${logdir}` corresponds to the directory you specified during configuration in which you want to store log files. See [Instant Messaging Log File Location](#).
- Each component's log configuration section starts with the following text:
`log4j.logger.`
where,
`xmppd` - Generates `xmppd.log`, which contains logging information for the server.
`iim_wd` - Generates `wd.log`, which contains information for the watchdog.
`xmppd.xfer` - Generates `xfer.log`, which contains only for XMPP traffic.
`agent-calendar` - Generates logging information for the Calendar agent.
`net.outer_planes.jso.BasicStream` - Generates `jso.log`, which contains information for Jabber stream objects. See the <http://jso.jabberstudio.org> website for more information.
`genredirect` - Generates `genredirect.log`, which contains information for the redirect database creation tool.
`muxd` - Generates `muxd.log`, which contains logging information for the multiplexor.
`smppbind` - Generates `smppbind.log`, which contains logging information for the sms gateway.
`router` - Generates `relay.log`, which contains logging information for shoal relay.
- A#, for example A1, are appender IDs.

Example 13-1 Log4j Template File

```

#Turn off unwanted logging
log4j.rootLogger=OFF
log4j.logger.xmppd=INFO, A1

# All logfiles are created in ${logdir}
# logdir is defined in sbin/adminrc
# It defaults to <instancedir>/log

# DEFAULT TO RollingFileAppender
log4j.appender.A1=org.apache.log4j.RollingFileAppender
log4j.appender.A1.file=${logdir}/xmppd.log
log4j.appender.A1.append=true
log4j.appender.A1.maxBackupIndex=7
log4j.appender.A1.maxFileSize=5mb
# More example appenders..
# Straight to console..
# log4j.appender.A1=org.apache.log4j.ConsoleAppender
# log4j.appender.A1.ImmediateFlush=true
# Rollover at midnight..
# log4j.appender.A1=org.apache.log4j.DailyRollingFileAppender
# log4j.appender.A1.DatePattern='.'yyyy-MM-dd
# log4j.appender.A1.file=${logdir}/xmppd.log
# log4j.appender.A1.ImmediateFlush=true
# log4j.appender.A1.append=true
# Send to SMTP..
# log4j.appender.A1=org.apache.log4j.SMTPAppender

# PATTERN LAYOUT AND OPTIONS
# DEFAULT TO PatternLayout
log4j.appender.A1.layout=org.apache.log4j.PatternLayout
# For full dates..
log4j.appender.A1.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n
# IM traditional output format..
#log4j.appender.A1.layout.ConversionPattern=%d{HH:mm:ss,SSS} %-5p %c
[%t] %m%n
# More example layouts
# XMLLayout for chainsaw consumption
# log4j.appender.A1.layout=org.apache.log4j.xml.XMLLayout
# TTCCLayout for NDC information
# log4j.appender.A1.layout=org.apache.log4j.xml.TTCCLayout
# log4j.appender.A1.layout.DateFormat=ISO8601
# log4j.appender.A1.layout.TimeZoneID=GMT-8:00
# log4j.appender.A1.layout.CategoryPrefixing=false
# log4j.appender.A1.layout.ThreadPrinting=false
# log4j.appender.A1.layout.ContextPrinting=false

# Now we list logger/appender/layout for the other default loggers, but
# only the defaults..
log4j.logger.iim_wd=ERROR, A2
log4j.appender.A2=org.apache.log4j.RollingFileAppender
log4j.appender.A2.file=${logdir}/iim_wd.log
log4j.appender.A2.append=true
log4j.appender.A2.maxBackupIndex=7
log4j.appender.A2.maxFileSize=5mb

```

```

log4j.appender.A2.layout=org.apache.log4j.PatternLayout
log4j.appender.A2.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n

# For separate xmpp traffic log, disabled by default.
log4j.logger.xmppd.xfer=INFO, A3
log4j.appender.A3=org.apache.log4j.varia.NullAppender
# Select next block instead of previous line to enable separate transfer
log
# log4j.appender.A3=org.apache.log4j.RollingFileAppender
# log4j.appender.A3.file=${logdir}/xfer.log
# log4j.appender.A3.append=true
# log4j.appender.A3.maxBackupIndex=7
# log4j.appender.A3.maxFileSize=5mb
# log4j.appender.A3.layout=org.apache.log4j.PatternLayout
# # Note, simpler default output than above 3 loggers:
# log4j.appender.A3.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t]
%m%n

log4j.logger.agent-calendar=ERROR, A4
log4j.appender.A4=org.apache.log4j.RollingFileAppender
log4j.appender.A4.file=${logdir}/agent-calendar.log
log4j.appender.A4.append=true
log4j.appender.A4.maxBackupIndex=7
log4j.appender.A4.maxFileSize=5mb
log4j.appender.A4.layout=org.apache.log4j.PatternLayout
log4j.appender.A4.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n

log4j.logger.net.outer_planes.jso.BasicStream=OFF, A5
log4j.appender.A5=org.apache.log4j.RollingFileAppender
log4j.appender.A5.file=${logdir}/jso.log
log4j.appender.A5.append=true
log4j.appender.A5.maxBackupIndex=7
log4j.appender.A5.maxFileSize=5mb
log4j.appender.A5.layout=org.apache.log4j.PatternLayout
log4j.appender.A5.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n

log4j.logger.genredirect=INFO, A6
log4j.appender.A6=org.apache.log4j.RollingFileAppender
log4j.appender.A6.file=${logdir}/genredirect.log
log4j.appender.A6.append=true
log4j.appender.A6.maxBackupIndex=7
log4j.appender.A6.maxFileSize=5mb
log4j.appender.A6.layout=org.apache.log4j.PatternLayout
log4j.appender.A6.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n

log4j.logger.muxd=ERROR, A7
log4j.appender.A7=org.apache.log4j.RollingFileAppender
log4j.appender.A7.file=${logdir}/muxd.log
log4j.appender.A7.append=true
log4j.appender.A7.maxBackupIndex=7
log4j.appender.A7.maxFileSize=5mb
log4j.appender.A7.layout=org.apache.log4j.PatternLayout
log4j.appender.A7.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n

log4j.logger.smpbind=INFO, A8
log4j.appender.A8=org.apache.log4j.RollingFileAppender

```

```
log4j.appender.A8.file=${logdir}/smppbind.log
log4j.appender.A8.append=true
log4j.appender.A8.maxBackupIndex=7
log4j.appender.A8.maxFileSize=5mb
log4j.appender.A8.layout=org.apache.log4j.PatternLayout
log4j.appender.A8.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n

log4j.logger.router=INFO, A9
log4j.appender.A9=org.apache.log4j.RollingFileAppender
log4j.appender.A9.file=${logdir}/relay.log
log4j.appender.A9.append=true
log4j.appender.A9.maxBackupIndex=7
```

```
log4j.appender.A9.maxFileSize=5mb
log4j.appender.A9.layout=org.apache.log4j.PatternLayout
log4j.appender.A9.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n
```

Log4j Log Levels for Instant Messaging Components

The log4j logger uses the same logging levels described for the `iim.conf` parameter-based logging mechanism in [Instant Messaging Component Logging Levels](#).

To Specify the Location of the Log4j Configuration File (`Log4j.conf`)

1. Open `iim.conf`.
See [iim.conf File Location](#) for information on locating this file.
2. Set the `iim.log4j.config` parameter to the path in which you want the logger to look for `log4j.conf`.
For example,
On Oracle Solaris:
`iim.log4j.config=/etc/opt/SUNWiim/default/config/log4j.conf`
On Linux:
`iim.log4j.config=/etc/opt/sun/im/default/config/log4j.conf`
3. Save and close `iim.conf`.
4. Refresh the server.
`imadmin refresh`

To Enable or Disable Log4j Logging for an Instant Messaging Component

By default, log4j logging is used for all components for which logging information is generated.

1. To disable log4j logging, set the logging level for the component to `OFF` in both `log4j.conf` and `log4j.conf.template`.
See [To Set Log4j Log Levels for Instant Messaging](#) for more information.
2. To enable log4j logging, set the logging level for the component to any logging level other than `OFF` in both `log4j.conf` and `log4j.conf.template`.

To Set Log4j Log Levels for Instant Messaging

You can set log levels by modifying either the template or the log configuration file. However, if you only modify the configuration file, any changes you make will be overwritten the next time you run `configure`. To prevent this, you should make your changes to both the configuration file and the template.

1. Open `log4j.conf.template`.
By default, this file is stored in the `im-svr-base/lib` directory.
2. For each component, specify the logging level you want to use.
For example, to set the log level for the server:
`log4j.logger.xmppd=log-level`

Where *log-level* is one of `FATAL`, `ERROR`, `WARNING`, `INFO`, or `DEBUG`.
See [Table 13-1](#) for detailed information on each of these logging levels.

1. Save and close `log4j.conf.template`.
2. Repeat the procedure for the configuration file `log4j.conf`.

To Specify the Maximum Log4j Log File Size for Instant Messaging Components

You can set log levels by modifying either the template or the log configuration file. However, if you only modify the configuration file, any changes you make will be overwritten the next time you run `configure`. To prevent this, you should make your changes to both the configuration file and the template.

1. Open `log4j.conf.template`.
By default, this file is stored in the `im-svr-base/lib` directory.
2. For each component, specify the maximum size for the component's log file.
For example, to set the size for the server log file:
`log4j.appender.A1.maxFileSize=max-logfile-size`
Where `A1` is the default appender ID for the server, `max-logfile-size` is in MB, for example `5MB`.
3. Repeat the procedure for the configuration file `log4j.conf`.

Configuring Logging for Instant Messaging Components Using `iim.conf` Parameters

If you are not using log4j to generate log files, you need to set a configuration parameter specific to each component for which you want Instant Messaging to generate logging information. This method is referred to as parameter-based logging for Instant Messaging. You can use parameter-based logging for all server instances including redirect servers, multiplexor, calendar-agent, and watchdog.



Note

This `iim.conf` parameter-based logging mechanism may be deprecated in a future release. Use log4j when possible.

Table 13-2 provides the name of the log files and the configuration parameter in `iim.conf` used to set the logging level for each Instant Messaging component log file.

Table 13-2 Log File Names and Logging Level Configuration Parameters for Instant Messaging Components

Component	Log File Name	Logging Level Configuration Parameter
Server	<code>xmppd.log</code>	<code>iim.log.iim_server.severity</code>
Multiplexor	<code>mux.log</code>	<code>iim.log.iim_mux.severity</code>
Calendar agent	<code>agent-calendar.log</code>	<code>iim.log.agent-calendar.severity</code>
Watchdog	<code>iim_wd.log</code>	<code>iim.log.iim_wd.severity</code>

The configuration parameters can have the following values:

- `fatal`
- `error`
- `warning`
- `info`
- `debug`

See [Instant Messaging Component Logging Levels](#) for information on the details logged for each logging level.

In addition, logging configuration in deployments with Access Manager is determined by the `com.iplanet.services.debug.level` property. You set this property in the `AMConfig.properties` file on the Access Manager host. By default, this file is installed in the following location:

`AM-svr-base/lib/AMConfig.properties`

Where `AM-svr-base` is the directory in which you installed *Access Manager*.

This property can contain the following values:

- message
- warning
- error
- off

By default, the Portal Server desktop log file (`desktop.debug`) and archive log files (`IMArchiveSearch.log` and `IMArchiveSubmit.log`) are stored in the following locations:

- Oracle Solaris: `/var/opt/SUNWam/debug`
- Red Hat Linux: `/var/opt/sun/am/debug`

To Set Log Levels for Instant Messaging Components Using `iim.conf` Parameters

1. Modify logging parameters in `iim.conf`.
See [Table 13-2](#) for a list of the log files and the associated parameter you need to set for each component.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`. For more information on the watchdog, see [Managing the Watchdog Process](#). For more information on the Calendar agent, see [Using Calendar Pop-up Reminders](#).

Administering Logging for Instant Messenger

By default, Instant Messenger data is not logged. You may be asked to collect client data during a support call. In this situation, you will need to enable logging before you can view client log data.

Instant Messenger logs are created on demand and stored in the user's home directory (`usr_home/.sunmsgr/messenger.log`).

Setting Up Logging for Instant Messenger

To set up logging for Instant Messenger you will need to:

1. Determine the type of data you want to collect.
2. Modify `im.jnlp` to include the `logconfig` parameter.
3. Specify a type for the `logconfig` parameter based on the type of data you want to collect.
4. Redeploy the resource files.

To Enable Logging for Instant Messenger

1. Make a backup copy of `im.jnlp`.
2. Open the `im.jnlp` Instant Messenger resource file in a text editor.
3. Search for the line:

```
<application-desc main-class="com.ipplanet.im.client.iIM">
```

4. Add the following argument to the end of the section:

```
<argument>logconfig=type</argument>
```

Where *type* is one of ALL, API, XMPPTRAFFIC, or CLIENT. See [Instant Messenger Log File Content Options](#) for details.

5. Save and close the `im.jnlp` file.
6. If you are using Application Server or Web Server, redeploy the resource files as described in [Redeploying Resource Files](#).
7. Relaunch Instant Messenger.
8. Locate the log file.
By default the log file is stored as `usr_home/.sunmsgr/messenger.log`.

You should revert back to the backup copy of `im.jnlp` when you have finished troubleshooting Instant Messenger. Then, redeploy the resource files as described in [Redeploying Resource Files](#).

Locating the Instant Messenger Log File (`messenger.log`)

By default, the Instant Messenger log file is stored as `messenger.log` under the user's home directory as follows:

```
/usr_home/.sunmsgr/messenger.log
```

Instant Messenger Log File Content Options

You can determine what activity you want logged in `messenger.log` by specifying a value for the `logconfig` parameter in `im.jnlp`. [Table 13-3](#) describes the configuration parameters for `logconfig`. See [To Enable Logging for Instant Messenger](#) for instructions on setting the `logconfig` parameter and generating Instant Messenger logs.

Table 13-3 Instant Messenger Logging Options for `messenger.log`

<i>logconfig</i> value	<code>messenger.log</code> Contains...
ALL	Information for the API, all traffic between client and server, as well as debugging information for the Instant Messenger client application itself.
API	API information only.
XMPPTRAFFIC	Client to server communication only.
CLIENT	Client application (Instant Messenger) details only.

Chapter 30. Troubleshooting and Monitoring Instant Messaging

Troubleshooting and Monitoring Oracle Communications Instant Messaging Server

This chapter lists the common problems that might occur during installation and deployment of Instant Messaging and provides an overview of the watchdog. The log information generated by the various system components on their operation can be extremely useful when trying to isolate or troubleshoot a problem. In addition, you can use the monitoring framework agent to monitor the general health of Instant Messaging processes to help prevent problems before they occur, assess usage levels to help you scale your deployment, and to prevent downtime. This chapter contains information in the following sections:

- [Troubleshooting Instant Messenger](#)
- [Problems and Solutions](#)
- [Troubleshooting Instant Messaging and LDAP](#)
- [Troubleshooting Connectivity Issues in a Multi-Node Deployment \(Server Pool\)](#)
- [Monitoring Instant Messaging](#)
- [Managing the Watchdog Process](#)

For details and more information on managing server, multiplexor, watchdog, Calendar agent, and client logging, and for default log file locations, see [Managing Logging for Instant Messaging](#).

Troubleshooting Instant Messenger

Instant Messenger provides two ways for you to help troubleshoot the client. You can gather runtime information about the client system and generate an Instant Messenger log file on demand.

Obtaining Instant Messenger Runtime Information

You can obtain information about a client system from the Instant Messenger client.

To Obtain Instant Messenger Runtime Information from the About Dialog

1. In Instant Messenger, select Help->About.
The About dialog box appears.
2. Select the Details tab.
The Details tab contains information about the client system that you can use when troubleshooting problems.

Obtaining Instant Messenger Logs

You generate client logs as required. By default, no logs are generated. See [Administering Logging for Instant Messenger](#) for information on configuring client logging.

Problems and Solutions

Listed below are some problems and their possible causes, and information to help troubleshoot these problems:

- Unable to Connect to Instant Messaging Redirect Server from Client
- Unable to Log into Instant Messenger through the XMPP/HTTP Gateway
- Messages Not Archived With Sun Java System Portal Server 7 2006Q1 or Later
- Instant Messenger Resource Customizations Lost After `patchrm` and `patchadd`
- Cannot Forward Mail to Offline Users
- Calendar Pop-up Reminders Do Not Work
- Single Sign-on Does Not Work
- Instant Messenger Does Not Load or Start
- Connection Refused or Timed Out
- Authentication Errors
- Instant Messenger Channel Display Error
- Instant Messaging Content is not Archived
- Server-to-Server Communication Fails to Start
- Catastrophic Installation Failure Leaves Server in an Inconsistent State
- Instant Messaging Services Do Not Appear in the Access Manager Console (amconsole)

Unable to Connect to Instant Messaging Redirect Server from Client

You must use a client that support XMPP redirection in order to successfully deploy the redirect server. Use Instant Messenger 2006Q1 or later, or if you use a third party client, ensure that the client that supports XMPP redirection.

Unable to Log into Instant Messenger through the XMPP/HTTP Gateway

If the XMPP/HTTP Gateway is serving two domains and the `im.jnlp` file contains an argument for only one domain, users not in the listed domain cannot authenticate. For example, if the `im.jnlp` file contains the following argument:

```
<argument>domain=mydomain.siroe.com</argument>
```

Users who attempt to log in from a domain other than `mydomain` will receive errors and cannot authenticate. To work around this problem, you need to configure Instant Messenger to authenticate to other domains.

To Configure Instant Messenger to Authenticate from a Specific Domain

1. Open the `im.jnlp` resource file.
2. Remove the domain argument entry.
For example, remove:

```
<argument>domain=mydomain.siroe.com</argument>
```

3. Download Instant Messenger again.
4. Run Instant Messenger.
The Login page appears.
5. Click More Detail.
The Login page expands to show connection settings for the client.
6. In the Server text box, enter the URL to the gateway and append `?to=domain`.
For example, if the user is part of `mydomain.siroe.com`, append the following to the URL:

```
?to=mydomain.siroe.com
```

7. To test the configuration, log in using a valid username and password.

Messages Not Archived With Sun Java System Portal Server 7 2006Q1 or Later

If you have set up a Portal Archive with Sun Java System Portal Server 7 2006Q1 or later and your messages are not being archived, ensure that the `iim_arch.portal.search` parameter is set in `iim.conf`:

```
iim_arch.portal.search="_Portal Server Search URL_"
```

Where *Portal Server Search URL* is the Search URL for the Portal Server. For example:

```
iim_arch.portal.search="http://portal.siroe.com:8080/search1/search"
```

Instant Messenger Resource Customizations Lost After `patchrm` and `patchadd`

(Issue Number: 6361796) The `patchrm` and `patchadd` processes redeploy the client resources. When this occurs, all customized files are overwritten. You need to back up any customized files you want to save before performing these actions.

Cannot Forward Mail to Offline Users

By default, Instant Messaging uses the `mail` attribute to determine the email address to which it forwards instant messages when a recipient is offline. If your directory does not use the `mail` attribute for email addresses, you need to configure Instant Messaging to use the same attribute as your directory.

To Configure the Attribute Used for User Email Addresses

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Change the value of the `iim_ldap.user.mailattr` parameter to the attribute your directory uses to contain email addresses in user entries.

Calendar Pop-up Reminders Do Not Work

If Calendar pop-ups are not being delivered as expected, you can troubleshoot the configuration as described in this section. For instructions on setting up Calendar pop-ups, see [Using Calendar Pop-up Reminders](#).

The most common error in Calendar pop-up configuration is incorrectly entered parameter names in the configuration files. This includes typos and misspelled parameter names. Ensure that you have correctly entered all of the configuration parameters and values in `iim.conf` and `ics.conf`. If you have already configured pop-ups, use [Table A-12](#) to compare your entries with the required parameters.

If your Instant Messaging and Calendar Server configuration files are correct, but pop-ups are still not arriving as expected, ensure the Calendar client and Instant Messenger are configured correctly.

To Troubleshoot Calendar Client and Instant Messenger Configuration for Pop-Ups

1. Log into the Calendar client.
2. Ensure that the time zone settings are correct.
If you are using Calendar Express, select Tools->Options->Settings from the menu.
3. Schedule an email reminder.
If you are using Calendar Express, select Tools->Options->Settings from the menu.
4. Save your settings.

5. Log into Instant Messenger with the same user.
6. Select Tools->Settings.
The Settings dialog box appears.
7. Select the Alerts tab.
8. Check the Show Calendar Reminders checkbox and click OK.
9. Leave the Instant Messenger user logged in.
10. Check to see whether or not the user received the email alert and pop-up at the time configured in the Calendar client.
If you did not receive the email alert, the Calendar client is incorrectly configured. Refer to the Calendar client documentation for further troubleshooting information.
If you received the email alert, but not the Calendar pop-up, and you are sure that you have configured both servers and clients correctly, check the `xmppd.log` for further information. You may need to set this log to a more verbose setting, for example `DEBUG`. For instructions on changing the log level, see [To Set Log Levels for Instant Messaging Components Using `im.conf` Parameters](#).

Single Sign-on Does Not Work

If you are using SSO with Sun Java System Access Manager, the Access Manager server and Instant Messaging server must be configured to use the same web container.

Instant Messenger Does Not Load or Start

The following are the possible causes for this problem:

- Wrong codebase in the applet page.
- Application/x-java-jnlp-file MIME type not defined in the web container configuration.
- Plug-in of Java Web Start not installed or functional.
- No compatible Java version available.
- Security exception, cannot verify signature of `.jar` files.

Where to get the necessary information:

- In the Java Web Start or plug-in errors (exception stack trace, launch page.)
- In the applet page source on the browser.

Connection Refused or Timed Out

The following are the possible causes for this problem:

- Either the Instant Messaging server or the multiplexor is not running.
- Incorrect multiplexor host or port names used in the Applet descriptor file `.jnlp` or `.html`.
- Different SSL settings used between Instant Messenger and the multiplexor.
- Client and server version mismatch.

Where to get diagnostic information:

- Instant Messaging server and multiplexor log files.
- Instant Messenger logs.
- Instant Messenger About dialog box, Details tab.

Authentication Errors

The following are the possible causes for this problem:

- Problems while accessing the LDAP server, such as the LDAP server is not running, or a provisioning error, such as a schema violation, has occurred.

- End user not found.
- Invalid credentials.
- Invalid Instant Messenger session.

Where to get diagnostic information:

- Instant Messaging server, Identity authentication, and LDAP log files.
- In deployments using Sun Java System Access Manager, ensure that the user entries in your Directory contain the `iplanet-am-managed-person` objectclass. The Instant Messaging server uses this object class when it searches for valid users in an Access Manager deployment. For more information about this object class and how Access Manager uses it, refer to the Sun Java System Access Manager documentation.

Instant Messenger Channel Display Error

The following are the possible causes for this problem:

- The server cannot validate the session token.
- Instant Messaging channel is not configured properly. For example, incorrect Instant Messaging server host, port, or both.
- Plug-in or Java Web Start is not installed or is not functional.
- End user not found and the Instant Messaging server cannot find the end user when performing an LDAP lookup.

Where to get diagnostic information:

Instant Messaging server and Instant Messaging channel logs.

Instant Messaging Content is not Archived

The following are the possible causes for this problem:

- Content is actually archived but the end user has insufficient rights to access it.
- The content has not yet been committed to the database.
- The archive provider has been disabled in the Instant Messaging server.

Where to get diagnostic information:

Instant Messaging server and the archive log files.

Server-to-Server Communication Fails to Start

The following are the possible causes for this problem:

- Incorrect server identification.
- Mismatch in the SSL settings.

Where get diagnostic information:

The Instant Messaging server log file for both servers.

Catastrophic Installation Failure Leaves Server in an Inconsistent State

If a catastrophic error occurs while installing or uninstalling Instant Messaging, the system might be left in an inconsistent state. This results in both install and uninstall being unable to complete. In this circumstance, you must manually remove all the Instant Messaging components so that a fresh install can be attempted. The clean up procedure consists of removing packages and registry information.

To Manually Remove All Instant Messaging Components

1. Back up any information you might need in a future installation.
See [Backing Up Instant Messaging Data](#) for instructions.

2. Manually edit the product registry information.

For Oracle Solaris 9, type the following command:

```
prodreg(1)
```

For all other operating systems:

- a. Edit `productregistry.xml` and remove all Instant Messaging XML elements from the file.

By default, the `productregistry` XML file is stored in the following locations:

Oracle Solaris: `/var/sadm/install/productregistry`

Red Hat Linux: `/var/tmp/productregistry`

- b. Remove the following packages or RPMs if they are still present:

- SUNWiim
- SUNWiimc
- SUNWiimd
- SUNWiimid
- SUNWiimin
- SUNWiimjd
- SUNWiimm
- SUNWiimc-110n
- SUNWiimd-110n
- SUNWiimid-110n
- SUNWiimin-110n

Instant Messaging Services Do Not Appear in the Access Manager Console (amconsole)

If Instant Messaging uses Access Manager policies in a Sun Java System Application Server deployment, you need to restart the Application Server when you finish configuring Instant Messaging. If you do not restart the Application Server, Instant Messaging services will not appear in the Access Manager console (amconsole).

Troubleshooting Instant Messaging and LDAP

The following LDAP issues might arise in a given deployment. Change the LDAP parameters in `iim.conf` accordingly.

Using a Directory That Does not Permit Anonymous Bind

By default, Instant Messaging server performs an anonymous search of the LDAP directory. However, it is common for sites to prevent anonymous searches in their directory so that any random person cannot do a search and retrieve all the information. If your site's directory is configured to prevent such anonymous searches, and you didn't provide bind credentials during post-installation configuration, you need to configure the Instant Messaging server needs with a user ID and password it can use to bind and perform searches.

Use the `iim_ldap.usergroupbinddn` and `iim_ldap.usergroupbindcred` parameters to configure the necessary credentials.

To Configure Bind Credentials for the Instant Messaging Server

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Specify the DN you want the server to use to bind to the directory as the value for

iim_ldap.usergroupbinddn.

```
iim_ldap.usergroupbinddn=bind-DN
```

3. Specify the password that corresponds to the bind DN as the value for *iim_ldap.usergroupbindcred*

```
iim_ldap.usergroupbindcred=password
```

4. Save and close the file.

Displaying Contact Names Using an Attribute Other than `cn`

You can customize how Instant Messenger displays contact names. The default attribute used by Instant Messenger to display contact names is `cn`. Contact names appear as *First Name, Last Name*. For example, Frank Smith, Mary Jones, and so on.

Use the *iim_ldap.userdisplay* and *iim_ldap.groupdisplay* parameters to specify which attribute to use to display contact names.

To Change the Attribute Used to Display Contact Names

1. Open *iim.conf*.
See [iim.conf File Syntax](#) for instructions on locating and modifying *iim.conf*.
2. Specify the attribute you want to use to display user names as the value for *iim_ldap.userdisplay*.

```
iim_ldap.userdisplay=user-name-attribute
```

3. Specify the attribute you want to use to display group names as the value for *iim_ldap.groupdisplay*

```
iim_ldap.groupdisplay=group-name-attribute
```

4. Save and close the file.

Searching the Directory Using Wildcards

If your directory is indexed to allow the use of wildcards, and you want to be able to use wildcards while searching for contact names, you need to configure the Instant Messaging server to allow wildcard searches. However, allowing wildcard searches can impact performance unless User IDs are indexed for substring search. See [Modifying How Client Users Search for Contacts](#) for instructions on allowing wildcard searches in Instant Messaging.

Using Nonstandard Objectclasses for Users and Groups

If your directory uses nonstandard objectclasses to define users and groups you need to change the appropriate *iim_ldap.** parameters, replacing `inetorgperson` and `groupofuniquenames` with your values.

See [LDAP and User Registration Configuration Parameters](#) for a list of LDAP parameters.

To Change the Objectclasses Used to Specify Users and Groups

1. Open *iim.conf*.
See [iim.conf File Syntax](#) for instructions on locating and modifying *iim.conf*.

2. Search for and replace `inetorgperson` with the object class used to define users in your directory.
3. Search for and replace `groupofuniqueNames` with the object class used to define groups in your directory.
4. Save and close the file.

Using an Attribute Other than `uid` for User Authentication

If your directory does not use the `uid` attribute for user authentication, you need to configure the Instant Messaging server with the attribute used by your directory. By default, Instant Messaging uses `uid`. You also need to change each filter parameter that contains `uid` in its value.

Use the `iim_ldap.loginfilter` parameter to specify which attribute to use for user authentication.

To Change the Attribute Used for User Authentication

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Search for and replace `uid` with the attribute you want to use for user authentication in the following parameters:
`iim_ldap.loginfilter`
`iim_ldap.usergroupbyidsearchfilter`
3. Save and close the file.

Using an Attribute Other than `uid` for User IDs

If your directory does not use the `uid` attribute for user IDs, you need to configure the Instant Messaging server with the attribute used by your directory. By default, Instant Messaging uses `uid`. In addition, you should index the attribute in the directory to help offset any performance degradation caused by searching on unindexed attributes.

Use the `iim_ldap.useruidattr` parameter to specify which attribute to use for user IDs.

To Change the Attribute Used for User IDs

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Specify the attribute you want to use for user IDs as the value for `iim_ldap.useruidattr`.
The default value is `uid`.
For example, to use the `loginname` attribute, set the `iim_ldap.useruidattr` attribute as follows:
`iim_ldap.useruidattr=loginname`
3. Save and close the file.
4. Add the index directive to the indexing rules in LDAP:
`index loginname eq`

Troubleshooting Connectivity Issues in a Multi-Node Deployment (Server Pool)

If you are receiving errors where presence status is not being shared between servers in a server pool:

- Ensure that the nodes are configured correctly to enable server-to-server communication. See [Configuring Server-to-Server Communication Between Instant Messaging Servers in a Server Pool](#) for a list of configuration parameters and appropriate values.
- Check for server-to-server session establishment errors in the log file.

Monitoring Instant Messaging

Instant Messaging provides an agent to help you monitor activity. This agent is called the monitoring framework management agent, or `mfwk` agent. The `mfwk` agent is contained within the Common Agent Container (CAC). The `mfwk` agent is installed with Instant Messaging. The CAC ships with Java ES and is installed using the Java ES installer. For more information about installing, enabling, and administering monitoring, as well as an overview of Instant Messaging objects monitored, see the [Sun Java Enterprise System 5 Monitoring Guide](#).

Managing the Watchdog Process

The watchdog process monitors the server and multiplexor components and attempts to restart a component if it determines that the component is not running.

For the server, the watchdog determines whether the server is running by periodically attempting to make a connection, either directly to the server or through the multiplexor, based on the current configuration of the server. The watchdog tries to poll the server's operational status and if it cannot determine the status, it then tries to make a connection to the server. If both operations fail, the watchdog stops and then restarts the server.

Before you use the watchdog, verify that it is enabled and running using the `imadmin status` command. By default, the watchdog is enabled and running when you install Instant Messaging.

More information about the `imadmin` utility is available in [Instant Messaging imadmin Tool Reference](#).

Determining the Status of the Watchdog

You use the `imadmin` command-line utility to check the status of the watchdog.

To Determine the Status of the Watchdog

1. Change to the directory that contains the `imadmin` command-line utility.
`cd im-svr-base/sbin`
2. Run `imadmin status`:
`./imadmin status watchdog`
The `imadmin` utility returns the current status of the watchdog.

Enabling and Disabling the Watchdog

By default, the watchdog is enabled when you install Instant Messaging. You can disable or enable the watchdog by setting a configuration parameter in `iim.conf`.

To Enable or Disable the Watchdog

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Enable or disable the watchdog by setting the `iim_wd.enable` parameter as follows:
To enable the watchdog:
`iim_wd.enable=true`
To disable the watchdog:
`iim_wd.enable=false`
3. Save and close the `iim.conf` file.
4. Refresh the Instant Messaging server configuration:

```
cd im-svr-base/sbin
./imadmin refresh
```


Managing Logging for the Watchdog

You manage logging for the watchdog the same way you manage logging for the server, multiplexor, and the Calendar agent. The watchdog log file is saved as *im-db-base/log/iim_wd.log*.

For more information on setting logging levels for all Instant Messaging components including the watchdog, see [Managing Logging for Instant Messaging](#).

Chapter 31. Using Calendar Pop-up Reminders

Using Calendar Pop-up Reminders

Instant Messaging is integrated with Calendar Server to provide automatic pop-up reminders to Instant Messenger users for both calendar events and tasks.

This section contains the following topics:

- [Pop-up Reminders Overview](#)
- [Configuring Instant Messaging Pop-ups](#)
- [Configuring Calendar Pop-ups in a Server Pool](#)
- [Administering the Calendar Agent](#)

Pop-up Reminders Overview

This section contains information about Calendar pop-up reminders in the following topics:

- [Pop-up Reminders Operation](#)
- [Pop-up Reminders Architectural Flow](#)
- [iim.conf Calendar Pop-up Configuration Parameters](#)

Pop-up Reminders Operation

Users can receive Instant Messenger pop-up reminders for upcoming events and tasks on their calendars. To enable these pop-up reminders, the following must occur:

- The administrator must configure the Calendar server and the Instant Messaging server to allow pop-up notifications.
- The end user must specify email reminders in the Options tab of either Calendar Express or Communications Express, which sets an alarm in the Event Notification System.
- The end user must enable calendar reminders in Instant Messenger.

With pop-ups enabled, when an impending event or task nears, the alarm set in the Event Notification System causes Calendar Server to send an email notification and Instant Messaging to display a pop-up reminder.

Pop-up Reminders Architectural Flow

If configured, Instant Messaging pop-up reminders follow this architectural flow:

1. The Instant Messaging JMS subscriber subscribes to Calendar server events and notifications in the Event Notification Service (ENS).
2. Calendar server publishes an event or task notification in `text/xml` or `text/calendar` format to ENS.
3. The Instant Messaging JMS subscriber receives the calendar event or task notification and then generates a message in `text/calendar` format.
4. The Instant Messaging server sends the message to the calendar owner, if the end user is online.
5. If the recipient is available, Instant Messenger generates an HTML pop-up reminder on the end user's desktop based on the message. If the recipient is not available, the Instant Messaging

server discards the message.

`iim.conf` Calendar Pop-up Configuration Parameters

When you install Instant Messaging, several configuration parameters used with the Calendar agent are added by default to `iim.conf`. You can also enable the Calendar agent and provide associated configuration information when you run the `configure` utility. However, you might want to manually configure pop-ups, for example, if you have customized the resource files for Instant Messenger. If you rerun `configure`, you will then need to redeploy the resource files. If you choose to manually configure the Instant Messaging server for Calendar pop-ups instead of running the `configure` utility, you will need to provide values for these parameters. See [Configuring Instant Messaging After Installation](#) for information on the `configure` utility.

[Table 16-1](#) lists the configuration parameters you will use to configure the Instant Messaging server and the Calendar agent in order to use Calendar pop-ups.

Table 16-1 `iim.conf` Parameters for Configuring Calendar Pop-ups

Parameter or Section in <code>iim.conf</code>	Description and Appropriate Value
JMS Consumers Section	
<code>jms.consumers</code>	Name of alarm. Set the value to: <code>calreminder</code>
<code>jms.consumer.calreminder.destination</code>	Destination of the alarm. This must be the same as the value of the <code>caldb.serveralarms.url</code> configuration parameter in the <code>ics.conf</code> file. For example, <code>enp:///ics/customalarm</code>
<code>jms.consumer.calreminder.provider</code>	The name of the provider. Set to <code>ens</code> . This must be the same as the name in the <code>jms.providers</code> parameter in the JMS Providers section.
<code>jms.consumer.calreminder.type</code>	The type of alarm to set. Set the value to: <code>topic</code>
<code>jms.consumer.calreminder.param</code>	The alarm parameter. Set the value as follows including the quotes: <code>"eventtype=calendar.alarm"</code>
<code>jms.consumer.calreminder.factory</code>	A listener that registers itself for the new calendar reminder messages. Set the value to: <code>com.iplanet.im.server.JMSCalendarMessageListener</code> Enter the value on a single line.
JMS Providers Section	
<code>jms.providers</code>	The name of the provider. Set value to <code>ens</code> . This must be the same as the value listed in the JMS Consumers Section for the <code>jms.consumer.calreminder.provider</code> parameter.
<code>jms.provider.ens.broker</code>	Hostname of the ENS and the port number on which the ENS listens for incoming requests. Set to the port specified in the <code>ics.conf</code> file parameter <code>service.ens.port</code> . The default is 57997. For example: <code>jms.provider.ens.broker=cal.example.com:57997</code>
<code>jms.provider.ens.factory</code>	Factory class used for creating the topic connection objects. Set the value to: <code>com.iplanet.ens.jms.EnsTopicConnFactory</code>
Instant Messaging General Parameters	
<code>iimagent.enable</code>	Enables agents for Instant Messaging. By default, this parameter is set to <code>False</code> . Set the value as follows including the quotes: <code>iimagent.enable="true"</code>
<code>iimagent.agent-calendar.enable</code>	Loads a component that enables the Calendar agent. Set the value as follows including the quotes: <code>iimagent.agent-calendar.enable="true"</code>
<code>agent-calendar.jid</code>	The JID of the Calendar agent. Set this value as follows: <code>agent-calendar.jid=calimbot.{server}.{domain}</code>
<code>agent-calendar.password</code>	Set this parameter to a password you want the Calendar agent to use to connect to the Instant Messaging server. Set this value as follows: <code>agent-calendar.password=password</code>
<code>iimserver.components</code>	Set this value as follows: <code>iimserver.components=agent-calendar</code>

The following parameters are introduced in IM 8.0. If upgrading from versions earlier than IM 8.0, you

must set the parameters manually.

Parameter	Value	Description
<code>agent-calendar.imadmin.enable</code>	"false"	If set to true, you can start the agent-calendar by using the <code>imadmin</code> command.
<code>agent-calendar.iim_server.host</code>		Hostname of the Instant Messaging server with which the agent calendar communicates.
<code>agent-calendar.iim_server.port</code>		Port number of the Instant Messaging server with which the agent calendar communicates.

Configuring Instant Messaging Pop-ups

This section includes the following configuration instructions:

- [To Configure Instant Messaging Server for Calendar Pop-ups Using the `configure` Utility](#)
- [To Manually Configure Instant Messaging Server for Calendar Pop-ups](#)
- [To Configure Calendar Server for Pop-ups](#)
- [To Configure Instant Messenger for Calendar Pop-ups](#)

To Configure Instant Messaging Server for Calendar Pop-ups Using the `configure` Utility

1. Run `configure`.
See [Completing the Configuration Checklist](#) for more information about the `configure` utility.
2. On the Calendar Agent configuration screen, select the Enable Calendar Agent check box.
3. Enter the Notification Server hostname and port number.
Use the same port number as the port number specified by the `service.ens.port` parameter in the `ics.conf` file on the Calendar Server.
The values you provide are combined and stored as the value for the `jms.provider.ens.broker` parameter in `iim.conf`. For example, if you enter `localhost` for the hostname and `57997` for the port number, the `jms.provider.ens.broker` parameter would be set as follows:

```
jms.provider.ens.broker=localhost:57997
```
4. Enter the Calendar Alarm URL.
This URL is the destination of the alarm. For example:

```
enp:///ics/customalarm
```


Use the same URL as the URL specified by the `caldb.serveralarms.url` parameter in the `ics.conf` file on the Calendar Server.
The value you provide is stored as the value for the `jms.consumer.cal_reminder.destination` parameter in `iim.conf`.
5. Click Next and continue with configuration.
See [Configuring Instant Messaging After Installation](#) for more information about the `configure` utility.

To Manually Configure Instant Messaging Server for Calendar Pop-ups

Before you begin, gather the information in [Table 16-1](#).

1. Edit one or more of the parameters in the `iim.conf` file as shown in [Table 16-1](#).
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
The parameter values shown assume you want pop-up reminders for both events and tasks. If these parameters do not already exist in `iim.conf`, add them.
2. Start the Calendar agent using `imadmin`.

```
imadmin start agent-calendar
```

The `imadmin` command-line utility is located in the following directory:

```
im-svr-base/sbin
```

Where `im-svr-base` is the directory in which you installed Instant Messaging.

To Configure Calendar Server for Pop-ups

1. Log in to the Calendar server host as an administrator with permission to change the configuration.
2. Change to the `cal-svr-base/SUNWics5/cal/config` directory.
Where `cal-svr-base` is the directory in which you installed Calendar Server.
3. Save your old `ics.conf` file by copying and renaming it.
4. Confirm that the parameters shown in the following table have the values shown. If they do not, you need to modify them.

Parameter	Description and Default Value
<code>caldb.serveralarms</code>	Enables calendar alarms to be queued. The default is "1" (enabled).
<code>caldb.serveralarms.contenttype</code>	Output format for alarm content. The default is "text/xml".
<code>caldb.serveralarms.dispatch</code>	Enables calendar alarms to be dispatched. The default is "yes".
<code>caldb.serveralarms.dispatchtype</code>	The type of server alarm to dispatch. The default is "ens".
<code>caldb.serveralarms.url</code>	This is the URL for alarm retrieving alarm contents. The default is "enp:///ics/customalarm".

5. Save the `ics.conf` file.
6. Restart Calendar server.
`cal-svr-base/SUNWics5/cal/sbin/start-cal`
Where `cal-svr-base` is the directory in which you installed Calendar Server.

To Configure Instant Messenger for Calendar Pop-ups

1. On the Instant Messenger Main window, select Tools Settings.
2. On the Settings window, click the Alerts tab.
3. Check the Show Calendar Reminders option.
4. Click OK.
Users can now receive Calendar pop-ups through Instant Messenger while they are online.

Configuring Calendar Pop-ups in a Server Pool

To configure Calendar pop-ups to work in a server pool deployment, you only need to configure one server's Calendar agent in the pool. A pop-up will be delivered for each configured Calendar agent in the pool.

Administering the Calendar Agent

The Calendar agent is an Instant Messaging component that provides pop-up functionality to Calendar and Instant Messaging users. In addition, using tools provided with Instant Messaging, you can start, stop, restart, or check the status of the Calendar agent as well as monitor its activity through log files. See [Stopping, Starting, Refreshing, and Checking Instant Messaging Components](#) for information on administering the Calendar agent component. Also see [Managing Logging for Instant Messaging](#) for information about Calendar agent logs. This section describes enabling and disabling Instant Messaging agents.

Enabling and Disabling Instant Messaging Agents

1. Open `iim.conf`.
See [iim.conf File Syntax](#) for instructions on locating and modifying `iim.conf`.
2. Set the `iim_agent.enable` parameter to `true`:
`iim_agent.enable="true"`
3. Save and close `iim.conf`.
4. Refresh the server.
`imadmin refresh server`

Chapter 32. Using the Instant Messaging XMPP and HTTP Gateway

Using the Oracle Communications Instant Messaging Server XMPP/HTTP Gateway

The XMPP/HTTP Gateway provides Instant Messaging access to non-XMPP 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.

The XMPP/HTTP Gateway is deployed with the Instant Messenger resource files as a webapp on the web container.

This chapter provides information on configuring and maintaining the XMPP/HTTP Gateway in the following sections:

- [Instant Messaging XMPP/HTTP Gateway Configuration Files](#)
- [Configuring the Instant Messaging XMPP/HTTP Gateway](#)
- [Securing Communication Between the XMPP/HTTP Gateway and Instant Messaging Server Using StartTLS](#)
- [Managing Logging for the XMPP/HTTP Gateway](#)

Instant Messaging XMPP/HTTP Gateway Configuration Files

The XMPP/HTTP Gateway uses the following files for configuration:

- Gateway web application configuration file (`web.xml`). The contents of this file determine which gateway configuration file to use. For information on using a non-default configuration file, see [To Configure the Instant Messaging XMPP/HTTP Gateway to Use a Non-default Configuration File](#).
- Gateway configuration file (typically `httpbind.conf`). See [Configuring the Instant Messaging XMPP/HTTP Gateway](#) for instructions on configuring the gateway. See [Instant Messaging XMPP/HTTP Gateway Configuration Parameters in `httpbind.conf`](#) for a description of `httpbind.conf` file syntax, file location, and a list of configuration parameters in this file.
- Gateway logging configuration file (typically `httpbind_log4j.conf`). See [Managing Logging for the XMPP/HTTP Gateway](#) for more information on configuring logging. See [XMPP/HTTP Gateway log4j Log Configuration File Syntax](#) for logging configuration file syntax.

Configuring the Instant Messaging XMPP/HTTP Gateway

When you run the `configure` utility after installation, you can choose to deploy the XMPP/HTTP Gateway or not. If enabled, the `configure` utility creates a default configuration file (`httpbind.conf`) for the gateway. You can change the configuration by modifying the values in this file. See [Instant Messaging XMPP/HTTP Gateway Configuration Parameters in `httpbind.conf`](#) for a description of `httpbind.conf` file syntax, file location, and a list of configuration parameters in this file, or refer to the instructions in this section.

In addition, when you choose to deploy the gateway during initial configuration, the `configure` utility creates a `.war` file in the `im-svr-base/work` directory and then deploys this file on the web or application server in the directory you specified for the codebase.

You can also configure the gateway to use a non-default configuration file by modifying the values in `web.xml` which is deployed with the client resources on the web container.

The instructions in this section assume the gateway configuration file is `httpbind.conf`. If you are using a non-default configuration file, substitute your configuration file for `httpbind.conf` in the instructions.

Any time you make a change to `httpbind.conf`, you will need to restart the XMPP/HTTP Gateway.

This section contains the following instructions:

- To Enable or Disable the Instant Messaging XMPP/HTTP Gateway
- To Configure the Number of Concurrent Requests Handled by the XMPP/HTTP Gateway
- To Set the JEP 124 *hold* Attribute for Client Requests to the XMPP/HTTP Gateway
- To Specify the Allowed Client Inactivity Time for the XMPP/HTTP Gateway
- To Set the `content-type` HTTP Header for the XMPP/HTTP Gateway
- To Set the Round Trip Delay for the XMPP/HTTP Gateway
- To Set the Default Time Within Which the XMPP/HTTP Gateway Will Send a Response to the Client
- To Configure an XMPP/HTTP Gateway in a Instant Messaging Gateway Pool
- To Configure the List of Key IDs for Supported XMPP/HTTP Gateway Domains
- To Configure the Instant Messaging XMPP/HTTP Gateway to Use a Non-default Configuration File

For instructions on configuring logging for the gateway, see [Managing Logging for the XMPP/HTTP Gateway](#).

To Enable or Disable the Instant Messaging XMPP/HTTP Gateway

You enable the gateway by running the `configure` utility and then setting a parameter in `iim.conf`. You can disable the gateway later using tools provided by your web container or application server.

1. To enable the gateway:
 - a. Invoke the `configure` utility.
 - b. Choose to deploy the gateway when prompted.
See [Configuring Instant Messaging After Installation](#) for more information.
 - c. Set the `iim_agent.httpbind.enable` parameter to `true` in the `iim.conf` file.
For example, `iim_agent.httpbind.enable=true`
2. To disable the gateway, disable the webapp using the tools provided by the web or application server.

To Configure the Number of Concurrent Requests Handled by the XMPP/HTTP Gateway

Ensure that you are familiar with the JEP 124 draft standard. More information is available at <http://www.jabber.org/jeps/jep-0124.html>.

1. Open `httpbind.conf`.
See [httpbind.conf File Location](#) for information on finding this file.
2. Set the `httpbind.requests` parameter to the maximum number of concurrent requests a single client can send to the gateway. The default is 2. For example:

```
httpbind.requests=2
```

The number of concurrent requests a client can make to the gateway. If the value of this parameter is less than the value for the JEP 124 *hold* attribute in the client request, the value for this parameter will be set to *hold*+1. Do not set this parameter to 1, as doing so could severely degrade performance. See [To Set the JEP 124 hold Attribute for Client Requests to the XMPP/HTTP Gateway](#) and [Table B-1](#) for more information on the `httpbind.hold` parameter.
3. Save and close `httpbind.conf`.

4. Restart the gateway using the tools provided by the web or application server.

To Set the JEP 124 *hold* Attribute for Client Requests to the XMPP/HTTP Gateway

Ensure that you are familiar with the JEP 124 draft standard. More information is available at <http://www.jabber.org/jeps/jep-0124.html>.

1. Open `httpbind.conf`.
See [httpbind.conf File Location](#) for information on finding this file.
2. Set the `httpbind.hold` parameter to the maximum value you want the gateway to allow for the *hold* attribute in the client request. The default is 5. For example:
`httpbind.hold=5`
If the hold value sent by the client is greater than the gateway's hold value, the gateway's hold value is used.
3. Save and close `httpbind.conf`.
4. Restart the gateway using the tools provided by the web or application server.

To Specify the Allowed Client Inactivity Time for the XMPP/HTTP Gateway

1. Open `httpbind.conf`.
See [httpbind.conf File Location](#) for information on finding this file.
2. Set `httpbind.inactivity` parameter to the time in seconds after which you want the gateway to terminate idle connections. The default is 180 seconds. For example:
`httpbind.inactivity=180`
If clients do not poll the gateway before this time elapses, the gateway terminates the connection.
3. Save and close `httpbind.conf`.
4. Restart the gateway using the tools provided by the web or application server.

To Set the `content-type` HTTP Header for the XMPP/HTTP Gateway

1. Open `httpbind.conf`.
See [httpbind.conf File Location](#) for information on finding this file.
2. Set the `httpbind.content_type` parameter to the content-type you want the gateway to use if the client does not specify one in its initial request. The default is `text/xml; charset=utf-8`. For example:
`httpbind.content_type=text/xml; charset=utf-8`
3. Save and close `httpbind.conf`.
4. Restart the gateway using the tools provided by the web or application server.

To Set the Round Trip Delay for the XMPP/HTTP Gateway

The round trip delay is the amount of time, in seconds, you want to allow in addition to time-outs for round trips between gateway and client. This helps to account for network latencies.

1. Open `httpbind.conf`.
See [httpbind.conf File Location](#) for information on finding this file.
2. Set the `httpbind.round_trip_delay` parameter as required.
Setting this value too high may degrade performance. The value is in seconds. The default is 1 second. For example:
`httpbind.round_trip_delay=1`
Setting this value too high may degrade performance. Consider the general latency in your network before changing this parameter.
3. Save and close `httpbind.conf`.
4. Restart the gateway using the tools provided by the web or application server.

To Set the Default Time Within Which the XMPP/HTTP Gateway Will Send a Response to the Client

1. Open `httpbind.conf`.
See [httpbind.conf File Location](#) for information on finding this file.
2. Set the `httpbind.wait_time` parameter as required.
The client is guaranteed a response from the XMPP/HTTP Gateway within the wait time you designate with this parameter. Consider the speed of your network when setting this parameter. Do not set the value so low that the XMPP/HTTP Gateway is unlikely to be able to send the request in time.
The value is in seconds. The default is 120 seconds. For example:
`httpbind.wait_time=120`
If the value set for the client is greater than the value for the gateway, the gateway wait time is used.
3. Save and close `httpbind.conf`.
4. Restart the gateway using the tools provided by the web or application server.

To Configure an XMPP/HTTP Gateway in a Instant Messaging Gateway Pool

1. Open `httpbind.conf`.
See [httpbind.conf File Location](#) for information on finding this file.
2. To configure the gateway as part of a deployment with an Instant Messaging gateway pool:
 - a. Set the `httpbind.pool.support` parameter to `true`:
`httpbind.pool.support=true`
 - b. Set the `httpbind.pool.nodeld` parameter to the full URL of the gateway, for example, `http://host.domain.com:80/httpbind/httpbind`.
The URL is used as the gateway's `nodeld`. This `nodeld` must be unique within the server pool. The gateway uses this `nodeld` to determine whether it must service a received request or forward the request to another gateway in the pool.
3. To configure the gateway not to work within a gateway pool, set the `httpbind.pool.support` parameter as follows:

```
httpbind.pool.support=false
```

4. Save and close `httpbind.conf`.
5. Restart the gateway by using the tools provided by the web or application server.



Note

Support for `httpbind` pool is currently not available for `async` or `Comet` `httpbind` implementations. In such cases, if a load balancer (for example) is used in front of the `httpbind` gateways, the load balancer needs to maintain session stickiness.

To Configure the List of Key IDs for Supported XMPP/HTTP Gateway Domains

1. Open `httpbind.conf`.
See [httpbind.conf File Location](#) for information on finding this file.
2. Set the `httpbind.config` parameter to the list of IDs you want the gateway to use.
For each domain you need to specify a separate ID for this parameter. For example:

```
httpbind.config=gwdomain-id
```

Where `gwdomain-id` is the identifier you want to use for the domain.
For example:

```
httpbind.config=siroe.com
```

3. For each *gwdomain-id* you specify, add the following parameters to the `httpbind.conf` file:

```
gwdomain-id.domain=domain-name  
gwdomain-id.hosts=gateway-host  
gwdomain-id.componentjid=component-jid  
gwdomain-id.password=password
```

Where:

gwdomain-id is the ID specified for the gateway in `httpbind.config` in the previous step.

domain-name is the domain in which the identified gateway runs.

gateway-host is a comma-separated or space-separated list of the fully-qualified domain name (FQDN) and port number of the gateway hosts that support this domain.

component-jid is the component JID of the gateway.

password is the password of the identified gateway.

For example, if the *gwdomain-id* is set to `siroe`:

```
siroe.domain=siroe.com  
siroe.hosts=gateway.siroe.com:5222  
siroe.componentjid=http.gateway.siroe.com  
siroe.password=gatewaypassword
```

See [Gateway Domain ID Key Parameters for httpbind.config](#) for more information about these key parameters.

4. Save and close `httpbind.conf`.
5. Restart the gateway using the tools provided by the web or application server.

To Configure the Instant Messaging XMPP/HTTP Gateway to Use a Non-default Configuration File

1. On the web container on which Instant Messenger resource files are deployed, edit `web.xml`. Use your web container's tools to edit this file.
2. Change the value for the `httpbind.config.file` parameter to the location of the configuration file you want the gateway to use.

Securing Communication Between the XMPP/HTTP Gateway and Instant Messaging Server Using StartTLS

The XMPP/HTTP Gateway only supports StartTLS for secure communications. If the multiplexor is configured to use legacy SSL, you need to configure the gateway to connect directly to the server, bypassing the multiplexor. The gateway will always attempt to use StartTLS if it is available. See [Securing Instant Messaging Using TLS and Legacy SSL](#) for more information.

Managing Logging for the XMPP/HTTP Gateway

You can configure the level of logging for the XMPP/HTTP Gateway, enable or disable logging entirely, and change the location of the gateway log file or the gateway log configuration file as described in the following sections:

- To Enable or Disable Logging for the XMPP/HTTP Gateway
- To Change the Location of the XMPP/HTTP Gateway Log Configuration File
- Linux: To Set the Location of the XMPP/HTTP Gateway Log File After Install or Upgrade
- To Change the Location of the XMPP/HTTP Gateway Log File
- To Use a Non-default Log File Location for the XMPP/HTTP Gateway
- To Set the XMPP/HTTP Gateway Logging Level
- XMPP/HTTP Gateway log4j Log Configuration File Syntax

More information about the log4j format supported by Instant Messaging's is described at the <http://logging.apache.org>.

To Enable or Disable Logging for the XMPP/HTTP Gateway

You can enable or disable logging for the gateway in two ways:

- Adding or removing the value for the *httpbind.log4j.config* parameter in *httpbind.conf*.
- (Recommended) Modifying the configuration within the gateway's log4j configuration file (*httpbind_log4j.conf*).

Under most circumstances, you should modify the configuration in the *httpbind_log4j.conf* file itself, leaving the *httpbind.log4j.config* parameter set to the location of the *httpbind_log4j.conf* file. This procedure describes modifying the configuration within the *httpbind_log4j.conf* file.

1. Open the *httpbind_log4j.conf* file.
This file is stored at the location you specified in *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 instance:

```
<im-cfg-base>/httpbind_log4j.conf
```

2. To disable logging for the gateway, set the *log4j.logger.httpbind* parameter as follows:

```
log4j.logger.httpbind=OFF
```

3. To enable logging, set the *log4j.logger.httpbind* parameter to the desired logging level.
For example:

```
log4j.logger.httpbind=ERROR
```

See [Table 13-1](#) for a list of valid logging levels you can use.

4. Save and close *httpbind_log4j.conf*.

To Change the Location of the XMPP/HTTP Gateway Log Configuration File

1. Open *httpbind.conf*.
See [httpbind.conf File Location](#) for information on finding this file.
2. Set the value of the *httpbind.log4j.config* parameter to the location of the XMPP/HTTP Gateway log configuration file.
3. Save and close *httpbind.conf*.
4. Restart the gateway using the tools provided by the web or application server.

Linux: To Set the Location of the XMPP/HTTP Gateway Log File After Install or Upgrade

On Linux, after you install and configure the XMPP/HTTP Gateway, you need to modify the location of the default log file for the XMPP/HTTP gateway in `httpbind_log4j.conf`.

1. Open the `httpbind_log4j.conf` file.
This file is stored at the location you specified in `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 instance:

```
<im-cfg-base>/httpbind_log4j.conf
```

2. Set the value of the `log4j.appender.appender_ID.file` parameter to the location where log files are stored.

To Change the Location of the XMPP/HTTP Gateway Log File

Ensure that you are familiar with the log4j syntax and general implementation described at the <http://logging.apache.org>.

1. Open `httpbind_log4j.conf`.
This file is stored at the location you specified in `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 instance:

```
<im-cfg-base>/httpbind_log4j.conf
```

2. Set the value for the `log4j.appender.appender-ID` parameter to the location where you want to store the log file.
3. Save and close `httpbind_log4j.conf`.
4. Restart the web container.

To Use a Non-default Log File Location for the XMPP/HTTP Gateway

If you choose to use a location for logs other than the default, you need to modify the location of the default log file for the XMPP/HTTP gateway in `httpbind_log4j.conf`.

1. Open the `httpbind_log4j.conf` file.
This file is stored at the location you specified in `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 instance:

```
<im-cfg-base>/httpbind_log4j.conf
```

2. Set the value of the `log4j.appender.appender_ID.file` parameter to the location where log files are stored.

To Set the XMPP/HTTP Gateway Logging Level

Ensure that you are familiar with the log4j syntax and general implementation described at the <http://logging.apache.org>.

1. Open `httpbind_log4j.conf`.
This file is stored at the location you specified in `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 instance:

```
<im-cfg-base>/httpbind_log4j.conf
```

2. Set the *log4j.logger.httpbind* parameter to the desired logging level. For example:

```
log4j.logger.httpbind=ERROR
```

See [Table 13-1](#) for a list of valid logging levels you can use.

XMPP/HTTP Gateway log4j Log Configuration File Syntax

For more information about the log4j syntax and general implementation, see the <http://logging.apache.org>. The gateway log configuration file syntax is as follows.

```
log4j.logger.httpbind=_logging-level_, _Appender-ID_
# DEFAULT TO RollingFileAppender
log4j.appender._Appender-ID_=org.apache.log4j.RollingFileAppender
log4j.appender._Appender-ID_.file=_log-dir_/httpbind.log
log4j.appender._Appender-ID_.append=true|false
log4j.appender._Appender-ID_.maxBackupIndex=7
log4j.appender._Appender-ID_.maxFileSize=_max-log-file-size_
log4j.appender._Appender-ID_.layout=org.apache.log4j.PatternLayout
log4j.appender._Appender-ID_.layout.ConversionPattern=_log-entry-syntax_
```

Example 10-1 XMPP/HTTP Gateway Log Configuration File (*httpbind_log4j.conf*)

```
log4j.logger.httpbind=ERROR, A7
# DEFAULT TO RollingFileAppender
log4j.appender.A7=org.apache.log4j.RollingFileAppender
# log4j.appender.A7.file=$(logdir)/httpbind.log
log4j.appender.A7.file=_log-dir_/httpbind.log
log4j.appender.A7.append=true
log4j.appender.A7.maxBackupIndex=7
log4j.appender.A7.maxFileSize=5mb
log4j.appender.A7.layout=org.apache.log4j.PatternLayout
log4j.appender.A7.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n
```

Chapter 33. Configuring External Gateways with Instant Messenger

Configuring External Gateways with Oracle Communications Instant Messaging Server

This chapter describes the external gateway feature and explains the procedure to configure external gateways with Instant Messaging.

- [External Gateway Overview](#)
- [Configuring External Gateways](#)

External Gateway Overview

Instant Messaging client uses the XMPP protocol to interact with the Instant Messaging server. The external gateways allow the Instant Messaging users to communicate with their contacts on other networks such as, MSN, AOL, Yahoo. Users who want to communicate with their contacts on other networks should download and configure gateways that are specific to each type of network.

Configuring External Gateways

To set up the external gateway with Instant Messaging, perform the following steps:

1. Download the external gateway that you require. For example,
PyMSNt: Gateway for the MSN client
PyAIMt: Gateway for the AIM client
PyYIMt: Gateway for the YAHOO client
2. Enable the gateway to interact with the Instant Messenger server.

To enable the gateway, make changes to the gateway configuration and the Instant Messaging server-side configuration.

Gateway Configuration

To configure the gateway to interact with Instant Messaging, do the following:

1. Download the gateway that you require. For example, PyMSNt.
2. Set up the gateway according to the instructions in the gateway's documentation.
3. Replace the value of the JID class, password, server name, port, socks server name and port number parameters with the Instant Messaging server credentials in the gateway's `config.xml` file.

Instant Messaging Server-Side Configuration

1. Register the 3rd party messaging client by adding the following parameters in the `iim.conf` file.


```
iim_agent.enable = <true|false>
iim_server.components = <component>
iim_server.components.password = <password>
iim_server.components.jid=<hostname>
```

Example to register the MSN client

```
iim_agent.enable = true
iim_server.components = "agent-calendar,httpbind,msngateway-comp"
msngateway-comp.password=secret
msngateway-comp.jid=msn.hostname.siroe.com
```

2. Restart the Instant Messaging server.

To access the external gateway feature in the Instant Messaging client, do the following:

1. Launch the Instant Messaging client.
2. Go to File -> Add Services.
A list of transports that you configured are displayed, For example, MSN Transport.
3. Select the required transport and provide the user name and password. For example, provide the MSN user name and password to connect to the MSN transport.
Once you are logged in, your contacts on the external network gets added to existing contacts list in the Instant Messaging client.

Chapter 34. Switching httpbind from servlet to async Mode

Switching httpbind from servlet to async Mode

The `httpbind` application provided with Instant Messaging server runs in servlet (blocking) mode by default, which imposes limitations on the number of simultaneous `httpbind` client connections. In servlet mode, each `httpbind` client connection holds at least one application-server worker thread.

`httpbind` can be re-configured to make use of the [Grizzly framework](#) and run in asynchronous mode instead, which improves scalability by only holding an application-server worker thread when there is client/server activity.



Note

Convergence uses an internal deployment of `httpbind` for Instant Messaging functionality. The internal Convergence `httpbind` deployment operates in asynchronous mode by default.

- [Prerequisites for running httpbind in asynchronous mode](#)
- [Instructions for switching httpbind from servlet to asynchronous mode](#)
- [How can I tell if httpbind is running in servlet mode?](#)
- [Do you re-apply the `web.xml.template` changes after Instant Messaging server is upgraded?](#)

Prerequisites for running httpbind in asynchronous mode

1. Use a supported version of GlassFish Server.
Refer to the latest [Convergence release notes](#) for the supported versions of GlassFish Server.
2. Ensure Application Server has `cometSupport` enabled:

```
# cd /opt/SUNWappserver/bin
# ./asadmin get --user admin --secure=true --host localhost --port
4848 \
'server.http-service.http-listener.http-listener-*.property.cometSupport
= true
server.http-service.http-listener.http-listener-2.property.cometSupport
= true
```

If `cometSupport` isn't enabled you will see:

```
No matches resulted from the wildcard expression.
CLI137 Command get failed.
```

`cometSupport` can be enabled by running:

```
# cd /opt/SUNWappserver/bin
# ./asadmin set --user admin --secure=true --host localhost --port
4848 \
'server.http-service.http-listener.http-listener-1.property.cometSupp
./asadmin set --user admin --secure=true --host localhost --port
4848 \
'server.http-service.http-listener.http-listener-2.property.cometSupp
```

Instructions for switching httpbind from servlet to asynchronous mode

These steps were tested with Instant Messaging Server 8u2 running on Solaris 10(x86_64) with Sun GlassFish Enterprise Server 2.1.1 Patch04 with HADB.

1. Backup the `web.xml.template` file of `httpbind` application which exists in `/opt/sun/comms/im/lib/` by default e.g.

```
cp -p /opt/sun/comms/im/lib/web.xml.template
/opt/sun/comms/im/lib/web.xml.template.orig
```

2. Modify the `web.xml.template` file of `httpbind` application which exists in `/opt/sun/comms/im/lib/` by default.

- Change the servlet class entry with the `async-mode` version:
replace:

```
<servlet-class>com.sun.im.gateway.http.servlet.ConnectionManager
```

with:

```
<servlet-class>com.sun.im.gateway.http.async.comet.HttpbindComet
```

- Add a servlet `init-param` setting – `servlet_uri` – and set this value to match the `servlet_mapping`'s URL pattern (which is `/httpbind` by default) e.g.

```
<init-param>
  <param-name>servlet_uri</param-name>
  <param-value>/httpbind</param-value>
</init-param>
```

- Comment out the `httpbind.library.checkpermission` `init-param` setting.
- This results in the following `web.xml` file when the `httpbind` application is deployed to `/httpbind`:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web
Application 2.3//EN"
"http://java.sun.com/xml/ns/j2ee/web-app_2_3.xsd">
<web-app>
  <servlet>
    <servlet-name>httpbind</servlet-name>
<!-- Replace the default servlet connection manager with
the async manager -->
<!--
<servlet-class>com.sun.im.gateway.http.servlet.ConnectionManager
-->

<servlet-class>com.sun.im.gateway.http.async.comet.HttpbindComet
<init-param>
  <param-name>httpbind.config.file</param-name>

<param-value>/etc/opt/SUNWiim/default/config/httpbind.conf</para
</init-param>
<!-- Commenting out the httpbind.library.checkpermission
setting -->
<!--
  <init-param>

<param-name>httpbind.library.checkpermission</param-name>

<param-value>/opt/sun/comms/im/sbin/../../lib/libcheckperm.so</pa
</init-param>
-->
<!-- New init-param setting for httpbind running in async mode
-->
  <init-param>
    <param-name>servlet_uri</param-name>
    <param-value>/httpbind</param-value>
  </init-param>
</servlet>
<servlet-mapping>
  <servlet-name>httpbind</servlet-name>
  <url-pattern>/httpbind</url-pattern>
</servlet-mapping>
<session-config>
  <session-timeout>
    30
  </session-timeout>
</session-config>
</web-app>

```

3. Redeploy the httpbind application

```

cd /opt/sun/comms/im/sbin/
./iwadmin undeploy httpbind
./iwadmin deploy httpbind

```

4. Restart Application Server (Glassfish)

How can I tell if httpbind is running in servlet mode?

Check the `jstack` output of the Application Server instance which is running the httpbind application and check for a thread-stack containing

```
com.sun.im.gateway.http.servlet.ConnectionManagerServlet.doPost.
```

For example:

```
-bash-3.00# /usr/jdk/latest/bin/jstack 16841
...
"httpSSLWorkerThread-80-0" daemon prio=3 tid=0x08b7e800 nid=0x69 waiting
on condition [0xcbc78000]
  java.lang.Thread.State: TIMED_WAITING (sleeping)
    at java.lang.Thread.sleep(Native Method)
    at
com.sun.im.gateway.http.HTTPBindSession.dequeueMessages(HTTPBindSession.java:325)
at
com.sun.im.gateway.http.HTTPBindSession.process(HTTPBindSession.java:325)
at
com.sun.im.gateway.http.servlet.ConnectionManagerServlet.processRequest(ConnectionManagerServlet.java:754)
at
com.sun.im.gateway.http.servlet.ConnectionManagerServlet.doPost(ConnectionManagerServlet.java:754)
at javax.servlet.http.HttpServlet.service(HttpServlet.java:847)
  at sun.reflect.GeneratedMethodAccessor76.invoke(Unknown Source)
  at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:597)
at java.lang.reflect.Method.invoke(Method.java:597)
  at
org.apache.catalina.security.SecurityUtil$1.run(SecurityUtil.java:292)
  at java.security.AccessController.doPrivileged(Native Method)
```

This stack indicates you are running httpbind in servlet mode.

Do you re-apply the `web.xml.template` changes after Instant Messaging server is upgraded?

Yes.

After upgrading Instant Messaging server, the `web.xml.template` file will be replaced by the default version which uses servlet mode.

Chapter 35. Writing a Custom SSO Module for Instant Messaging Server

Writing a Custom SSO Module for Instant Messaging Server

Instant Messaging Server provides a Single Sign-on (SSO) mechanism for [Sun Access Manager \(Legacy Mode\)](#).

This information describes how to write a custom SSO module for Instant Messaging Server so that SSO support can be added for third-party SSO solutions.

As with any SSO aware application, when a user is authenticated by using Access Manager for example, Instant Messaging Server loads the authentication module to validate the user. On successful validation, the user is allowed to access the application. If the validation is not successful, the standard password validation occurs.

Topics:

- [API Reference Documentation](#)
- [How the Custom SSO Module for Instant Messaging Server Works](#)
- [Troubleshooting](#)



Note

Starting with Instant Messaging 7.3, the custom SSO provider could not be enabled without a fix for CR 6981119: Cannot enable custom SSO provider without setting `iim.policy.modules = "identity"`. This issue is now fixed by the most current Instant Messaging 8 Update 3 patch.

API Reference Documentation

The SSOProvider API documentation is provided as part of the Instant Messaging installation.

If the Instant Messaging client has been deployed to a web-container then the API documentation can be found in the following location:

```
http://web-container/IM client deployment path/apidoc/
```

For example:

```
http://host1.example.com:80/im/apidoc
```

Alternatively, the documentation can be found in `im-svr-base/html/apidoc`, that is, `/opt/sun/comms/im/html/apidoc`.

How the Custom SSO Module for Instant Messaging Server Works

Instant Messaging calls the `verify` method provided by the custom SSO provider with the `uid` and `token` arguments.

The token argument is the password provided by the Instant Messaging client during the SASL PLAIN authentication. For example the client has sent the following:

```
<body rid='607740' sid='7675823097240743042'  
xmlns='http://jabber.org/protocol/httpbind'  
key='c654f46426c6d12cf2a0e1a9beb218915e0afd6f' ><auth  
xmlns='urn:ietf:params:xml:ns:xmpp-sasl'  
mechanism='PLAIN'>c2hqb3J0aEBhdS5vcmFjbGUuY29tAHNoam9ydGgAc2VjcmV0dG9rZW4=
```

c2hqb3J0aEBhdS5vcmFjbGUuY29tAHNoam9ydGgAc2VjcmV0dG9rZW4= is the base64 encoded SASL PLAIN string as per [XEP-0034](#) and this decodes to:

```
shj@example.com<NUL>shjorth<NUL>secrettoken
```

where:

shj@example.com is the authorization identity

shj is the authentication identity

secrettoken is the password (or in this case the SSO token) provided by the IM client

<NUL> is the NUL character

Implementing the Custom SSO Module

Before designing a solution for the custom SSO module, the Instant Messaging SSO provider framework needs to be implemented:

- All custom SSO modules must implement SSOProvider interface.
- The SSO verification implementation must provide the domain of the authenticating user if `iim.userprops.store` is set to `ldap`.
- The SSO implementation can use any other classes that are required to make the custom SSO module work.

To Implement the Custom SSO Module

1. Create the Custom SSO java file. Use the sample code that follows.

```
mkdir -p com/client/sample/
```

com/client/sample/CustomSSOProvider.java

```
import java.util.Map;  
import com.iplanet.im.server.RealmManager;  
import com.iplanet.im.server.LocalUser;  
import com.iplanet.im.server.LDAPUserSettings;  
import com.iplanet.im.common.util.Log;  
  
public class CustomSSOProvider implements SSOProvider {  
  
    // This is called for each authentication. It can return true or  
    // false.  
    public boolean verify(String uid, String token, java.util.Map  
attributes, java.util.Set attributeNames){
```

```

        String domain = "your.domain.com";
        Log.debug(String.format("CustomSSO: Trying to authenticate
user: %s, token = %s\n", uid, token));

        // Replace the following check with your SSO token
verification routine
        if (token.equalsIgnoreCase("secrettoken") == false) {
            Log.debug("CustomSSO: 'secrettoken' not provided as
password, SSO login attempt failed");
            return false;
        }

        // If user properties are stored in LDAP then need to retrieve
them
        if (RealmManager.getUserSettingsStorageProvider() instanceof
LDAPUserSettings) {
            Log.debug("CustomSSO: iim.userprops.store = \"ldap\"");

            try{
                LocalUser u = RealmManager.getUser(uid, domain, true);

                // the domain is mandatory if using a hosted domain
configuration
                // the boolean third parameter forces an LDAP fetch
                // (disregarding previously cached entries)
                if(u != null){
                    for(Object o: attributeNames){
                        String s = (String) o;
                        Log.debug("CustomSSO: attributeName: " + s);
                        Set prop = u.getAttributeValues(s);
                        attributes.put(s, prop);
                    }
                    return true;
                }
            }
            catch(RealmException ex){
                Log.error("CustomSSO: could not load user: " + uid + " in
domain: " + domain);
                return false;
            }
        }
        else {
            Log.debug("CustomSSO: iim.userprops.store = \"file\"");
            return true;
        }
        return false;
    }

    // This is called each time there's some XMPP activity by the
user. The
    // SSO implementation can use this to keep the session from
timing out.
    public boolean refresh(String uid) {
        Log.debug("CustomSSO: refresh called " + uid);
        return true;
    }

```



```
}

// This is for any SSO initialization and is called once on
server startup
public void open() throws Exception {
    Log.debug("CustomSSO: open called");
}

// this is called before the server is shutdown, though its not
guaranteed
public void close() {
```

```
        Log.debug("CustomSSO: close called");
    }
}
```

2. Compile the source.

Make sure to compile with JDK1.6. This example uses the JDK installed by the Communications Suite installer under `/usr/jdk/latest` on Solaris OS and Instant Messaging Server under `/opt/sun/comms/im`.

```
/usr/jdk/latest/bin/javac -classpath
/usr/share/lib/xmpp/improvider.jar:/opt/sun/comms/im/lib/xmppd.jar:/opt/
com/client/sample/CustomSSOProvider.java
```

3. Create the jar archive from the compiled files.

```
/usr/jdk/latest/bin/jar -cvf CustomSSOProvider.jar
com/client/sample/CustomSSOProvider.class
```

4. Move the compile jar file to the Instant Messaging library directory.

```
chown bin:bin CustomSSOProvider.jar
cp -p CustomSSOProvider.jar /opt/sun/comms/im/lib
```

5. Add the jar file to the imadmin command.

```
cp imadmin imadmin.orig
```

Edit the imadmin command and add the following to the end of the `server_classpath` variable setting:

```
$PCD/CustomSSOProvider.jar
```

6. Modify the `iim.conf` file to enable the provider.

```
iim_server.usesso = "1"
iim_server.ssoprovider = "com.client.sample.CustomSSOProvider"
```

7. Restart the XMPP server.

```
cd /opt/sun/comms/im/
./imadmin stop
./imadmin start
```

8. You should now see the following in the `xmppd.log` file (when debug log-level is enabled in the `log4j.conf` file) when logging in by using the password `secrettoken`:

```
[13 Sep 2010 08:53:34,857] DEBUG xmppd [Thread-15] CustomSSO:
Trying to authenticate user: shjorth, token = secrettoken
[13 Sep 2010 08:53:34,857] DEBUG xmppd [Thread-15] CustomSSO:
iim.userprops.store = "ldap"
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: uid
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: uniquemember
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: givenname
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: sunPresenceAccessPermitted
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: sunIMUserNewsRoster
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: sunIMUserConferenceRoster
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: userpassword
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: sunIMRoster
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: sunIMConferenceRoster
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: sunIMNewsRoster
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: sunIMUserProperties
[13 Sep 2010 08:53:34,898] DEBUG xmppd [Thread-15] CustomSSO:
attributeName: sunPresenceEntityDefaultAccess
...
```

Troubleshooting

- For Solaris OS installations of Instant Messaging, check the following file for errors when starting the Instant Messaging server:

```
/var/svc/log/application-sunim:default.log
```

Chapter 36. Performance, Scalability, and Sizing Considerations for Instant Messaging

Performance, Scalability, and Sizing Considerations for Instant Messaging

This information describes how to enhance tuning and performance of Instant Messaging.

Topics:

- [Tuning Instant Messaging Server Memory](#)
- [Instant Messaging Thread Pooling and Service Port Configuration](#)
- [Service Port Configuration](#)
- [Sample Load Test of the Instant Messaging Server](#)

Tuning Instant Messaging Server Memory

Use the J2SE (Java 2 Platform Standard Edition) platform, version 6 for running the Instant Messaging server, because of increased performance. The J2SE platform does not require command-line tuning as it supports ergonomic features.

For more information about the use of J2SE, see <http://java.sun.com/javase/6/docs/>

The Instant Messaging server uses the `im.jvm.maxmemorysize` parameter in the `im.conf` file to set the maximum size of the JVM (Java Virtual Machine) software heap to allocate. The default value of this parameter is 256 Mbytes. However, a large active deployment of Instant Messaging needs more memory. Determining the amount of memory to allocate for the Instant Messaging server depends on the number of concurrent active users that you need to support.

Additional load per user, use of additional Instant Messaging services like news or file transfer, and use of features such as message filters, archiving, or SSL require more memory. You should perform load profiling of typical user activity before deploying Instant Messaging into a production environment. Contact Oracle Support Services for more information about load profiling an Instant Messaging deployment.

Instant Messaging Thread Pooling and Service Port Configuration

Instant Messaging provides a set of configuration options to tailor the size and behavior of thread pools used to service client-to-server and server-to-server requests. These thread pools combined with the associated service ports can improve the throughput of an Instant Messaging server.

Option Name	Description	Default Value
<code>iim_server.maxthreads</code>	Maximum number of threads for the default thread pool	50
<code>iim_server.threadpool</code>	List of independent thread pools	All parameters use the default thread pool
<code>iim_server.threadpool.capacity</code>	Capacity of the default thread pool	10 * maxthreads
<code>iim_server.threadpool.aaa.maxthreads</code>	Maximum threads for the named thread pool <code>aaa</code> : <code>maxthreads (aaa)</code>	4
<code>iim_server.threadpool.aaa.capacity</code>	Capacity of the named thread pool <code>aaa</code> .	10 * maxthreads <code>aaa</code>

The following table lists the defined thread pools for Communications Suite

Name	Use
<code>s2s-in</code>	All server-to-server inbound communications. If the port allows server-to-server inbound communications, Instant Messaging uses this thread pool.
<code>s2s-out</code>	All server-to-server outbound communications. If the port allows server-to-server outbound communications, Instant Messaging uses this thread pool.
<code>s2s</code>	All server-to-server communications. The combination of <code>s2s-in</code> and <code>s2s-out</code> .

Defined thread pools can be specified and used with an associated server-only service port, as described in [Service Port Configuration](#). You can edit the thread and port configurations in the `iim.conf` file. You need to restart the server after making changes to the thread and port configurations.

Service Port Configuration

The following are the service port configuration options.

Option	Definition	Default Value
<code>iim_server.useport</code>	Open normal ports (allow StartTLS)	true
<code>iim_server.usesslport</code>	Open SSL ports (non-negotiable TLS)	false
<code>iim_server.usemuxport</code>	Open Multiplexor ports	true
<code>iim_server.port</code>	List of normal ports	5269
<code>iim_server.sslport</code>	List of SSL ports	5270
<code>iim_mux.serverport</code>	List of Multiplexor ports	45222
<code>iim_server.port.port.sndbuf</code>	Socket send buffer size	none
<code>iim_server.port.port.rcvbuf</code>	Socket recv buffer size	none
<code>iim_server.port.port.interface</code>	List of specific network interfaces to bind	none (Indicates any)
<code>iim_server.port.port.protocol</code>	List of protocols permitted on this port. The value can be client, server, component, or peer	all or any
<code>iim_server.port.port.nodelay</code>	Enables the Nagles algorithm	false

The throughput of a service port can be improved by adjusting the send or receive buffer of the port.

The following example shows the service ports configuration for Instant Messaging

```
iim_server.port = 5269, 45269, 15222
iim_server.port.5269.protocol = s2s
iim_server.port.45269.protocol = component
iim_server.port.45269.sndbuf= 512000
iim_server.port.45269.rcvbuf= 512000
iim_server.port.15222.protocol = c2s
```

Sample Load Test of the Instant Messaging Server

The following table shows a sample load test of the Instant Messaging server.

Platform Details	System Configuration	Server Heap Size	No. of Users	No. of Concurrent Sessions	User Cache	Load Per 10 Seconds
<ul style="list-style-type: none"> • Oracle's Sun Fire T1000 Server • Solaris 10 OS • RAM 16 GB 	Server and Multiplexor installed in the same box	1 GByte for Mux 5 GBytes for server	100,000	60000	128 count	<ul style="list-style-type: none"> • 50 users login to the server • 50 users logout • 1450 presence updates • 350 messages sent to offline destinations • 2400 messages sent to online destinations • 850 messages sent to random destinations • 50 roster additions • 50 roster rename • 50 roster removal

The above sample has the following configuration parameters in the `iim.conf` file.

```

iim.jvm.maxmemorysize = "4096"
iim_server.memory.user.cache_count="128"
iim_server.scratch_directory="/tmp/imsscratch"
iim_ldap.maxconns=70
iim_server.maxthreads=50
iim_server.jvm.options="-d64"
iim_mux.jvm.options="-d64"
iim_mux.maxsessions="100000"
iim_server.maxsessions="150000"

```

where,

- `iim_server.memory.user.cache_count` specifies the memory user cache size. In the above sample, the value is set to 128 for a user base of 100,000. If the user base is more than 100,000, increase this value proportionately.
- `iim_server.scratch_directory` specifies the directory where the user cache is written to the disk. It is recommended to have the scratch directory on `tempfs`. For 100,000 user base in the Solaris 10 OS, around 500 to 600 MBytes of space is required on a filesystem and around 4 to

5 GBytes of space is required on `tempfs`.

- `iim_ldap.maxconns` specifies the LDAP context pool size. In case of more roster operations and in a server pool environment, increase this value appropriately.
- `iim_server.maxthreads` specifies the size of the thread pool. If you do not have sufficient memory to keep user cache in `tempfs`, you can increase the value of the thread pool.
- `iim_server.jvm.options` enables you to run the 64-bit JVM thereby enabling big heap sizes.
- `iim_mux.jvm.options` enables you start the multiplexor in the 64-bit mode.
- `iim_mux.maxsessions` specifies the maximum number of concurrent client connection a multiplexor can accept.
- `iim_server.maxsessions` specifies the number of sessions allowed through an instance of multiplexor connected to the server.



Warning

Disable the watchdog by setting the `iim_wd.enable` parameter to `false` in the `iim.conf` file.

Chapter 37. Reference Information

Oracle Communications Instant Messaging Server Reference Information

- [Appendix A Instant Messaging Configuration Parameters in iim.conf](#) describes the settings you can configure for Instant Messaging components.
- [Appendix B Instant Messaging XMPP/HTTP Gateway Configuration Parameters in httpbind.conf](#) describes the settings you can configure for the gateway.
- [Appendix C Instant Messaging imadmin Tool Reference](#) describes the imadmin command used to administer Instant Messaging.
- [Appendix D Instant Messaging APIs](#) provides an overview of the APIs used by Instant Messaging.
- [Appendix E Instant Messaging LDAP Schema](#) defines modifications made to the LDAP schema for Instant Messaging.

Chapter 38. Instant Messaging Configuration Parameters in `iim.conf`

Oracle Communications Instant Messaging Server Configuration Parameters in `iim.conf`

This chapter explains the Instant Messaging configuration parameters in the `iim.conf` file in the following sections:

- `iim.conf` File Location
- `iim.conf` File Syntax
- General Configuration Parameters
- LDAP and User Registration Configuration Parameters
- Logging Configuration Parameters
- Instant Messaging Server Configuration Parameters
- Multiple Server Configuration Parameters
- Shoal Configuration Parameters
- Multiplexor Configuration Parameters
- Redirect Server Parameters
- Archive Parameters
- Watchdog Parameters
- Monitoring Parameters
- Agent Parameters
- HTTP/XMPP Gateway Parameters
- SMS Integration Parameters
- MSN Gateway Integration Parameters
- AIM Gateway Integration Parameters
- Yahoo Gateway Integration Parameters
- IMPS Gateway Parameters
- Java Message Queue (JMQ) Parameters
- Conference History Parameters

`iim.conf` File Location

Instant Messaging stores configuration settings in the `iim.conf` file within the Configuration Directory (*im-cfg-base*).

- On Solaris OS:
`/etc/opt/SUNWiim/default/config/iim.conf`
- On Red Hat Linux:
`/etc/opt/sun/im/default/config/iim.conf`

If you created multiple instances of Instant Messaging, the name of the `/default` directory will vary depending on the instance. See [Creating Multiple Instances from a Single Instant Messaging Installation](#) for more information.

`iim.conf` File Syntax

This file is a plain ASCII text file, with each line defining a server parameter and its value(s):

- A parameter and its value(s) are separated by an equal sign (=) with spaces and tabs allowed before or after the equal sign.
- A value can be enclosed in double quotes (" "). If a parameter allows multiple values, the entire value string must be enclosed in double quotes.
- A comment line must have an exclamation point (!) as the first character of the line. Comment lines are for informational purposes and are ignored by the server.
- If a parameter appears more than once, the value of the last parameter listed overrides the previous value.
- A backslash (\) is used for continuation and indicates the value(s) are longer than one line.
- Each line is terminated by a line terminator (\n, \r, or \r\n).
- The key consists of all the characters in the line starting with the first non-whitespace character and up to the first ASCII equal sign (=) or semi-colon (;). If the key is terminated by a semi-colon, it is followed by "lang-" and a tag that indicates the language in which this value is to be interpreted. The language tag is followed by an equal sign (=). All whitespace characters before and after the equal sign are ignored. All remaining characters on the line become part of the associated value string.
- Multiple values in the value string are separated using commas (,).
- Within a value, if any special characters like comma, space, newline, tab, double quotes, or backslash are present, the entire value needs to be within double quotes. In addition, every carriage return, line feed, tab, backslash, and double quotes within the value must be specified with a backslash (\).
- If you make changes to `iim.conf`, you must refresh the Instant Messaging server in order for the new configuration settings to take effect.



Note

The `iim.conf` file is initialized by the installation process and should be modified only as described in this guide.

General Configuration Parameters


Table A-1 lists and describes the general configuration parameters.

Table A-1 General Configuration Parameters

Parameter	Default Value	Description
<code>iim.comm.modules</code>	<code>iim_server, iim_mux</code>	The communication modules used. Possible values are <code>iim_server</code> and <code>iim_mux</code> . The default value is <code>iim_server, iim_mux</code> , which means both the server and multiplexor are used. The <code>iim_mux</code> value is useful for the multiplexor.
<code>iim.smtpserver</code>	<code>localhost</code>	SMTP server to send mail to end users who have set the option for forwarding their messages as emails or to pages.
<code>iim.instancedir</code>	<code>/opt</code>	The installation directory root.

<code>iim.instancevardir</code>	Solaris: <code>/var/opt/SUNWiim/default</code> Linux: <code>/var/opt/sun/im/default</code>	Sets the directory to contain runtime files, including the end-user profile database, logs, and other files created by the server and multiplexor at runtime. The name of the <code>/default</code> directory may vary if you created multiple instances of Instant Messaging.
<code>iim.user</code>	<code>inetuser</code> for LDAP deployments. <code>root</code> for portal deployment.	The end-user name with which the server processes run.
<code>iim.group</code>	<code>inetgroup</code> for LDAP deployments. <code>root</code> for portal deployment.	The group using which the server processes run.
<code>iim.jvm.maxmemorysize</code>	256	The maximum number heap size in bytes the JVM running the server is allowed to use. Used to construct the <code>-mx</code> argument of the Java command.
<code>iim.mail.charset</code>	None	This parameter specifies if the headers of the mail are in ASCII and not encoded. It contains the name of the character set to be used to encode the headers of the mail message sent for offline alerts. For example: <code>iim.mail.charset=iso-2022</code>
<code>iim.jvm.command</code>	<code>/usr/j2se/bin/java</code>	The location of the Java Runtime Executable (JRE).
<code>iim.identity.basedir</code>	<code>/opt</code>	The default installation directory, also referred to as the base directory, for Sun Java System Access Manager.
<code>iim.identity.jre</code>	<code>/usr/java_1.3.1_04</code>	The location of the JRE used by the Access Manager to run all its processes.
<code>iim.portal.deployuri</code>	<code>/portal</code>	The URI using which the Portal Server <code>war</code> files are deployed in the Access Manager.
<code>iim.portal.host</code>	<code>imhostname</code>	The host name of the server on which the Portal Server is running. Specify port number if a non default port number is used.
<code>iim.portal.protocol</code>	<code>http</code>	The protocol used to access the Portal Server.

<code>iim.policy.cache.validity</code>	10	<p>Defines the cache validity interval (in minutes) for a single user's information. The Instant Messaging server saves the last date a single end-user's information was cached. If the end-user's information is accessed after the interval determined by this parameter, the server will recache the end user's information and reset the cache date on the <code>LocalUser</code> object.</p>
<code>iim.policy.modules</code>	<code>iim_ldap</code>	<div data-bbox="1068 489 1430 768" style="background-color: #ffffcc; padding: 5px;"> <p>Starting with Instant Messaging 8 Update 3, the value <code>identity</code> which indicates that the Sun Java System Access Manager is used for policy storage, has been deprecated.</p> </div> <p>By default, LDAP is used for policy storage. The property <code>iim.policy.modules</code> must be set to <code>identity</code>, <code>iim_ldap</code>, <code>iim_ldap_schema1</code>, or <code>iim_ldap_schema2</code> only if hosted domain support is required. If you configure Instant Messaging via the configurator, the possible values are <code>identity</code> and <code>iim_ldap</code>. The allowed values which can be manually set for <code>iim.policy.modules</code> are <code>identity</code>, <code>iim_ldap</code>, <code>iim_ldap_schema1</code>, and <code>iim_ldap_schema2</code>. Only in the absence of hosted domain support <code>iim_ldap</code> be used as a value for <code>iim.policy.modules</code> for both <code>schema1</code> and <code>schema2</code> of the LDAP server.</p> <div data-bbox="1068 1409 1430 1776" style="background-color: #d9e1f2; padding: 5px;"> <p>Note The properties <code>iim_ldap_schema1</code> and <code>iim_ldap_schema2</code> must be set in conjunction with other parameters mentioned in Configuring Hosted Domain Support.</p> </div>

<code>iim.policy.resynctime</code>	720	Defines the cache validity interval (in minutes) for all end-user information. The Instant Messaging server clears cached end-user information on a regular basis in order to eliminate stale end-user information. This parameter specifies the frequency at which the cached end-user information is cleared.
<code>iim.userprops.store</code>	file	<p>By default, user properties are stored in a user properties file if you chose to use Access Manager for policy when you ran the <code>configure</code> utility. If you chose to use Access Manager for policy, the default is <code>ldap</code>. Change the value to change the location where user properties are stored. If you change the value from <code>file</code> to <code>ldap</code>, you need to run <code>imadmin assign_services</code> to assign the required objectclasses to user entries in the directory. This parameter is only significant when the service definitions for the Presence and Instant Messaging services have been installed.</p> <div data-bbox="1068 894 1430 1415" style="background-color: #e6f2ff; padding: 10px; border: 1px solid #add8e6;"> <p> Note</p> <p>In Convergence, if the Instant Messaging server is configured with a user properties file, end users are unable to upload Avatars. When using Instant Messaging in Convergence, the <code>iim.userprops.store</code> parameter should be set to <code>ldap</code> not <code>file</code> in <code>iim.conf</code>.</p> </div>
<code>iim_wd.jvm.maxmemorysize</code>	256	The maximum heap size in MB that the running the watchdog is allowed to use. Used to construct the <code>-mx</code> argument of the Java command.
<code>iim_mux.jvm.maxmemorysize</code>	256	The maximum heap size in MB that the running the multiplexor is allowed to use. Used to construct the <code>-mx</code> argument of the Java command.
<code>iim_server.classpath</code>	custom classpath used by imadmin	Can be used to append any additional jar files to the existing classpath of the server process.

LDAP and User Registration Configuration Parameters

Table A-2 lists and describes the parameters used by Instant Messaging for LDAP, user registration, and user source configuration.

Table A-2 LDAP, User Registration, and Source Configuration Parameters

Parameter	Default Value	Description
<code>iim_ldap.host</code>	<code>localhost:389</code>	LDAP server name and port used by Instant Messaging server for end-user authentication.
<code>iim_ldap.searchbase</code>	<code>o=internet</code>	The string used to search for the users and groups on the LDAP server.
<code>iim_ldap.usergroupbinddn</code>	None (the server performs anonymous searches)	Specifies the DN to bind to the LDAP server for searches.
<code>iim_ldap.usergroupbindcred</code>	None (the server performs anonymous searches)	Specifies the password to use with the <code>iim_ldap.usergroupbinddn</code> for LDAP searches.
<code>iim_ldap.loginfilter</code>	<code>(&((objectclass=inetorgperson)(objectclass=webtopuser))(uid={0}))</code>	Search filter used for end-user login. The filter is entered as a single line.
<code>iim_ldap.usergroupbyidsearchfilter</code>	<code>((&(objectclass=groupofuniquenames)(uid={0})) (&((objectclass=inetorgperson)(objectclass=webtopuser))(uid={0})))</code>	The search filter used to search for end-user groups in the directory under the base search by ID. The entire filter is entered as one line.
<code>iim_ldap.usergroupbynamefilter</code>	<code>((&(objectclass=groupofuniquenames)(cn={0})) (&((objectclass=inetorgperson)(objectclass=webtopuser))(cn={0})))</code>	The search filter used to search for end-user groups in the directory under the base search by name.
<code>iim_ldap.allowwildcardinuid</code>	False	Determines if wildcards should be enabled for UID searches. As most installations have users indexed for exact searches only, the default value is False. Setting this value to True can impact search performance unless all users are indexed for search.

<code>iim_ldap.userclass</code>	<code>inetOrgPerson,webtopuser</code>	The LDAP class indicates that an belongs to an en
<code>iim_ldap.groupclass</code>	<code>groupOfUniqueNames</code>	The LDAP class indicates that an belongs to a gro
<code>iim_ldap.groupbrowsefilter</code>	<code>(objectclass=groupofuniqueNames)</code>	The search filter browse all group directory under th specified search
<code>iim_ldap.searchlimit</code>	40	Maximum number entries to be returned a search. A value means search is on this server and a value of 0 indicates unlimited search
<code>iim_ldap.userdisplay</code>	<code>cn</code>	LDAP attribute to display name of users.
<code>iim_ldap.groupdisplay</code>	<code>cn</code>	LDAP attribute to display name of
<code>iim_ldap.useruidattr</code>	<code>uid</code>	LDAP attribute to end users' UID.
<code>iim_ldap.groupmemberattr</code>	<code>uniquemember</code>	LDAP attribute to the list of members group.
<code>iim_ldap.usermailattr</code>	<code>mail</code>	LDAP attribute to should contain email provisioned email addresses. Use the email message sent to an offline user.
<code>iim_ldap.user.attributes</code>	None	LDAP attribute to contains the list of custom attributes the LDAP user e
<code>iim_ldap.group.attributes</code>	None	LDAP attribute to contains the list of custom attributes the LDAP group
<code>iim_ldap.groupmemberurlattr</code>	None	The membership of a dynamic group which contains the filter or the LDAP

<code>iim_ldap.useidentityadmin</code>	The default value is <code>true</code> if you chose to leverage an Access Manager deployment for policy when you ran the <code>configure</code> utility. Otherwise, the default value is <code>false</code> .	If the value is <code>true</code> , the Access Manager Administrator credential will be used to bind to the Directory Server.
<code>iim.register.enable</code>	None	If <code>TRUE</code> , the server allows new Instant Messenger end users to register themselves (add themselves to the directory) using Instant Messenger.
<code>iim_ldap.register.basedn</code>	None	If self-registration is enabled, the value of this parameter is the location in the directory in which entries are stored. For example: <code>"ou=people,dc=sipro,dc=com"</code>
<code>iim_ldap.register.domain</code>	None	The domain to which new users will be added. For example, <code>directory.sipro.com</code> .

Logging Configuration Parameters

Table A-3 lists and describes the logging configuration parameters for both log4j-based logging and `iim.conf` parameter-based logging.

Table A-3 Logging Configuration Parameters

Parameter	Default Value	Description
<code>iim.log.iim_server.severity</code>	INFO	Level of logging required for the server module. The possible values from highest to lowest are: FATAL, ERROR, WARNING, INFO, and DEBUG. If a lower level of logging is chosen, it is implied that you get the higher levels too. For example, if you choose WARNING you get FATAL, ERROR, and WARNING.
<code>iim.log.iim_server.url</code>	<code>iim-runtime-base/log/xmppd.log</code>	Location of the server log file. This file needs to be periodically trimmed to prevent disk space from filling up.

<code>iim.log.iim_mux.severity</code>	INFO	Level of logging required for the multiplexor module. The possible values from highest to lowest are: FATAL, ERROR, WARNING, INFO, and DEBUG. If a lower level of logging is chosen, it is implied that you get the higher levels too. For example, if you choose WARNING you get FATAL, ERROR, and WARNING.
<code>iim.log.iim_mux.url</code>	<i>im-runtime-base</i> <i>/log/mux.log</i>	Location of the multiplexor log file. This file needs to be periodically trimmed to prevent disk space from filling up.
<code>iim.log.iim_mux.maxlogfiles</code>	10	The maximum number of log files to store for the multiplexor. Once this number is exceeded, the oldest multiplexor log file is deleted.
<code>iim.log.iim_mux.maxlogfilesize</code>	10 MB	This parameter contains the maximum size of a multiplexor log file. If the log files exceeds the size specified in this parameter then a new log file is created.
<code>iim.log.iim_server.maxlogsize</code>		This parameter contains the maximum size of a server log file. If the log files exceeds the size specified in this parameter then a new log file is created.
<code>iim.log.iim_wd.severity</code>	INFO	Level of logging required for the watchdog. The possible values from highest to lowest are: FATAL, ERROR, WARNING, INFO, and DEBUG. If a lower level of logging is chosen, it is implied that you get the higher levels too. For example, if you choose WARNING you get FATAL, ERROR, and WARNING.
<code>iim.log.agent-calendar.severity</code>	INFO	Level of logging required for the Calendar agent. The possible values from highest to lowest are: FATAL, ERROR, WARNING, INFO, and DEBUG. If a lower level of logging is chosen, it is implied that you get the higher levels too. For example, if you choose WARNING you get FATAL, ERROR, and WARNING.

<code>iim.log4j.config</code>	<code>im-cfg-base</code>	Specifies the location and name of the log4j configuration file. If no value exists for this parameter, the logger will look for <code>log4j.conf</code> in <code>im-cfg-base</code> . If the logger does not find <code>log4j.conf</code> in <code>im-cfg-base</code> , it uses the parameter-based logging method, instead of log4j.
-------------------------------	--------------------------	--

Instant Messaging Server Configuration Parameters

Table A-4 lists and describes the Instant Messaging server configuration parameters.

Table A-4 General Instant Messaging Server Configuration Parameters

Parameter	Default Value	Description
<code>iim_server.autosubscribe</code>	FALSE	Indicates whether subscriptions are a possible values are TRUE and FALSE followed by a subscribed response or modified roster to the subscriber and user and the contact must be in the s
<code>iim_server.domainname</code>	host's domain name	The logical Instant Messaging server is the name that is used by other servers the name used by this server to identify necessarily the Fully Qualified Domain Messaging server. For example, if the Messaging server for a company xyz.com.
<code>iim_server.port</code>	5269	IP address and port for the server to servers. IP address setting is useful for only one particular IP address. If no IP INADDR_ANY on localhost.
<code>iim_server.useport</code>	TRUE	Indicates whether the server should listen. The possible values are TRUE and FALSE defined by <code>iim_server.port</code> or on port
<code>iim_server.clienttimeout</code>	15	Specifies the time, in minutes, before longer active. For example, when a user is 5.
<code>iim_server.usesso</code>	The default value is 1, if you chose to leverage an Access Manager deployment for SSO when you ran the <code>configure</code> utility. Otherwise, the default value is 0.	This parameter tells the server whether authentication. An SSO provider is a server with a SSO service. The Access Manager server with the ability to validate session parameter can either be 0 or 1. Use 0 authentication even when the SSO provider parameter is used in conjunction with
<code>iim_server.ssoprovider</code>	None	Specifies the class implementing the interface. If <code>iim_server.usesso</code> is uses the default Access Manager-based

<code>iim.policy.modules</code>	The default value is <code>identity</code> , if you chose to leverage an Access Manager deployment for policy when you ran the <code>configure</code> utility. Otherwise, the default value is <code>iim_ldap</code> .	If the value is <code>identity</code> , indicates that policy storage. If the value is <code>iim_ldap</code>
<code>iim.userprops.store</code>	<code>file</code>	If the value is <code>file</code> , indicates that the file. If the value is <code>ldap</code> , directory is u
<code>iim_server.msg_archive</code>	<code>false</code>	This parameter specifies whether the Set this value to <code>false</code> to disable all archiving, including Portal, email, and
<code>iim_server.msg_archive.provider</code>	<code>None</code>	This parameter contains the list of ar values and each value is separated by Search based archive, the value should . If you are using email archiving, the <code>com.iplanet.im.server.Email</code>
<code>iim_server.conversion</code>	<code>false</code>	This parameter specifies whether me whether the configured list of Message message conversion.
<code>iim_server.conversion.provider</code>	<code>None</code>	This parameter contains the list of Message message conversion. This parameter separated by a comma (,).
<code>iim_server.conversion.external.command</code>	<code>None</code>	Contains the external command used <code>iim_server.conversion.external</code> default implementation of the Message <code>com.iplanet.im.server.External</code> invokes an external third party application class <code>com.iplanet.im.server.External</code> class, and set the parameter <code>iim_server.conversion.external</code> <code>%i %o</code> ", where <code>%i</code> and <code>%o</code> will automatically names generated dynamically by External conversion application is located at <code>/iim_server.conversion.external</code> <code>%i %o</code> .
<code>iim_server.servertimeout</code>	<code>-1</code>	The server can be configured to auto server, if the remote server is inactive time the last request was made by the the remote server is terminated, if the server exceeds the value of the <code>iim_s</code> value is in minutes.
<code>iim_server.enable</code>	<code>true</code>	This value defines whether or not the parameter is set false to enable the li
<code>iim_server.stat_frequency</code>	<code>1</code>	This parameter contains the frequency activities to the log file. The server log the server minimum log severity is se

<code>iim_server.certnickname</code>	Server-Cert	This value should contain the name of the certificate. The certificate name is case sensitive.
<code>iim_server.sslkeystore</code>	None	Contains the relative path and filename of the keystore. For example, <code>/im-cfg-base/server-keystore.jks</code> .
<code>iim_server.keystorepasswordfile</code>	<code>sslpassword.conf</code>	This value should contain the relative path and filename of the password file for the key database. This file is used by the (Software) Token: <code>password_Wallet</code> database.
<code>iim_server.requiresssl</code>	false	If true, the server will terminate any connections after the initial stream session is set up on all servers, and server components, such as the multiplexor.
<code>iim_server.trust_all_cert</code>	false	If this value is true then the server will trust all certificate information into the log files.
<code>iim_server.deliverofflinechat</code>	false	Determines whether the capability is supported. For deployment, set the <code>iim_server.deliverofflinechat</code> parameter to true. For non-deployment, set the <code>iim_server.deliverofflinechat</code> parameter to false. Do not set <code>deliverofflinechat.domain</code> .
<code>deliverofflinechat.domain</code>	None	This parameter is used to blacklist or whitelist domains. For deployment, set the <code>deliverofflinechat.domain</code> parameter to true. For non-deployment, set the <code>deliverofflinechat.domain</code> parameter to false, and set the <code>deliverofflinechat.domain</code> parameter to the list of domains to be whitelisted.
<code>deliverofflinechat.maxsize</code>	50	This parameter is used to determine the maximum size of the chat history and must be a positive integer.

Multiple Server Configuration Parameters

For communication between multiple Instant Messaging servers in your network, you need to configure your server to identify itself with the other servers and identify itself with each coserver, or cooperating server, which will have a connection to your server. The coserver identifies itself with its Instant Messaging domain name, host and port number, server ID, and password.

Each cooperating server is given a symbolic name, which is a string consisting of letters and digits, for example, `coserver1`. Using the symbolic naming convention you can specify multiple servers.

When Instant Messaging servers are configured in this manner, you can form a larger Instant Messaging community. Therefore, end users on each server can do the following:

- Communicate with end users on every other server
- Use conferences rooms on other servers
- Subscribe to news channels on other servers (subject to access privileges)

Table A-5 lists and describes the multiple server configuration parameters.

Table A-5 Multiple Server Configuration Parameters

Parameter	Default Value	Description
-----------	---------------	-------------

<code>iim_server.serverid</code>	None	String used by this server to identify itself to all o servers.
<code>iim_server.password</code>	None	Password used by this server to authenticate its all other servers.
<code>iim_server.coservers</code>	None	Comma separated list containing symbolic name the servers that can connect to this server. Any meaningful names are allowed, they must match what you use for the <code>*.serverid</code> , <code>*.password</code> , and <code>*.host</code> parameters. Examples: <code>iim_server.coservers=coserver1,coser</code> <code>or iim_server.coservers=abc,xyz,ntc</code>
<code>iim_server.{coserver1}{.serverid}</code>	None	String that identifies the cooperating server represented by the name, <code>coserver1</code> to authenticate to this server. For example, if you used <code>abc</code> in the <code>iim_server.coservers</code> list, then the corresponding name for its <i>serverid</i> would be <code>iim_server.abc.server</code>
<code>iim_server.{coserver1}{.password}</code>	None	Password used by cooperating server represent the name, <code>coserver1</code> to authenticate to this server. F example, if you used <code>abc</code> in the <code>iim_server.coservers</code> list, t the corresponding name for its password would l <code>iim_server.abc.password</code> .
<code>iim_server.{coserver1}{.host}</code>	None	IP address and the port to connect to, for end users on this server to communicate to e users on the server represented by the name <code>coserver1</code> . For example, if you us <code>abc</code> in the <code>iim_server.coservers</code> list, then the corresponding name for its host wo <code>iim_server.abc.host</code> . The format is <i>name:port</i> or <i>IPAddress:port</i> .
<code>iim_server.{coserver1}{.requiresssl}</code>	False	Indicates if this server should require TLS when communicating with the server identified b <code>coserver1</code> . The possible values are TRUE and FALSE. Note : This parameter can be used only with Ser Server Communication and Static ServerPool Configurations. It cannot be used with Shoal configuration since the co-servers are discovere the fly with Shoal.
<code>iim_server.coservers.requiresssl</code>	False	Indicates if this server should require TLS when communicating with the co-servers. The possible values are TRUE and FALSE.

<code>iim_server.openfederation.enabled</code>	None	<p>To enable open federation, set the parameter to true. To disable open federation, set the parameter to false. For example:</p> <pre>iim_server.openfederation.enabled = true</pre> <p>By default, open federation is not enabled. If the parameter is not in <code>iim.conf</code>, add it. Once configured, the server's users will be able to communicate with users on any other Instant Messaging server that is likewise configured.</p>
--	------	--

Shoal Configuration Parameters

You can use the Shoal clustering framework to automatically discover and add peer servers in a server pool.

[Table A-6](#) lists and describes the shoal configuration parameters.

Table A-6 Shoal Configuration Parameters

Parameter	Default	Required?	Description
<code>iim_server.peer.autodiscover</code>	false	No	The parameter that enables auto-discovery using Shoal. It is recommended to delete all static co-server definitions before setting this to true.
<code>iim_server.serverid</code>	None	Yes	The ID that uniquely identifies the server instance within the pool. It could be an identifier such as <code>server1</code> , or a host name.
<code>iim_server.password</code>	None	Yes	The password that is shared across the pool and enables identification of members of one pool from the other. Also ensures that unidentified members of a Shoal group can not join the pool.
<code>iim_server.hostname</code>	<code>local-hostname:5269</code>	No	The connection string that the other pool members can establish connections with. It is the host name and port of the specified server.
<code>iim_server.pool.groupname</code>	<code>iim.server.pool</code>	No	The Shoal group name that the peers will attempt to join. You will need to change the default only if multiple clusters of peer servers need to run on the same subnet.

For Shoal Across Subnets

Parameter	Default	Required?	Description
<code>relay.imadmin.enable</code>	true	Yes	Starts the relay server.
<code>relay.listen_address=<address of relay server></code>	None	Optional	Specifies the address of the relay server.
<code>relay.uri_list</code>	None	Yes	Displays the list of relay servers added.

Multiplexor Configuration Parameters

Table A-7 lists and describes the multiplexor configuration parameters.

Table A-7 Multiplexor Configuration Parameters

Parameter	Default Value	Description
<code>iim_mux.listenport</code>	<i>multiplexorname or IP address:5222</i>	IP address or FQDN and listening port on which the multiplexor listens for incoming requests from Instant Messenger. The value format is <i>IPaddress:port</i> or <i>multiplexorname:port</i> . If no IP address or domain name is listed, <code>INADDR_ANY</code> on <code>localhost</code> is assumed.If you change this value, also change the <code>im.html</code> and <code>im.jnlp</code> files so that they match the port value.
<code>iim_mux.serverport</code>	45222	The Instant Messaging server and port the multiplexor communicates to. The value format is <i>servername:port</i> or <i>IPaddress:port</i> .
<code>iim_mux.numinstances</code>	1	Number of instances of the multiplexor. This parameter is valid only for Solaris platforms.
<code>iim_mux.maxthreads</code>	5	Maximum number of threads per instance of the multiplexor.
<code>iim_mux.maxsessions</code>	2000	Maximum number of concurrent connections per multiplexor process.
<code>iim_mux.usessl</code>	off	If the value is set to <code>on</code> , the multiplexor requires an SSL handshake for each connection it accepts, before exchanging any application data.
<code>iim_mux.seconfigdir</code>	<code>/etc/opt/SUNWiim/default/config</code>	This directory contains the key and certificate databases. In addition, it also usually contains the security module database. The name of the <code>/default</code> directory may vary if you created multiple instances of Instant Messaging.

<code>iim_mux.keydbprefix</code>	None	This value should contain the key database filename prefix. The key database file name must always end with <code>key3.db</code> . If the Key database contains a prefix, for example <code>This-Database-key3.db</code> , then value of this parameter is <code>This-Database</code> .
<code>iim_mux.certdbprefix</code>	None	This value should contain the certificate database filename prefix. The certificate database file name must always end with <code>cert7.db</code> . If the certificate database contains a prefix, for example <code>Secret-stuff-cert7.db</code> , then value of this parameter is <code>Secret-stuff</code> .
<code>iim_mux.secmofile</code>	<code>secmod.db</code>	This value should contain the name of the security module file.
<code>iim_mux.certnickname</code>	<code>Multiplexor-Cert</code>	This value should contain the name of the certificate you entered while installing the certificate. The certificate name is case-sensitive.
<code>iim_mux.keystorepasswordfile</code>	<code>/etc/opt/SUNWiim/default/config/sslpassword.conf</code>	This value should contain the relative path and the name of the file containing the password for the key database. This file should contain the following line: Internal (Software) Token:password Where <i>password</i> is the password protecting the key database. The name of the <code>/default</code> directory may vary if you created multiple instances of Instant Messaging.
<code>iim_mux.stat_frequency</code>	600	This value should contain the frequency at which the multiplexor logs the summary of activities to the log file. The minimum value is 10 seconds.
<code>iim_mux.enable</code>	<code>true</code>	If the value is <code>true</code> then the multiplexor will run for this instance. If the value is <code>false</code> then the multiplexor will not run for this instance.

Redirect Server Parameters

Table A-8 lists the parameters you use to administer the Instant Messaging redirect server.

Table A-8 Redirect Server Parameters

Parameter	Default Value	De
<code>iim_server.redirect.provider</code>	None	Co prc im co inte def ser ro tha co inte
<code>iim_server.redirect.to</code>	None	Co this cor alp sup <i>iim</i>
<code>iim_server.redirect.to.nodename.host</code>	None	Wh noc .TL to l
<code>iim_server.redirect.to.nodename.usessl</code>	False	If tr use TL Me
<code>iim_server.redirect.db.users</code>	<code>im-db-base/redirect.db</code>	Na dat
<code>iim_server.redirect.db.partitions</code>	<code>im-cfg-base/redirect.partitions</code>	Na par
<code>iim_server.redirect.db.partitionsize</code>	5000	The par
<code>iim_server.redirect.roundrobin.partitions</code>	<code>im-cfg-base/redirect.partitions</code>	Na par
<code>iim_server.redirect.pollfrequency</code>		The by def The det acc

Archive Parameters

Table A-9 lists the parameters you use to manage Instant Messaging archiving.

Table A-9 Archive Parameters

Parameter	Default Value	Description
<code>iim_arch.title.attr</code>	Title	This parameter contains the name of the field equivalent to the <code>Title</code> field in the default schema of the Portal Server Search.
<code>iim_arch.keyword.attr</code>	Keyword	This parameter contains the name of the field equivalent to the <code>Keyword</code> field in the default schema of the Portal Server Search.
<code>iim_arch.readacl.attr</code>	ReadACL	This parameter contains the name of the field equivalent to the <code>ReadACL</code> field in the default schema of the Portal Server Search.
<code>iim_arch.description.attr</code>	Description	This parameter contains the name of the field equivalent to the <code>Description</code> field in the default schema of the Portal Server Search.
<code>iim_arch.fulltext.attr</code>	Full-Text	This parameter contains the name of the field equivalent to the <code>Full-Text</code> field in the default schema of the Portal Server Search.
<code>iim_arch.category.attr</code>	Category	This parameter contains the name of the field equivalent to the <code>Category</code> field in the default schema of the Portal Server Search.
<code>iim_arch.readacl.admin</code>	None	This parameter contains the administrator DN. Multiple values should be separated by a semi colon (;).
<code>iim_arch.readacl.adminonly</code>	false	This parameter will contain <code>true</code> or <code>false</code> . <code>true</code> - Only the administrator's DN specified by the parameter <code>iim_arch.readacl.admin</code> will be added to the <code>ReadACL</code> field overwriting default behavior of the <code>ReadACL</code> field. <code>false</code> - The administrator's DN specified by the parameter <code>iim_arch.readacl.admin</code> will be added to the <code>ReadACL</code> field in addition to the default behavior.
<code>iim_arch.categories</code>	all	This parameter contains a list of message types that can be archived. The value can be: <code>pollalertchatconference</code> <code>{{news}}</code> Multiple values can be specified separated by commas (,).
<code>iim_arch.categoryname</code>	None	If a category name is not assigned for any of the categories then the value of this parameter is used as the category name.

<code>iim_arch.alert.categoryname</code>	None	This parameter contains the name of the category containing the archived alert messages. It is not required to dedicate a category to alert messages.
<code>iim_arch.poll.categoryname</code>	None	This parameter contains the name of the category containing the archived poll messages. It is not required to dedicate a category to poll messages.
<code>iim_arch.conference.categoryname</code>	None	This parameter contains the name of the category containing the archived conference messages. It is not required to dedicate a category to conference messages.
<code>iim_arch.chat.categoryname</code>	Name	This parameter contains the name of the category containing the archived chat messages. It is not required to dedicate a category to chat messages.
<code>iim_arch.news.categoryname</code>	None	This parameter contains the name of the category containing the archived news messages. It is not required to dedicate a category to news messages.
<code>iim_arch.conference.quiettime</code>	5	This parameter contains the maximum duration of silence between two consecutive messages in a room (both public and private) after which the RD expires and a new RD is created for archiving the message. The value is in minutes.
<code>iim_arch.poll.maxwaittime</code>	15	This parameter contains the (maximum) time for which poll data is buffered in the server. The value is in minutes.
<code>iim_arch.ignoreexplicitdeny</code>	true	This parameter will contain <code>true</code> or <code>false</code> . <code>true</code> - For Poll and Conference categories, the data with explicit deny access will not be archived. Each time when these messages are not archived this information will be logged into the <code>xmppd.log</code> file. <code>false</code> - For Poll and Conference categories, the data with explicit deny access will not be archived and the message will be added to the Portal Server Search database. Note: If you do not explicitly deny access to a room or a news channel the default access is either <code>READ</code> or <code>WRITE</code> or <code>MANAGE</code> . Some end users can also be granted <code>NONE</code> access.

<code>iim_arch.portal.search</code>	None	The value of the this parameter should be the URL of the Portal Server Search servlet. For example: http://www.example.com/portal/search . This parameter is not present then the Archive Provider determines the value of the Portal Server Search URL based on the <code>AMConfig.properties</code> file present on the system.
<code>iim_arch.portal.adminDN</code>	None	The value of this parameter should be the DN of the admin user. For example: <code>uid=amadmin,ou=People,o=inter</code> . This parameter is required when the Document level Security in the Portal Server Server is on.
<code>iim_arch.portal.adminpassword</code>	None	The value of this parameter should be the password of the administrative user as specified by the <code>iim_arch.portal.adminDN</code> parameter. This parameter is required when the Document level Security in the Portal Search Server is on.
<code>iim_arch.portal.search.database</code>	None	The value of this parameter should be the name of the database where the Instant Messaging server stores archived messages. If this parameter is not defined then all messages are stored in the default database of Portal Server Search.
<code>iim_arch.admin.email</code>	Empty String	Comma-separated list of administrator email addresses.
<code>iim_arch.alert.admin.email</code>	None	Comma-separated list of administrator email addresses to which all archived alert messages will be sent. This parameter overrides <code>iim_arch.admin.email</code> for alert messages.
<code>iim_arch.chat.admin.email</code>	None	Comma-separated list of administrator email addresses to which all archived chat messages will be sent. This parameter overrides <code>iim_arch.admin.email</code> for chat messages.
<code>iim_arch.conference.admin.email</code>	None	Comma-separated list of administrator email addresses to which all archived conference messages will be sent. This parameter overrides <code>iim_arch.admin.email</code> for conference messages.
<code>iim_arch.poll.admin.email</code>	None	Comma-separated list of administrator email addresses to which all archived poll messages will be sent. This parameter overrides <code>iim_arch.admin.email</code> for poll messages.

<code>iim_arch.news.admin.email</code>	None	Comma-separated list of administrator email addresses to which all archived messages will be sent. This parameter overrides <code>iim_arch.admin.email</code> for new messages.
<code>iim_arch.email.archiveheader.name</code>	None	Name of the extended RFC 822 header
<code>iim_arch.email.archiveheader.value</code>	all	Value corresponding to the header name for <code>iim_arch.email.archiveheader.name</code> .

Watchdog Parameters

The watchdog monitors the server process and attempts to restart the server if it determines that the server is not running. See [Managing the Watchdog Process](#)

Table A-10 lists and describes the watchdog configuration parameters.

Table A-10 Watchdog Configuration Parameters

Parameter	Default Value	Description
<code>iim_wd.enable</code>	<code>true</code>	Enables the watchdog feature. To reset this parameter or disable the watchdog, set this to <code>false</code> . To avoid conflicts, you should disable the watchdog if you are monitoring the Instant Messaging server using the operating system administration console.
<code>iim_wd.period</code>	300 (seconds)	The watchdog periodically polls the server to check whether it is running. This parameter sets the interval between two status polls.
<code>iim_wd.maxRetries</code>	3	Sets the number of retries, times the watchdog will attempt to contact the Instant Messaging server, before shutting down and restarting the server. The maximum is ten retries.

Monitoring Parameters

The parameter in Table A-11 configures how the server interacts with the Sun Java Enterprise System Monitoring Framework.

Table A-11 Monitoring Parameters

Parameter	Default Value	Description
<code>iim_server.monitor.enable</code>	<code>false</code>	Used by the Sun Java Enterprise System Monitoring Framework. If <code>true</code> , configures the server to make its activities available to <code>mfwk</code> . Otherwise, the server does not make its activities available.
<code>iim_server.monitor.htmlport</code>	None	If specified, opens the JMX HTML adaptor port on the specified port. By default, this port is not enabled as opening this port can present a security risk.

Agent Parameters

Agents, such as the Calendar agent, enable functionality within the Instant Messaging server and enhance its interoperability with other Sun Java System servers.

Table A-12 lists and describes agent configuration parameters.

Table A-12 Agent Configuration Parameters

Parameter	Default Value	Description
<code>jms.consumers</code>	None	Used with the Calendar agent. Contains the list of JMS consumer names. The value for this parameter must be set to a list of consumer names.
<code>jms.consumer.cal_reminder.destination</code>	None	Used with the Calendar agent. Destination name for calendar reminder messages. Must be the same as the value of the <code>calendar.destination</code> configuration parameter in the <code>ics.conf</code> file. Example: <code>enp:///ics/customalarm</code>
<code>jms.consumer.cal_reminder.provider</code>	None	Used with the Calendar agent. The name of the JMS provider. Typically, this is set to <code>ens</code> . The value for this parameter must be the same as the name in the <code>jms.providers</code> parameter.
<code>jms.consumer.cal_reminder.type</code>	None	Used with the Calendar agent. The type of calendar reminder. The value for this parameter must be set to <code>topic</code> .
<code>jms.consumer.cal_reminder.param</code>	None	Used with the Calendar agent. The alarm parameters for this parameter must be set as follows in the <code>ics.conf</code> file: <code>eventtype=calendar.alarm</code>
<code>jms.consumer.cal_reminder.factory</code>	None	Used with the Calendar agent. A listener that processes the new calendar reminder messages. The value for this parameter must be set to: <code>com.ipplanet.im.server.JMSCalendarReminderFactory</code>
<code>jms.providers</code>	None	Used with the Calendar agent. The name of the JMS provider. Typically, you set the value of this parameter to be the same as the value for the <code>jms.consumer.cal_reminder.provider</code> parameter.
<code>jms.provider.ens.broker</code>	None	Used with the Calendar agent. Hostname and port number on which the ENS listens for incoming messages to the port specified in the <code>ics.conf</code> file's <code>calendar.destination</code> parameter. The default is 57997. For example: <code>jms.provider.ens.broker=cal.example.com:57997</code>
<code>jms.provider.ens.factory</code>	None	Used with the Calendar agent. Factory class that creates the topic connection objects. The value for this parameter must be set as follows. Enter the value on a single line: <code>com.ipplanet.ens.jms.EnsTopicConnectionFactory</code>
<code>iim_agent.enable</code>	False	If <code>TRUE</code> , <code>iim.conf</code> , enables Instant Messaging. If <code>FALSE</code> , or remove the parameter from the file, disables all agents.
<code>iim_agent.agent-calendar.enable</code>	None	Used with the Calendar agent. If <code>TRUE</code> or a non-empty value, loads a component that enables the Calendar agent specifically.
<code>agent-calendar.jid</code>	None	The JID of the Calendar agent.

<code>agent-calendar.password</code>	None	Defines the password with which the Calendar agent communicates with the Instant Messaging server.
<code>iim_server.components</code>	None	Describes the Calendar agent as a component of the Instant Messaging server. The value of this parameter is <code>agent-calendar</code> .
<code>agent-calendar.imadmin.enable</code>	<code>false</code>	Start the agent-calendar by using the <code>imadmin</code> command. Set to <code>true</code> .
<code>agent-calendar.iim_server.host</code>		Host name of the Instant Messaging server with which the calendar agent communicates.
<code>agent-calendar.iim_server.port</code>		Port number of the Instant Messaging server with which the calendar agent communicates.

HTTP/XMPP Gateway Parameters

Table A-13 lists the parameters you use to bind to the HTTP/XMPP gateway.

Table A-13 HTTP/XMPP Gateway Parameters

Parameter	Default Value	Description
<code>iim_agent.httpbind.enable</code>	<code>false</code>	Set to <code>true</code> to enable the HTTP/XMPP gateway.
<code>httpbind.jid</code>		A jabber ID (JID) to bind the HTTP/XMPP gateway.
<code>httpbind.password</code>		Password to authenticate the HTTP/XMPP gateway to the Instant Messaging server.

SMS Integration Parameters

Table A-14 lists the SMS integration parameters.

Table A-14 SMS Integration Parameters

Parameter	Default Value	Description
<code>smsgw.imadmin.enable</code>	<code>false</code>	Enables or disables the SMS gateway. If set to <code>true</code> , you can start the SMS gateway by using the <code>imadmin</code> command.
<code>smsgw.jid</code>	None	A jabber ID (JID) to bind the SMS gateway to the Instant Messaging server. The value of this parameter should be the same as the value that you define for the <code>smpplib.jid</code> parameter.
<code>smsgw.password</code>		Password to authenticate the SMS gateway to the Instant Messaging server. The value of this parameter should be the same as the value that you define for the <code>smpplib.password</code> parameter.
<code>smsgw.iim_server</code>	None	Hostname and port number of the Instant Messaging server.
<code>smsgw.sms_limit</code>	-1	Number of messages that can be sent per hour. The default value is -1 and it indicates that unlimited number of SMS messages that can be sent per hour.
<code>smsgw.sms_queue_capacity</code>	512	Maximum number of messages that can be queued for SMS delivery.
<code>smsgw.im_char_limit</code>	500	Maximum number of characters that you can specify in one message. If the number of characters is greater than the specified value, the message is rejected.
<code>smpplib.smsc_ip_address</code>	None	IP address or hostname of the SMSC.
<code>smpplib.smsc_port</code>	2775	Port number of the SMSC.
<code>smpplib.bind_id</code>	None	Identifier used to bind the SMS gateway to the SMSC.
<code>smpplib.bind_password</code>		Password to authenticate the SMS gateway to the SMSC.
<code>smpplib.sender_id</code>	None	Sender ID of the outgoing SMS.
<code>iim_server.components</code>	None	List of component identifiers that should have <code>smpplib</code> . For example, <code>httpbind</code> , <code>smpplib</code> .
<code>iim_agent.smpplib.enable</code>	<code>false</code>	Enables the Instant Messaging server to identify the SMS gateway.
<code>smpplib.jid</code>	None	A jabber ID (JID) for binding the SMS gateway to the Instant Messaging server.
<code>smpplib.password</code>		Password to authenticate the SMS gateway to the Instant Messaging server.

MSN Gateway Integration Parameters

As of Instant Messaging 9.0.1.4.0, the MSN gateway is deprecated: it may be removed in a future release.

Table A-15 lists the MSN gateway integration parameters.

Table A-15 MSN Gateway Integration Parameters

Parameter	Default Value	Description
<code>iim_agent.msn_gateway.enable</code>	<code>false</code>	Set to <code>true</code> to enable the MSN gateway on IM server.
<code>msn_gateway.jid</code>	<code>msn.\$domainname</code>	A jabber ID (JID) for binding the MSN gateway to the Instant Messaging server.
<code>msn_gateway.password</code>	<code>random</code>	Password to authenticate the MSN gateway to the Instant Messaging server.
<code>msn_gateway.imadmin.enable</code>	<code>false</code>	If set to <code>true</code> , you can start the MSN gateway by using the <code>imadmin start</code> command.

AIM Gateway Integration Parameters

As of Instant Messaging 9.0.1.4.0, the AIM gateway is deprecated; it may be removed in a future release.

Table A-16 lists the AIM gateway integration parameters.

Table A-16 AIM Gateway Integration Parameters

Parameter	Default Value	Description
<code>iim_agent.aim_gateway.enable</code>	<code>false</code>	Set to <code>true</code> to enable the AIM gateway on IM server.
<code>aim_gateway.jid</code>	<code>aim.\$domainname</code>	A jabber ID (JID) for binding the AIM gateway to the Instant Messaging server.
<code>aim_gateway.password</code>	<code>random</code>	Password to authenticate the AIM gateway to the Instant Messaging server.
<code>aim_gateway.imadmin.enable</code>	<code>false</code>	If set to <code>true</code> , you can start the AIM gateway by using the <code>imadmin start</code> command.

Yahoo Gateway Integration Parameters

As of Instant Messaging 9.0.1.4.0, the Yahoo gateway is deprecated; it may be removed in a future release.

Table A-17 lists the Yahoo gateway parameters.

Table A-17 Yahoo Gateway Parameters

Parameter	Default Value	Description
<code>iim_agent.yim_gateway.enable</code>	<code>false</code>	Set to <code>true</code> to enable the Yahoo gateway on IM server.
<code>yim_gateway.jid</code>	<code>yim.\$domainname</code>	A jabber ID (JID) for binding the Yahoo gateway to the Instant Messaging server.
<code>yim_gateway.password</code>	<i>random</i>	Password to authenticate the Yahoo gateway to the Instant Messaging server.
<code>yim_gateway.imadmin.enable</code>	<code>false</code>	If set to <code>true</code> , you can start the Yahoo gateway by using the <code>imadmin start</code> command.

IMPS Gateway Parameters

Table A-18 lists the IMPS gateway parameters.

Table A-18 IMPS Gateway Parameters

Parameter	Default Value
<code>iim_ldap.sasl.mechanism.factories</code>	<code>"com.ipplanet.im.server.sasl.IMPSSASLProvide</code>
<code>iim_ldap.userpasswordattr</code>	<code>"userpassword"</code>

Java Message Queue (JMQ) Parameters

Table A-19a lists the JMQ parameters

This table lists some of the Java Message Queue (JMQ) parameters that need to be specified.

Table A-19a JMQ Parameters

Parameter	Value (including the quotation mark)	Description
JMS Consumers section		
<code>jms.consumers</code>	<code>"cal_reminder2"</code>	Name of the alarm. By default, <code>jms.consumers</code> is commented out in the <code>iim.conf</code> file. Make sure to uncomment this line.

<code>jms.consumer.cal_reminder2.provider</code>	<code>"jmq"</code>	Name of the provider. Java Message Queue should be mentioned as a provider. The string <code>jmq</code> is used in the agent code to instantiate the Java Message Queue specific classes.
<code>jms.consumer.cal_reminder2.type</code>	<code>"topic"</code>	The type of the alarm to set.
<code>jms.consumer.cal_reminder2.factory</code>	<code>"com.iplanet.im.server.JMSCalendarMessageListener"</code>	The name of the C++ factory.
<code>jms.consumer.cal_reminder2.destination</code>	<code>"testTopic"</code>	Destination of the alarm. The destination type is <code>topic</code> . The <code>topic</code> can be administratively created or the Java Message Queue provider can be configured to create a <code>topic</code> for publishing a message to it. When the Calendar agent starts, it tries to subscribe to the configured <code>topic</code> . If the <code>topic</code> is not already present in the Java Message Queue broker, the Calendar agent fails to subscribe to the <code>topic</code> . Therefore, it is necessary to start the Java Message Queue broker and create the <code>topic</code> before starting the Calendar agent. For the commands to start the Java Message Queue broker and create a <code>topic</code> , see Some Useful Java Message Queue Commands .
JMS Providers section		
<code>jms.providers</code>	<code>"jmq"</code>	The name of the provider.
<code>jms.provider.jmq.broker</code>	<code>"yourJMQserver:port"</code>	Port number that the Java Message Queue server listens to. <code>jms.provider.jmq.broker</code> should be the fully qualified host name or IP address and port that your Java Message Queue server is listening on. For example, <code>localhost:7676</code> or <code>jmqhost.beta.comms.com</code>
<code>jms.provider.jmq.factory</code>	<code>"com.sun.messaging.TopicConnectionFactory"</code>	Name of the C++ factory.

jms.provider. jmq.jmsuser	"guest"	The user ID of the JMS user. A Java Message Queue user is created in the Java Message Queue provider user database. If access to the Java Message Queue provider or the topic is controlled, specify the username. The username is provided by the Calendar agent while establishing connection with the Java Message Queue provider. If this parameter is not specified, the agent tries to connect anonymously.
jms.provider. jmq.jmspswd	"passwd"	The password of the JMS user.
iim_agent.enable	"true"	Enables agents for Instant Messaging. Set the value to <code>iim_agent.enable="true"</code> .
iim_agent. agent-calendar.enable	"true"	Loads a component that enables the Calendar agent. Set the value to <code>iim_agent. agent-calendar.enable="true"</code> .
agent-calendar.jid	calendar.siroe.com	Java ID of the Calendar agent. Set the value to <code>agent-calendar. jid=calimbot.server.domain</code> .
agent-calendar.password		Password you want the Calendar agent to use to connect to the Instant Messaging server. Set the value to <code>agent-calendar. password=password</code> .
iim_server.components	agent-calendar,httpbind	Set the value to <code>iim_server.components =agent-calendar</code> .
agent-calendar. imadmin.enable	"false"	Start the agent-calendar by using the <code>imadmin</code> command if set to <code>true</code> .
agent-calendar. iim_server.host		Host name of the Instant Messaging server with which the agent calendar communicates.
agent-calendar. iim_server.port		Port number of the Instant Messaging server with which the agent calendar communicates.

Conference History Parameters

Table A-20 Conference History Parameters

Parameter	Default Value	Description
iim_server. conference.history.maxstanzas	10	Maximum number of stanzas persisted by the server for a given conference.
iim_server. conference.history.maxstanzas.default	0	Number of stanzas sent to joining user.
iim_server. conference.history.persist	true	Enables conference history and stores it on persistent storage.

Chapter 39. Instant Messaging XMPP and HTTP Gateway Configuration Parameters in `httpbind.conf`

Oracle Communications Instant Messaging Server XMPP/HTTP Gateway Configuration Parameters in `httpbind.conf`

Topics:

- [httpbind.conf File Location](#)
- [httpbind.conf File Syntax](#)
- [How Load Balancing Occurs](#)
- [Instant Messaging XMPP/HTTP Gateway Configuration Parameters](#)
- [Gateway Domain ID Key Parameters for httpbind.config](#)

Any time you modify the `httpbind.conf` file, you need to restart the XMPP/HTTP Gateway by using the tools provided by your web container.

`httpbind.conf` File Location

By default, the `configure` utility creates the `httpbind.conf` file within the configuration directory (`im-cfg-base`) of the default server instance, for example:

- Solaris OS:
`/etc/opt/SUNWim/default/config/httpbind.conf`
- Red Hat Linux:
`/etc/opt/sun/im/default/config/httpbind.conf`

If you create multiple instances of Instant Messaging, the name of the `/default` directory varies depending on the instance. See [Creating Multiple Instances from a Single Instant Messaging Installation](#) for more information. This file is created by the `configure` utility only in the default instance's `im-cfg-base` directory.

`httpbind.conf` File Syntax

The `httpbind.conf` file is a plain ASCII text file, with each line defining a gateway parameter and its value(s):

- A parameter and its value(s) are separated by an equal sign (`=`) with spaces and tabs allowed before or after the equal sign.
- A value can be enclosed in double quotes (`" "`). If a parameter allows multiple values, the entire value string must be enclosed in double quotes.
- A comment line must have an exclamation point (`!`) as the first character of the line. Comment lines are for informational purposes and are ignored by the server.
- If a parameter appears more than once, the value of the last parameter listed overrides the previous value.
- A backslash (`\`) is used for continuation and indicates the value(s) are longer than one line.
- Each line is terminated by a line terminator (`\n`, `\r`, or `\r\n`).

- The key consists of all the characters in the line starting with the first non-whitespace character and up to the first ASCII equal sign (=) or semi-colon (;). If the key is terminated by a semi-colon, it is followed by "lang=" and a tag that indicates the language in which this value is to be interpreted. The language tag is followed by an equal sign (=). All whitespace characters before and after the equal sign are ignored. All remaining characters on the line become part of the associated value string.
- Multiple values in the value string are separated using commas (,).
- Within a value, if any special characters like comma, space, newline, tab, double quotes, or backslash are present, the entire value needs to be within double quotes. In addition, every carriage return, line feed, tab, backslash, and double quotes within the value must be specified with a backslash (\).
- If you make changes to `httpbind.conf`, you must refresh the gateway's web container in order for the new configuration settings to take effect.



Note

The `httpbind.conf` file is initialized by the `configure` utility and should be modified only as described in this information.

How Load Balancing Occurs

HTTPBIND performs round-robin load balancing among the component sessions (connections from HTTPBIND to a back end) in a circular linked-list fashion to decide which back end is used.

A change in connection status is reflected almost immediately (as soon as a `StreamStatusChanged` event occurs). Thus, if a disconnection happens for a particular back end, it is out of the list. When the connection resumes, it comes back to the available list.

Instant Messaging XMPP/HTTP Gateway Configuration Parameters

Table B-1 describes the configuration parameters in `httpbind.conf`.

Table B-1 XMPP/HTTP Gateway Configuration Parameters in `httpbind.conf`

Parameter	Default Value	Description
<code>httpbind.pool.nodeId</code>	N/A	If <code>httpbind.pool.support</code> is set to <code>true</code> , this parameter specifies the full URL for the server node in the server pool. This URL should not point to a load balancer, but to an Instant Messaging server instance.

<code>httpbind.pool.support</code>	<code>false</code>	<p>This parameter defines whether or not the gateway is in a server pool deployment. If no <code>httpbind.pool.nodeld</code> is specified, the value for this parameter is set to <code>false</code>. The value for this parameter can be:</p> <ul style="list-style-type: none"> <code>true</code> - the gateway is part of a server pool deployment. In addition, <code>enable</code>, <code>on</code>, <code>yes</code>, and <code>1</code> are also valid values. If you set this parameter to <code>true</code>, you must provide a value for <code>httpbind.pool.nodeld</code>. <code>false</code> - (default) the gateway is not part of a server pool deployment. Leaving the value blank (empty string) is also a valid value.
<code>httpbind.config</code>	N/A	Contains a comma-separated list of ID keys, or <i>gwdomain-id</i> , which the gateway uses as a configuration key to determine which domains, hosts, host passwords, and component JIDs the gateway should use. See Table B-2 for more information on ID keys.
<code>httpbind.content_type</code>	<code>text/xml; charset=utf-8</code>	The default value for the <code>content-type</code> HTTP header the gateway uses when sending a response back to the client.
<code>httpbind.hold</code>	N/A	Specifies the maximum permissible value for the <i>hold</i> attribute in the client request as defined in http://www.jabber.org/jeps/jep-0124.html . If the client specifies a value higher than the gateway in the request, the gateway's value will be used. Otherwise, the value in the client request will be used.
<code>httpbind.inactivity</code>	180	The maximum time in seconds of client inactivity after which the gateway will terminate the connection to the client.
<code>httpbind.log4j.config</code>	N/A	The location of the <code>log4j</code> configuration file the gateway will use for logging. If you leave this parameter blank, then logging for the gateway is turned off. The logger name is "httpbind" (<code>log4j.logger.httpbind</code>).
<code>httpbind.polling</code>	1 (second)	The minimum time, in seconds, a client must wait before sending another request.
<code>httpbind.requests</code>	2	The number of concurrent requests a client can make to the gateway. If the value of this parameter is less than the value for the JEP 124 <i>hold</i> attribute in the client request, the value for this parameter will be set to <code>hold+1</code> . Do not set this parameter to 1, as doing so could severely degrade performance. See httpbind.hold for more information.

<code>httpbind.round_trip_delay</code>	1 (second)	The amount of time, in seconds, to allow in addition to time-outs for round trips to account for network latencies. Setting this value too high may degrade performance.
<code>httpbind.wait_time</code>	120 (seconds)	The default time, in seconds, within which the gateway will send a response to the client. If the client wait time is set to a value higher than the gateway wait time, the gateway's wait time is used.

Gateway Domain ID Key Parameters for `httpbind.config`

Table B-2 describes the keys used to define each ID in the `httpbind.config_` parameter. In each key described in the table, `gwdomain-id` is a domain identifier specified in `httpbind.config`.

Table B-2 `httpbind.config` ID Keys

Key	Description
<code>gwdomain-id.domains</code>	Comma-separated list of domains for this ID.
<code>gwdomain-id.hosts</code>	Space-separated list of hosts for this ID. Each of these hosts must be able to service the domains listed in <code>gwdomain-id.domains</code> . This list helps provide failover across the domains. If no explicit route host mentioned in the request, one of the hosts listed in this key will be used to service that request.
<code>gwdomain-id.componentjid</code>	The component JID to use to connect to the host.
<code>gwdomain-id.password</code>	The password to use to connect to the host.

Chapter 40. Instant Messaging imadmin Tool Reference

Oracle Communications Instant Messaging Server imadmin Tool Reference

This information describes how to use the `imadmin` command to administer Instant Messaging.

Topics:

- [imadmin Overview](#)
- [imadmin Requirements](#)
- [imadmin Location](#)
- [imadmin Commands](#)
- [imadmin Syntax](#)
- [imadmin Options](#)
- [imadmin Actions](#)
- [imadmin Components](#)

imadmin Overview

You can use the `imadmin` utility to start, stop, and refresh the Instant Messaging server and multiplexor. Run `imadmin` as `root` or as the end user you specified during configuration.

imadmin Requirements

You must invoke the `imadmin` utility from the host on which Instant Messaging server is installed.

imadmin Location

By default, `imadmin` is installed in the following location:

```
im-svr-base/sbin
```

imadmin Commands

Table C-1 lists and describes commands related to the `imadmin` command.

Table C-1 `imadmin` Commands and Descriptions

Command	Description
---------	-------------

<pre>imadmin assign_services</pre>	<div data-bbox="521 159 1377 310" style="background-color: #ffffcc; padding: 10px; border: 1px solid #ccc;"> <p>Starting with Instant Messaging 8 Update 3, the value <code>identity</code> which indicates that the Sun Java System Access Manager is used for policy storage, has been deprecated.</p> </div> <p>By default, LDAP is used for policy storage. The property <code>iim.policy.modules</code> must be set to <code>iim_ldap_schema1</code> or <code>iim_ldap_schema2</code> only if hosted domain support is required. If you configure Instant Messaging via the configurator, the possible values are <code>identity</code> and <code>iim_ldap</code>. The allowed values which can be manually set for <code>iim.policy.modules</code> are, <code>identity</code>, <code>iim_ldap</code>, <code>iim_ldap_schema1</code>, and <code>iim_ldap_schema2</code>. Only in the absence of hosted domain support, can <code>iim_ldap</code> be used as a value for <code>iim.policy.modules</code> for both <code>schema1</code> and <code>schema2</code> of the LDAP server.</p> <div data-bbox="521 642 1377 823" style="background-color: #e6f2ff; padding: 10px; border: 1px solid #ccc;"> <p>Note The <code>iim_ldap_schema1</code> and <code>iim_ldap_schema2</code> properties must be set in conjunction with other parameters mentioned in Configuring Hosted Domain Support.</p> </div> <p>If <code>iim.policy.modules</code> is set to <code>identity</code>, this command assigns Instant Messaging and presence services to existing users under the base DN you specify. The DN should be the base DN of the organization under which user entries are stored. If <code>iim.policy.modules</code> is set to <code>iim_ldap</code>, and <code>iim.userprops.store</code> is set to <code>ldap</code>, this command adds objectclasses (<code>sunIMUser</code>, and <code>sunPresenceUser</code>) to user entries in the directory. Instant Messaging requires these objectclasses in order to store properties in LDAP. The <code>assign_services</code> command fails if the LDAP search limit exceeds the defined value. To avoid this failure, increase the search limit of the LDAP server. Set the values of the directory server parameters as follows:</p> <ul style="list-style-type: none"> • To set unlimited limit for the search size, type <code>./dsconf set-server-prop search-size-limit:unlimited</code> • To set unlimited limit for the search time, type <code>./dsconf set-server-prop search-time-limit:unlimited</code>
<pre>imadmin status (Previously imadmin check)</pre>	<p>Checks to see if the components (<code>server</code>, <code>multiplexor</code>, <code>agent-calendar</code>, and <code>watchdog</code> are up and running and displays the results. If you don't specify a component, the <code>imadmin</code> utility returns information about all components.</p>
<pre>imadmin start</pre>	<p>Starts the enabled component(s).</p>
<pre>imadmin stop</pre>	<p>Stops the enabled component(s).</p>
<pre>imadmin refresh</pre>	<p>Refreshes the enabled component(s).</p>
<pre>imadmin start server</pre>	<p>Starts only the server.</p>
<pre>imadmin stop server</pre>	<p>Stops only the server.</p>
<pre>imadmin refresh server</pre>	<p>Refreshes only the server.</p>

<code>imadmin start multiplexor</code>	Starts only the multiplexor.
<code>imadmin stop multiplexor</code>	Stops only the multiplexor.
<code>imadmin refresh multiplexor</code>	Refreshes only the multiplexor.
<code>imadmin start agent-calendar</code>	Starts only the Calendar agent.
<code>imadmin stop agent-calendar</code>	Stops only the Calendar agent.
<code>imadmin refresh agent-calendar</code>	Refreshes only the Calendar agent.
<code>imadmin start watchdog</code>	Starts only the watchdog.
<code>imadmin stop watchdog</code>	Stops only the watchdog.
<code>imadmin refresh watchdog</code>	Refreshes only the watchdog.
<code>imadmin version</code>	Displays the version.
<code>imadmin migratepropstore</code>	Migrates user data from one property store(ldap/file) to the other.
<code>imadmin migratecert</code>	Migrates the private key and multiplexor certificate from the Network Security Services (NSS) database to Java Key Store (JKS).
<code>imadmin start sms-gateway</code>	Starts the SMS Gateway.
<code>imadmin stop sms-gateway</code>	Stops the SMS Gateway.
<code>imadmin status sms-gateway</code>	Displays the status of SMS Gateway.
<code>imadmin refresh sms-gateway</code>	Refreshes the SMS Gateway.
<code>imadmin start msn-gateway</code>	Starts the MSN Gateway. <div style="border: 1px solid black; background-color: #ffffcc; padding: 5px; margin-top: 10px;">As of Instant Messaging 9.0.1.4.0, the MSN gateway is deprecated; it may be removed in a future release.</div>
<code>imadmin stop msn-gateway</code>	Stops the MSN Gateway. <div style="border: 1px solid black; background-color: #ffffcc; padding: 5px; margin-top: 10px;">As of Instant Messaging 9.0.1.4.0, the MSN gateway is deprecated; it may be removed in a future release.</div>

<pre>imadmin status msn-gateway</pre>	<p>Displays the status of MSN Gateway.</p> <p>As of Instant Messaging 9.0.1.4.0, the MSN gateway is deprecated; it may be removed in a future release.</p>
<pre>imadmin refresh msn-gateway</pre>	<p>Refreshes the MSN Gateway.</p> <p>As of Instant Messaging 9.0.1.4.0, the MSN gateway is deprecated; it may be removed in a future release.</p>
<pre>imadmin start aim-gateway</pre>	<p>Starts the AIM Gateway.</p> <p>As of Instant Messaging 9.0.1.4.0, the AIM gateway is deprecated; it may be removed in a future release.</p>
<pre>imadmin stop aim-gateway</pre>	<p>Stops the AIM Gateway.</p> <p>As of Instant Messaging 9.0.1.4.0, the AIM gateway is deprecated; it may be removed in a future release.</p>
<pre>imadmin status aim-gateway</pre>	<p>Displays the status of AIM Gateway.</p> <p>As of Instant Messaging 9.0.1.4.0, the AIM gateway is deprecated; it may be removed in a future release.</p>
<pre>imadmin refresh aim-gateway</pre>	<p>Refreshes the AIM Gateway.</p> <p>As of Instant Messaging 9.0.1.4.0, the AIM gateway is deprecated; it may be removed in a future release.</p>
<pre>imadmin start yim-gateway</pre>	<p>Starts the YIM Gateway.</p> <p>As of Instant Messaging 9.0.1.4.0, the Yahoo (Yim-) gateway is deprecated; it may be removed in a future release.</p>
<pre>imadmin stop yim-gateway</pre>	<p>Stops the YIM Gateway.</p> <p>As of Instant Messaging 9.0.1.4.0, the Yahoo (Yim-) gateway is deprecated; it may be removed in a future release.</p>

imadmin status yim-gateway	Displays the status of YIM Gateway. As of Instant Messaging 9.0.1.4.0, the Yahoo (Yim-) gateway is deprecated; it may be removed in a future release.
imadmin refresh yim-gateway	Refreshes the YIM Gateway. As of Instant Messaging 9.0.1.4.0, the Yahoo (Yim-) gateway is deprecated; it may be removed in a future release.
imadmin start relay	Starts the relay server.
imadmin stop relay	Stops the relay server.
imadmin refresh relay	Refreshes the relay server.
imadmin status relay	Displays the status of relay server.

imadmin Syntax

```
imadmin [options] [action] [component]
```

imadmin Options

Table C-2 lists and describes options for the `imadmin` command.

Table C-2 Options for `imadmin` command

Option	Description
-c <i>alt-config-file</i>	Used with the <code>start</code> and <code>refresh</code> actions, to specify a different configuration file other than <code>/etc/opt/SUNWiim/config/iim.conf</code> file
-h	Displays help on the <code>imadmin</code> command.

imadmin Actions

Table C-3 lists and describes actions performed after various `imadmin` commands are issued.

Table C-3 Actions for `imadmin` Command

Option	Description
status (Previously imadmin check)	Returns information about Instant Messaging components (server, multiplexor, agent-calendar, and watchdog). You do not need to provide a <i>component</i> with this action.
start	Sets the classpath, the Java heap size and starts all the specified components.
stop	Stops all the specified component's daemons.
refresh	Stops and starts the specified component(s). Useful after a configuration change.

imadmin Components

Table C-4 lists and describes the components for the `imadmin` command.

Table C-4 Components for imadmin Command

Option	Description
agent-calendar	Indicates the Calendar agent (agent-calendar).
multiplexor	Indicates the multiplexor alone.
server	Indicates the Instant Messaging server.
watchdog	Indicates the watchdog.
sms-gateway	Indicates the SMS Gateway.
msn-gateway	Indicates the MSN Gateway. As of Instant Messaging 9.0.1.4.0, the MSN gateway is deprecated: it may be removed in a future release.
aim-gateway	Indicates the AIM Gateway. As of Instant Messaging 9.0.1.4.0, the AIM gateway is deprecated; it may be removed in a future release.
yim-gateway	Indicates the YAHOO Gateway. As of Instant Messaging 9.0.1.4.0, the Yahoo gateway is deprecated; it may be removed in a future release.
relay	Indicates the relay server.

Chapter 41. Instant Messaging APIs

Oracle Communications Instant Messaging Server APIs

This chapter describes the APIs used by Instant Messaging in the following sections:

- [Instant Messaging APIs Overview](#)
- [Instant Messaging Service API](#)
- [Messenger Beans](#)
- [Service Provider Interfaces](#)

Instant Messaging APIs Overview

Instant Messaging provides Java APIs which can be used to develop extension or integration modules. Detailed documentation of these APIs are provided with the installed Instant Messenger component, in the form of HTML files generated by Javadocs. The Javadoc files are installed in the `im-svr-base/html/apidocs/` directory. To view the API documentation, point your browser to `codebase/apidocs` where *codebase* is the Instant Messenger resources codebase.

The following are the Instant Messaging APIs:

- [Instant Messaging Service API](#)
- [Messenger Beans](#)
- [Service Provider Interfaces](#)

Instant Messaging Service API

The Instant Messaging API is used by the applications located on the same host or in the remote host to access Instant Messaging services, such as Presence, Conference, Notification, Polls and News channels.

The Instant Messaging Service API can be used for:

- A Java-based or web-based client, such as a portal channel.
- A Bridge or a Gateway to enable another class of clients.
- Integration of Instant Messenger and Presence into existing applications.
- Displaying news feeds as Instant Messenger news.

Messenger Beans

A Messenger bean is a dynamically loaded module used to extend Instant Messenger functionality. Messenger beans can add action listeners, such as buttons and menu items, and item listeners, such as check boxes and toggle buttons in the existing Instant Messenger window. The item listeners are invoked when an end-user input is received and bean-specific actions are based on the end-user input. Beans have the ability to add their own settings panel and save bean-specific properties on the server. Beans can be notified of any event received by Instant Messenger, for example, a new alert message.

The applications that use Messenger Beans include the following:

- Ability for end users to share application and conference along with voice or video.
- Ability to retrieve and process the transcript of a conference. For example, the contents of a

received or sent alert, for archiving purposes.

Service Provider Interfaces

The Service Provider Interface APIs provide the ability to extend the Instant Messaging server functionality. The Service Provider Interface is composed of the following independent APIs:

- [Archive Provider API](#)
- [Message Conversion API](#)

Archive Provider API

An Archive Provider is a software module usually providing integration with the archive or auditing system. Each configured Archive Provider is invoked for each server process.

The Archive Provider is invoked for the following server processes:

- When an instant message is sent, such as alert, poll, chat, news or conference messages.
- During an authentication event, such as login or logout.
- When there is a change in the presence status.
- During a subscription event. For example, when someone joins or leaves a conference, or subscribes or unsubscribes to a news channel.

The application that uses the Archive Provider API are as follows:

- Instant Messaging Archive
The default Instant Messaging archive in Instant Messaging is based on the Archive Provider API. For more information on Instant Messaging Archive, see [Managing Archiving for Instant Messaging](#).
- The application that records the usage statistics for sizing purposes.

Message Conversion API

A Message Converter is invoked for every message or each message part going through the server. The Message Converter may leave the message part intact or modify or remove the message part. The text parts are processed as Java String Objects. The Message Converter processes other attachment as a stream of bytes and returns a potentially different stream of bytes, or nothing at all if the attachment is to be removed.

The applications that uses Message Conversion API include the following:

- Virus checking and removal
- Translation engine integration
- Message content filtering

Authentication Provider API

The Authentication Provider API provides the ability to deploy Instant Messaging in environments that are not using Access Manager password-based or token-based authentication service. This API is invoked whenever an end user requests authentication, and it can be used in conjunction with the LDAP authentication.

Single Sign-on (SSO) with Access Manager is performed using the Authentication Provider API. This API can also be used to integrate with other authentication systems.

Chapter 42. Instant Messaging LDAP Schema

Oracle Communications Instant Messaging Server LDAP Schema

This section describes modifications made to the LDAP schema for Instant Messaging.

Instant Messaging Objectclasses

The following table lists LDAP objectclasses added to the schema and to entries in the directory for Instant Messaging.

Table E-1 Instant Messaging Objectclasses

Name	Description
sunIMUser	Contains user properties. Added to user entries under base DN specified when you run the <code>imadmin assign_services</code> command.
sunPresenceUser	Contains user presence properties. Added to user entries under base DN specified when you run the <code>imadmin assign_services</code> command.
sunIMNews	Contains news channel properties. If <code>userprops.store</code> is set to <code>ldap</code> , when a new news channel is created, an entry for the news channel is added to the directory. The news channel entry will contain the <code>sunIMNews</code> objectclass.
sunIMConference	Contains conference room properties. If <code>userprops.store</code> is set to <code>ldap</code> , when a new conference room is created, an entry for the conference room is added to the directory. The conference room entry will contain the <code>sunIMConference</code> objectclass.