Sun Java System Instant Messaging 7.2 Administration Guide

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 819–4412–05
January 2007

080818@20795

Contents

Preface 21

Part I Postinstallation Configuration 27

1 Configuring Instant Messaging After Installation 29
Completing the Configuration Checklist 29
Creating a UNIX System User and Group 38
▼ To Create the Appropriate UNIX User and Group 38
Overview of the `configure` Utility 39
Configuring Instant Messaging After Installing or Upgrading 39
▼ To Create the PASSFILE for the BEA Web Container 39
▼ To Configure Instant Messaging After Installation 39
Adding Instant Messaging and Presence Services to a Sub-organization in Access Manager for Single Sign-On and Policy Management Support 41
▼ To Manually Assign Instant Messaging and Presence Services to a Sub-organization in Access Manager 42
Performing a Silent Instant Messaging Configuration 42
▼ To Generate a Configure State File and ID for Instant Messaging 43
Creating Multiple Instances from a Single Instant Messaging Installation 44
▼ To Create an Additional Instance of Instant Messaging from an Existing Installation 44

2 Setting up and Launching Instant Messenger 47
Enabling Java Web Start 47
▼ To Add the MIME Type to Sun Java System Web Server Enterprise Edition 47
▼ To Add the MIME Type to Apache Web Container 48
Configuring Client Systems for Instant Messaging 48
Launching Instant Messenger 49

Running Instant Messenger From a Web Browser 49
Running Instant Messenger as a Standalone Application 50

Part II Administering Instant Messaging 51

3 Instant Messaging Configuration File and Directory Structure Overview 53
Instant Messaging Server Directory Structure 53
Instant Messaging Server Configuration File 55
Instant Messaging Data 55

4 Configuring Instant Messaging for High Availability (Solaris Only) 57
Instant Messaging HA Overview 57
Instant Messaging HA Configuration Software Requirements 58
Instant Messaging HA Configuration Permission Requirements 58
Instant Messaging HA Configuration Terms and Checklist 59
Setting Up HA for Instant Messaging 60
Choosing a Local or Shared Disk for Configuration Files and Binaries 60
Preparing Each Node in the Cluster 60
Selecting the Installation Directory (*im-svr-base*) 61
Installing Sun Java System Products and Packages 61
Configuring the HA Environment 62
Configuring the Logical Host 65
Registering and Activating the Storage Resource 66
Registering the Resource Type and Creating a Resource 66
Verifying the Instant Messaging HA Configuration 67
Troubleshooting the Instant Messaging HA Configuration 67
Stopping, Starting, and Restarting the Instant Messaging HA Service 68
▼ To Start the Instant Messaging HA Service 68
▼ To Stop the Instant Messaging HA Service 68
▼ To Restart the Instant Messaging HA Service 68
Stopping, Starting, and Restarting Instant Messaging Components in a Deployment with Sun Cluster 68
Managing the HA RTR File for Instant Messaging 69
Instant Messaging RTR File Parameters 69
Customizing the RTR File for Instant Messaging 70

Removing HA for Instant Messaging 70
▼ To Remove HA for Instant Messaging 70
HA Related Documentation 71

5 Enabling Single Sign-On (SSO) for Instant Messaging 73
SSO Limitations and Notices 73
Configuring Instant Messaging to Support Access Manager-Based SSO and Policies 74
▼ To Enable SSO for Instant Messaging 74
Troubleshooting SSO for Instant Messaging 75

6 Scaling an Instant Messaging Deployment Using Server Pooling 77
Overview of Server Pooling for Instant Messaging 77
Availability in an Instant Messaging Server Pool 78
Configuring Server-to-Server Communication Between Instant Messaging Servers in a Server Pool 78
▼ To Set Up Communication Between Two Instant Messaging Servers in a Server Pool 80
Adding a New Node to an Existing Instant Messaging Deployment 81
Securing a Multi-node Deployment 81
▼ To Manually Define the Dialback Key for an Instant Messaging Server in a Server Pool 81

7 Optimizing an Instant Messaging Server Pool Using the Redirect Server 83
Overview of Instant Messaging Redirect 83
Instant Messaging User Partitioning Algorithm 84
About the Instant Messaging Redirect Database 85
Instant Messaging Redirect Server Overview 85
Instant Messaging Redirect Server and StartTLS 85
Configuring an Instant Messaging Server Instance as a Redirect Server 85
▼ To Configure an Instant Messaging Server as a Redirect Server 87
Administering the Instant Messaging Redirect Server 88
Stopping, Starting, Restarting, Refreshing, and Checking the Status of the Instant Messaging Redirect Server 88
Instant Messaging Redirect Server Logging 88
Setting the Partition Size for the Instant Messaging Redirect Server 88
Specifying the List of Partitions for the Instant Messaging Redirect Server 89
Creating and Managing the Instant Messaging Redirect Table Using the `rdadmin` Utility 90

▼ To Create a New or Update an Existing Instant Messaging Redirect Database 90
Instant Messaging Redirect Server Physical Host Monitoring 91
▼ Setting the Instant Messaging Redirect Server Host Polling Frequency 91
Instant Messaging Redirect Server Best Practices and Troubleshooting 92
Redirect Server Certificates 92
Instant Messaging Redirect Server Supported Clients 92
Using Redirect Server and Storing User Properties in LDAP 92
Determining the Partition Size for the Redirect Database 92
Using a Redirect Server as a Partition Host 93

8 Federating Deployment of Multiple Instant Messaging Servers 95
Configuring Federated Communication Between Instant Messaging Servers 95
▼ To Federate Communication Between Two Instant Messaging Servers 96

9 Administering Instant Messaging Components 99
Stopping, Starting, Refreshing, and Checking Instant Messaging Components 99
Starting Instant Messaging Components 100
Stopping Instant Messaging Components 101
Refreshing Component Configuration 102
Checking the Status of Instant Messaging Components 103
Changing Instant Messaging Server and Multiplexor Configuration Parameters 104
▼ To Change Configuration Parameters 104
Backing Up Instant Messaging Data 105
Backup Information 105
Performing a Backup 105
Restoring Backup Information 105

10 Using the Instant Messaging XMPP/HTTP Gateway 107
Instant Messaging XMPP/HTTP Gateway Configuration Files 107
Configuring the Instant Messaging XMPP/HTTP Gateway 108
▼ To Enable or Disable the Instant Messaging XMPP/HTTP Gateway 109
▼ To Configure the Number of Concurrent Requests Handled by the XMPP/HTTP Gateway 109
▼ To Set the JEP 124 *hold* Attribute for Client Requests to the XMPP/HTTP Gateway 110
▼ To Specify the Allowed Client Inactivity Time for the XMPP/HTTP Gateway 110

▼ To Set the `content-type` HTTP Header for the XMPP/HTTP Gateway 111
▼ To Set the Round Trip Delay for the XMPP/HTTP Gateway 111
▼ To Set the Default Time Within Which the XMPP/HTTP Gateway Will Send a Response to the Client 112
▼ To Configure an XMPP/HTTP Gateway in a Instant Messaging Gateway Pool 112
▼ To Configure the List of Key IDs for Supported XMPP/HTTP Gateway Domains 113
▼ To Configure the Instant Messaging XMPP/HTTP Gateway to Use a Non-default Configuration File 114
Securing Communication Between the XMPP/HTTP Gateway and Instant Messaging Server Using StartTLS 114
Managing Logging for the XMPP/HTTP Gateway 115
▼ To Enable or Disable Logging for the XMPP/HTTP Gateway 115
▼ To Change the Location of the XMPP/HTTP Gateway Log Configuration File 116
▼ Linux: To Set the Location of the XMPP/HTTP Gateway Log File After Install or Upgrade 116
▼ To Change the Location of the XMPP/HTTP Gateway Log File 117
▼ To Use a Non-default Log File Location for the XMPP/HTTP Gateway 117
▼ To Set the XMPP/HTTP Gateway Logging Level 117
XMPP/HTTP Gateway log4j Log Configuration File Syntax 118

11 Managing Instant Messaging's LDAP Access Configuration 119
Overview of how Instant Messaging Uses LDAP 119
Searching the Directory Anonymously 120
▼ To Enable the Server to Conduct Directory Searches as a Specific End User 120
Configuring Instant Messaging to Use LDAP Dynamic Groups 121
▼ To Configure Instant Messaging to Use Dynamic Groups 121

12 Securing Instant Messaging Using TLS and Legacy SSL 123
Overview of Using TLS and Legacy SSL in Instant Messaging 123
Setting Up TLS for the Instant Messaging Server 124
Activating TLS on the Instant Messaging Server 125
▼ To Activate TLS Communication in the Instant Messaging Server 126
Setting Up Legacy SSL for the Multiplexor and Instant Messenger 128
Requesting an SSL Certificate for the Instant Messaging Multiplexor from the CA 129
Installing the Certificate 130
Enabling Legacy SSL Between the Multiplexor and Instant Messenger 131

Invoking the Secure Version of Instant Messenger 134
▼ To Verify a Secure Instant Messenger Connection 134

13 Managing Logging for Instant Messaging 135
Instant Messaging Logging Overview 135
Instant Messaging Log File Location 136
Instant Messaging Component Logging Levels 136
Managing Instant Messaging Logging Using Log4j 137
Instant Messaging Log4j Configuration File (`log4j.conf`) Location 138
Instant Messaging Log4j Log File Syntax 138
Log4j Log Levels for Instant Messaging Components 141
▼ To Specify the Location of the Log4j Configuration File (`Log4j.conf`) 141
▼ To Enable or Disable Log4j Logging for an Instant Messaging Component 142
▼ To Set Log4j Log Levels for Instant Messaging 142
▼ To Specify the Maximum Log4j Log File Size for Instant Messaging Components 143
Configuring Logging for Instant Messaging Components Using `iim.conf` Parameters 143
▼ To Set Log Levels for Instant Messaging Components Using `iim.conf` Parameters 145
Administering Logging for Instant Messenger 145
Setting Up Logging for Instant Messenger 145
Locating the Instant Messenger Log File (`messenger.log`) 146
Instant Messenger Log File Content Options 146

14 Administering Instant Messaging End Users 149
Disabling End User Access to Instant Messenger 150
▼ To Disable Instant Messaging End Users 150
Registering New Instant Messaging Users 150
Configuring the Instant Messaging Server to Allow New User Registration 151
Customizing Instant Messenger to Allow New User Registration 152
Registering as a New Instant Messaging User 153
Storing Instant Messaging User Properties in LDAP 154
▼ To Store Instant Messaging User Properties in LDAP 154
Assigning Instant Messaging and Presence Services to End Users 154
▼ To Assign Instant Messaging and Presence Services to End Users 155

15 Managing Instant Messenger 157
Configuring Instant Messenger 157
Invoking Instant Messenger 158
▼ To Invoke Instant Messenger Using a Direct URL 158
▼ To Invoke Instant Messenger From the Command-Line (Solaris Only) 159
▼ To Invoke Instant Messenger Using a Desktop Shortcut 159
Changing the Codebase 159
▼ To Change the Codebase in the Resource Templates 160
Changing the Web Container Port 160
Customizing Instant Messenger 160
Instant Messenger Resource Files 160
Customizing the `index.html` and `im.html` Files 162
Launching Instant Messenger Using Sun Java System Access Manager SSO 163
Customizing the Application (Java Web Start) 164
Contents of `imbrand.jar` 165
Rebranding Instant Messenger 171
Customizing User Name and Group Name Display 172
Modifying How Client Users Search for Contacts 174
▼ To Allow Users to Search on Custom Attributes 174
▼ To Allow Wildcards in Searches 175
Administering Conference Rooms and News Channels 175
Modifying Instant Messenger Proxy Settings 176
▼ To Set Proxy Settings Manually for a Single Instant Messenger Client Using Java Web Start 176
▼ To Configure Proxy Settings for all Instant Messaging Client Connections in `im.jnlp` 177
Controlling the Exposed Messenger Feature Set 177
Instant Messenger Data Stored in the End User's System 178
Redeploying Resource Files 180
▼ To Redeploy Resource Files to Sun Java System Application Server or Sun Java System Web Server 180

16 Using Calendar Pop-up Reminders 181
Pop-up Reminders Overview 181
Pop-up Reminders Operation 181
Pop-up Reminders Architectural Flow 182
`iim.conf` Calendar Pop-up Configuration Parameters 182

Configuring Instant Messaging Pop-ups 185
▼ To Configure Instant Messaging Server for Calendar Pop-ups Using the `configure` Utility 185
▼ To Manually Configure Instant Messaging Server for Calendar Pop-ups 186
▼ To Configure Calendar Server for Pop-ups 186
▼ To Configure Instant Messenger for Calendar Pop-ups 187
Configuring Calendar Pop-ups in a Server Pool 187
Administering the Calendar Agent 187
▼ Enabling and Disabling Instant Messaging Agents 188

17 Managing Instant Messaging and Presence Policies 189
Overview of Privacy, Security, and Site Policies 189
Site Policies 189
Conference Room and News Channel Access Controls 190
User Privacy 190
Methods for Controlling End User and Administrator Privileges 191
Setting the Policy Management Method 192
Policy Configuration Parameters 192
Managing Policies Using Access Control Files 193
▼ To Change End-user Privileges in Access Control Files 194
Using Access Control Files in a Server Pool 195
Access Control File Location 195
Access Control File Format 195
Managing Policies using Sun Java System Access Manager 196
Instant Messaging Service Attributes 197
Modifying Attributes Directly 199
Predefined Instant Messaging and Presence Policies 201
Creating New Instant Messaging Policies 203
Assigning Policies to a Role, Group, Organization, or User 205
Creating New Suborganizations Using Access Manager 206
Assigning Roles to End Users in New Suborganizations 208

18 Managing Archiving for Instant Messaging 211
Archiving Overview 211
Enabling and Disabling Archiving for Instant Messaging 212

▼ To Enable Instant Messaging Archiving 212
▼ To Disable Instant Messaging Archiving 212
Managing the Instant Messaging Email Archive 213
Enabling and Disabling the Instant Messaging Email Archive Provider 213
Configuring Email Archive Settings 214
Email Header Format 216
Managing the Instant Messaging Portal Archive 219
Instant Messaging Portal Archive Overview 219
Enabling and Disabling the Portal Archive Provider 221
Configuring the Instant Messaging Portal Archive Provider 222
Managing Archived Data in the Portal Server Search Database 225
Changing the Display of Archived Data 227
Sample Deployment Scenario for Archive Provider 227
Using a Custom Archive Provider 228
▼ To Enable a Custom Archive Provider 229
▼ To Disable a Custom Archive Provider 229

19 Troubleshooting and Monitoring Instant Messaging 231
Troubleshooting Instant Messenger 231
Obtaining Instant Messenger Runtime Information 231
Obtaining Instant Messenger Logs 232
Problems and Solutions 232
Unable to Connect to Instant Messaging Redirect Server from Client 233
Unable to Log into Instant Messenger through the XMPP/HTTP Gateway 233
Messages Not Archived With Sun Java System Portal Server 7 2006Q1 or Later 234
Instant Messenger Resource Customizations Lost After `patchrm` and `patchadd` 234
Cannot Forward Mail to Offline Users 234
Calendar Pop-up Reminders Do Not Work 235
Single Sign-on Does Not Work 236
Instant Messenger Does Not Load or Start 236
Connection Refused or Timed Out 236
Authentication Errors 237
Instant Messenger Channel Display Error 237
Instant Messaging Content is not Archived 238
Server-to-Server Communication Fails to Start 238

Catastrophic Installation Failure Leaves Server in an Inconsistent State 238
Instant Messaging Services Do Not Appear in the Access Manager Console (amconsole) 239
Troubleshooting Instant Messaging and LDAP 239
Using a Directory That Does not Permit Anonymous Bind 239
Displaying Contact Names Using an Attribute Other than `cn` 240
Searching the Directory Using Wildcards 241
Using Nonstandard Objectclasses for Users and Groups 241
Using an Attribute Other than `uid` for User Authentication 242
Using an Attribute Other than `uid` for User IDs 242
Troubleshooting Connectivity Issues in a Multi-Node Deployment (Server Pool) 243
Monitoring Instant Messaging 243
Managing the Watchdog Process 243
Determining the Status of the Watchdog 244
Enabling and Disabling the Watchdog 244
Managing Logging for the Watchdog 244

Part III Reference Information 247

A Instant Messaging Configuration Parameters in `iim.conf` 249
`iim.conf` File Location 249
`iim.conf` File Syntax 250
General Configuration Parameters 250
LDAP and User Registration Configuration Parameters 253
Logging Configuration Parameters 255
Instant Messaging Server Configuration Parameters 256
Multiple Server Configuration Parameters 261
Multiplexor Configuration Parameters 263
Redirect Server Parameters 264
Archive Parameters 266
Watchdog Parameters 271
Monitoring Parameters 271
Agent Parameters 272

B Instant Messaging XMPP/HTTP Gateway Configuration Parameters in `httpbind.conf` 275
`httpbind.conf` File Location 275
`httpbind.conf` File Syntax 276
Instant Messaging XMPP/HTTP Gateway Configuration Parameters 276
Gateway Domain ID Key Parameters for *httpbind.config* 279

C Instant Messaging `imadmin` Tool Reference 281
`imadmin` Overview 281
`imadmin` Requirements 281
`imadmin` Location 282
`imadmin` Commands 282
`imadmin` Syntax 283
`imadmin` Options 283
`imadmin` Actions 283
`imadmin` Components 284

D Instant Messaging APIs 285
Instant Messaging APIs Overview 285
Instant Messaging Service API 285
Messenger Beans 286
Service Provider Interfaces 286
Archive Provider API 286
Message Conversion API 287
Authentication Provider API 287

E Instant Messaging LDAP Schema 289
Instant Messaging Objectclasses 289

Index 291

Figures

FIGURE 18–1 Instant Messaging Portal Archive Components 220

Tables

TABLE 1–1 Configuration Parameters for Instant Messaging 30
TABLE 3–1 Instant Messaging server directories 53
TABLE 4–1 Software Requirements for Instant Messaging HA Configuration 58
TABLE 4–2 HA Configuration Checklist 59
TABLE 4–3 Products and Packages Required for a Multiple Node Instant Messaging HA Configuration 61
TABLE 4–4 `SUNW.iim` Extension Properties 69
TABLE 5–1 Instant Messaging Single Sign-On Parameters 74
TABLE 6–1 Example Configuration Information for Two Instant Messaging Servers in a Server Pool 79
TABLE 7–1 Redirect Server Configuration Parameters in `iim.conf` 85
TABLE 8–1 Example Configuration Information for Two Federated Instant Messaging Servers 96
TABLE 12–1 Instant Messaging Server TLS Configuration Parameters 125
TABLE 12–2 Instant Messaging Multiplexor SSL Parameters 132
TABLE 13–1 Logging Levels for Instant Messaging Components 137
TABLE 13–2 Log File Names and Logging Level Configuration Parameters for Instant Messaging Components 144
TABLE 13–3 Instant Messenger Logging Options for `messenger.log` 146
TABLE 14–1 Instant Messaging Server New User Registration Configuration Parameters 151
TABLE 15–1 Instant Messenger Resource Files in *im-svr-base*`/html` 161
TABLE 15–2 Configuration Files 166
TABLE 15–3 Emoticons 166
TABLE 15–4 Application Icons – Windows 168
TABLE 15–5 Application Icons – All Platforms 168
TABLE 15–6 Toolbar Icons 168
TABLE 15–7 Contact List Icons 168
TABLE 15–8 Presence Icons - Contact List 168
TABLE 15–9 Presence Icons - Status Bar 169

TABLE 15–10 Backgrounds and Background Swatches for the Palette 170
TABLE 15–11 Sounds 171
TABLE 15–12 Instant Messenger Applet Parameters 177
TABLE 15–13 Cached Data Directory and Files 179
TABLE 15–14 Auto-logon Properties 179
TABLE 16–1 `iim.conf` Parameters for Configuring Calendar Pop-ups 182
TABLE 17–1 Parameters Related to Access Manager in `iim.conf` 193
TABLE 17–2 Access Control Files 194
TABLE 17–3 Access Manager Attributes for Instant Messaging 197
TABLE 17–4 Access Manager Policy Attributes for Instant Messaging 198
TABLE 17–5 Access Manager User and Dynamic Attributes for Instant Messaging 200
TABLE 17–6 Default Policies and Roles for Sun Java System Access Manager 201
TABLE 17–7 Default Policy Assignments 202
TABLE 18–1 Email Archive Configuration Parameters 214
TABLE 18–2 Unique ID and Description for Archive Provider Categories 223
TABLE A–1 General Configuration Parameters 251
TABLE A–2 LDAP, User Registration, and Source Configuration Parameters 253
TABLE A–3 Logging Configuration Parameters 255
TABLE A–4 General Instant Messaging server Configuration Parameters 257
TABLE A–5 Multiple Server Configuration Parameters 262
TABLE A–6 Multiplexor Configuration Parameters 263
TABLE A–7 Redirect Server Parameters 265
TABLE A–8 Archive Parameters 266
TABLE A–9 Watchdog Configuration Parameters 271
TABLE A–10 Monitoring Parameters 271
TABLE A–11 Agent Configuration Parameters 272
TABLE B–1 XMPP/HTTP Gateway Configuration Parameters in `httpbind.conf` 277
TABLE B–2 *httpbind.config* ID Keys 279
TABLE C–1 `imadmin` Commands and Descriptions 282
TABLE C–2 Options for `imadmin` command 283
TABLE C–3 Actions for `imadmin` Command 283
TABLE C–4 Components for `imadmin` Command 284
TABLE E–1 Instant Messaging Objectclasses 289

Examples

EXAMPLE 7–1 Instant Messaging Redirect Sequence of Events 84
EXAMPLE 7–2 `Redirect.partitions` File Configuration 89
EXAMPLE 10–1 XMPP/HTTP Gateway Log Configuration File (`httpbind_log4j.conf`) 118
EXAMPLE 12–1 TLS Configuration in `iim.conf` 128
EXAMPLE 12–2 Legacy SSL Multiplexor Configuration in `iim.conf` 133
EXAMPLE 13–1 Log4j Template File 139
EXAMPLE 15–1 Sample `im.jnlp` File 164
EXAMPLE 17–1 `sysTopicsAdd.acl` File 196
EXAMPLE 18–1 Archiving Related Instant Messaging Data Collectively 227

Preface

Instant Messaging enables end users to participate in real-time interactive messaging and discussions. Sun Java System Instant Messaging allows end users to participate in Instant Messaging and chat sessions, send alert messages to each other, and share group news instantly. It is suitable for both intranets and the Internet. The *Sun Java™ System Instant Messaging 7.2 Administration Guide* explains in detail how to perform the basic duties of administrating the Instant Messaging system.

Who Should Use This Book

Read this book if you are responsible for administering, configuring, and deploying Instant Messaging. This book assumes that you have an understanding of JavaScript™, HTML, and any of the following servers in your deployment:

- Sun Java System Portal Server
- A web container such as Sun Java System Application Server SE (Standard Edition)
- An SMTP server such as Sun Java System Messaging Server
- An LDAP server such as Sun Java SystemDirectory Server
- Sun Java System Calendar Server
- Sun Java System Access Manager

Before You Read This Book

You must be familiar with the following books and release notes before reading this book:

- *Sun Java Enterprise System 5 Release Notes for UNIX*
- Chapter 4, "Sun Java System Instant Messaging 7.2 Release Notes," in *Sun Java Communications Suite 5 Release Notes*
- *Sun Java Communications Suite 5 Deployment Planning Guide*
- *Sun Java Enterprise System 5 Installation Guide for UNIX*
- *Sun Java Enterprise System 5 Installation Planning Guide*
- *Sun Java Enterprise System 2006Q3 Upgrade Guide*

- *Sun Java Enterprise System 5 Installation Reference for UNIX*

Before performing the tasks in this book, you should have already installed Instant Messaging.

How This Book Is Organized

This book contains the following sections:

Chapter 1, "Configuring Instant Messaging After Installation," contains configuration steps you need to complete after you install or upgrade, before you can use Instant Messaging.

Chapter 2, "Setting up and Launching Instant Messenger," provides information about configuring client systems, enabling Java Web Start, and adding additional localization client files. Also explains how to launch the client.

Chapter 3, "Instant Messaging Configuration File and Directory Structure Overview," provides information about the configuration files you use to administer Instant Messaging.

Chapter 4, "Configuring Instant Messaging for High Availability (Solaris Only)," describes how to install and configure a highly available Instant Messaging service with Sun Cluster.

Chapter 5, "Enabling Single Sign-On (SSO) for Instant Messaging," describes SSO and how to configure it for Instant Messaging.

Chapter 6, "Scaling an Instant Messaging Deployment Using Server Pooling," gives instructions on creating a server pool for a single domain to increase horizontal scalability.

Chapter 7, "Optimizing an Instant Messaging Server Pool Using the Redirect Server," describes using the redirect server to optimize performance in an Instant Messaging server pool.

Chapter 8, "Federating Deployment of Multiple Instant Messaging Servers," details how to support multiple domains in your Instant Messaging deployment.

Chapter 9, "Administering Instant Messaging Components," describes how to administer the Instant Messaging server, multiplexor, Calendar agent, cluster agent, and watchdog.

Chapter 10, "Using the Instant Messaging XMPP/HTTP Gateway," provides instructions on setting up and using the gateway.

Chapter 11, "Managing Instant Messaging's LDAP Access Configuration," contains information on configuring LDAP for use with Instant Messaging.

Chapter 12, "Securing Instant Messaging Using TLS and Legacy SSL," provides information about and how Instant Messaging uses legacy SSL and TLS to ensure security.

Chapter 13, "Managing Logging for Instant Messaging," describes configuring logging for Instant Messaging components and XMPP.

Chapter 14, "Administering Instant Messaging End Users," provides instructions for disabling end user access to Instant Messenger, registering new users, using LDAP to store user properties, and assigning Instant Messaging and presence services to end users.

Chapter 15, "Managing Instant Messenger," describes how to customize and administer Instant Messenger.

Chapter 16, "Using Calendar Pop-up Reminders," describes how to configure the Instant Messaging server, Calendar agent, Calendar Server, and Instant Messenger to enable Calendar pop-up reminders.

Chapter 17, "Managing Instant Messaging and Presence Policies," describes how to manage administrator and end user privileges, especially with policies set in Sun Java System Access Manager.

Chapter 18, "Managing Archiving for Instant Messaging," explains how to manage and configure the Instant Messaging archive.

Chapter 19, "Troubleshooting and Monitoring Instant Messaging," lists the common problems that might occur during installation and deployment of Instant Messaging and provides instructions for using the monitoring agent.

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 XMPP/HTTP 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.

Related Third-Party Web Site References

Third-party URLs are referenced in this document and provide additional, related information.

Note – Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- Documentation (`http://www.sun.com/documentation/`)
- Support (`http://www.sun.com/support/`)
- Training (`http://www.sun.com/training/`)

Typographic Conventions

The following table describes the typographic conventions that are used in this book.

TABLE P–1 Typographic Conventions

Typeface	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`
`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`** `Password:`
aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. A *cache* is a copy that is stored locally. Do *not* save the file. **Note:** Some emphasized items appear bold online.

Shell Prompts in Command Examples

The following table shows the default UNIX® system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–2 Shell Prompts

Shell	Prompt
C shell	`machine_name%`
C shell for superuser	`machine_name#`
Bourne shell and Korn shell	`$`
Bourne shell and Korn shell for superuser	`#`

PART I

Postinstallation Configuration

- Chapter 1, "Configuring Instant Messaging After Installation," contains configuration steps you need to complete after you install or upgrade, before you can use Instant Messaging.
- Chapter 2, "Setting up and Launching Instant Messenger," provides information about configuring client systems, enabling Java™ Web Start, and adding additional localization client files. Also explains how to launch the client.

◆ ◆ ◆ CHAPTER 1

Configuring Instant Messaging After Installation

After installation, you need to complete a few configuration steps before using Sun Java™ System Instant Messaging. This chapter describes these configuration steps in the following sections:

- "Completing the Configuration Checklist" on page 29
- "Creating a UNIX System User and Group" on page 38
- "Overview of the `configure` Utility" on page 39
- "Configuring Instant Messaging After Installing or Upgrading" on page 39
- "Adding Instant Messaging and Presence Services to a Sub-organization in Access Manager for Single Sign-On and Policy Management Support" on page 41
- "Performing a Silent Instant Messaging Configuration" on page 42
- "Creating Multiple Instances from a Single Instant Messaging Installation" on page 44

Before you configure Instant Messaging, you should read and understand the information in the *Sun Java Communications Suite 5 Deployment Planning Guide*, perform the installation as described in *Sun Java Communications Suite 5 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 Chapter 4, "Configuring Instant Messaging for High Availability (Solaris Only)," before completing the steps in this chapter.

Completing the Configuration Checklist

You should gather this information before you begin. You will be 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.

(Solaris Only) If you will be configuring High Availability service for Instant Messaging, see "Instant Messaging HA Overview" on page 57 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	*im-svr-base* Directory in which Instant Messaging is installed. By default, Instant Messaging is installed into the `/opt` directory as follows: Solaris: `/opt/SUNWiim` Linux: `/opt/sun/im` (Solaris Only) If you will be configuring High Availability service for Instant Messaging, see "Selecting the Installation Directory (*im-svr-base*)" on page 61 for information about choosing an installation directory.	
Instant Messaging Server Host and Domain Name	Host name on which Instant Messaging is installed and the domain name associated with the host. For example: Host Name: `instantmessaging.siroe.com` Domain Name: `siroe.com` (Solaris Only) If you will be configuring High Availability service for Instant Messaging, use the logical host name.	
Instant Messaging Server Port Number	The port number on which the Instant Messaging Server listens for incoming requests from the multiplexor. Default: 45222	
Instant Messaging Server-to-Server Port Number	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. Default: 5269	

TABLE 1–1 Configuration Parameters for Instant Messaging *(Continued)*

Parameter	Description	Your Value
Multiplexor Port Number (Multiplexor Configuration Only)	The port number on which the Instant Messaging Server listens for incoming requests from Instant Messenger clients. Default: 5222	
Disable Server	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.	
Remote Instant Messaging Server Host Name (Multiplexor Configuration Only)	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 `localhost`. (Solaris Only) If you will be configuring High Availability service for Instant Messaging, use the logical host's name. Dependencies: The Disable Server parameter must be selected, that is, server functionality is disabled.	

TABLE 1–1 Configuration Parameters for Instant Messaging *(Continued)*

Parameter	Description	Your Value
Sun Java System Access Manager Configuration	If the `configure` 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: ■ Are you planning to leverage an Access Manager deployment for SSO? If you enter **yes**, configure sets the `iim_server.usesso` parameter in `iim.conf` 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 **yes** you need to run the `imadmin assign_services` command when you are finished running the `configure` utility. See "To Configure Instant Messaging After Installation" on page 39 and "Assigning Instant Messaging and Presence Services to End Users" on page 154 for more instructions on using the `imadmin assign_services` command. If you choose **no**, 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 (amconsole).	

TABLE 1–1 Configuration Parameters for Instant Messaging *(Continued)*

Parameter	Description	Your Value
Sun Java System Calendar Server and Calendar Agent Configuration	The `configure` 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: ■ Notification server hostname. ■ Notification server port number. ■ Calendar alarm URL. 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 Chapter 16, "Using Calendar Pop-up Reminders."	
Enable Instant Messaging Archive (Optional)	If selected, enables Sun Java System Portal Server search-based archiving for Instant Messaging. Dependencies: Sun Java System Portal Server and Sun Java System Access Manager.	
LDAP Host Name	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, `directory.siroe.com`. Dependencies: LDAP server such as Sun Java System Directory Server.	
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 Sun Java System Directory Server.	
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 Sun Java System Directory Server.	

TABLE 1–1 Configuration Parameters for Instant Messaging *(Continued)*

Parameter	Description	Your Value
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" on page 240. Dependencies: LDAP server such as Sun Java System Directory Server.	
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 Sun Java System Messaging Server.	

TABLE 1–1 Configuration Parameters for Instant Messaging *(Continued)*

Parameter	Description	Your Value
Database, Logs, and Runtime Files Pathname	The location where the runtime files, database, and logs are stored. Also referred to as *im-runtime-base*. 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. If you are configuring High Availability (HA) for Instant Messaging, this path must be globally available. See Chapter 4, "Configuring Instant Messaging for High Availability (Solaris Only)," for more information about HA. The `configure` utility appends a directory (`/default`) to the path you provide for the runtime files. The name of this directory is the instance to which the runtime files apply. Later, you can create multiple instances of Instant Messaging by creating additional instance directories with different names (for example `/secure`) and copying over files from the `/default` instance runtime directory. See "Creating Multiple Instances from a Single Instant Messaging Installation" on page 44 for specific instructions. If you accept the following defaults when you run `configure`: Solaris: `/var/opt/SUNWiim/` Linux: `/var/opt/sun/im/` The `configure` utility creates the following directories for the runtime files: Solaris: `/var/opt/SUNWiim/default` Linux: `/var/opt/sun/im/default` In addition, the following two subdirectories are created under the runtime directory. The database directory (*im-db-base*) defaults are as follows: Solaris: `/var/opt/SUNWiim/default/db` Linux: `/var/opt/sun/im/default/db` The log directory defaults are as follows: Solaris: `/var/opt/SUNWiim/default/log` Linux: `/var/opt/sun/im/default/log`	

TABLE 1–1 Configuration Parameters for Instant Messaging *(Continued)*

Parameter	Description	Your Value
Resources, Help Files, and HTTP Gateway Pathname	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 `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" on page 180 for information. Default: *im-svr-base*`/html`	
XMPP/HTTP Gateway Deployment	Determines whether or not the XMPP/HTTP gateway will be deployed. If you choose to deploy the gateway, the `configure` utility creates a default gateway configuration file (`httpbind.conf`) in the default Instant Messaging server instance's *im-cfg-base* directory if one does not already exist. If `httpbind.conf` already exists, the `configure` utility does not alter or overwrite the file. Default: `True` (gateway is deployed)	
XMPP/HTTP Gateway URI	Defines the URI for the HTTP component of the XMPP/HTTP gateway. Default: `http://`*web-svr-host*`:80/httpbind`	

TABLE 1–1 Configuration Parameters for Instant Messaging *(Continued)*

Parameter	Description	Your Value
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: Linux: `/opt/sun/im/html` Solaris: `/opt/SUNWiim/html` The `configure` utility uses the codebase to determine which web container instance to use. If it succeeds, the `configure` 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 will be prompted for a file system location in which to copy or link the resources. If you are using Instant Messaging with Sun Java System Application Server or Sun Java System Web Server, the `configure` utility automatically publishes the resource files to the web container for you. For Sun Java System Application Server, the `configure` utility uses the `asadmin` command, for Sun Java System Web Server 6, the `configure` utility uses the `wdeploy` command, for Sun Java System Web Server 7, the `configure` utility uses the `wadm` command. If you are using a different web container, the `configure` utility copies the files to a location you specify. This should include the web container's doc root. Alternatively, you can add the resource files installation directory as a doc root in your web container's configuration. See the documentation for your web container for more specific instructions. In addition, you can use a symbolic link to make the resources visible to the web container. For example, on Solaris the resources can be made visible to the web container by creating the following symbolic link: `ln -s /opt/SUNWiim/html` *docroot*`/im` Where *docroot* is the doc root of the web container, for example `/opt/web`. If you are using SSO with Sun Java System Access Manager, the Access Manager server	

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 Solaris, 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 Solaris, 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.

Overview of the `configure` Utility

You use the `configure` utility after you install the software to configure information about your deployment and to generate the configuration files you use to administer and run Instant Messaging.

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" on page 180 for information. Also see "Completing the Configuration Checklist" on page 29 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.

Configuring Instant Messaging After Installing or Upgrading

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 "To Configure Instant Messaging After Installation" on page 39.

▼ To Create the PASSFILE for the BEA Web Container

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.**

▼ To Configure Instant Messaging After Installation

1. **Change to the directory in which you installed Instant Messaging.**

   By default, this directory is `/opt/SUNWiim` on Solaris, and `/opt/sun/im` on Linux.

2 **Run the `configure` utility in one of the following ways:**

Graphical user interface:

```
configure
```

Command-line:

```
configure --nodisplay
```

From a state file:

```
configure --nodisplay --noconsole --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" on page 42 for information on generating a state file.

If you are configuring using the graphical user interface 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" on page 29.

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 Solaris and `/opt/sun/im/lib/` on Linux.

b. **Copy the files to the `locale` directory on the Access Manager host.**

By default this directory is `/opt/SUNWam/locale` on Solaris and `/opt/sun/identity/locale` on 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 Solaris, you chose to use a location for logs other than the default
   - On 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 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" on page 41 for information.

8. **Configure the web container and client systems to support Instant Messaging.**

   For instructions, see Chapter 2, "Setting up and Launching Instant Messenger."

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 only automatically adds these services to the top-level organization and only if you select `yes` when prompted if you are planning 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 checkboxes 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.

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. Next, you will create a platform-appropriate state file ID and modify the state file to include this ID.

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" on page 39 for information on using the state file to configure a new instance of Instant Messaging.

▼ To Generate a Configure State File and ID for Instant Messaging

1 **Log in as superuser.**

2 **Change to the directory in which you installed Instant Messaging.**

By default, this directory is `/opt/SUNWiim` on Solaris, and `/opt/sun/im` on 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 --noconsole --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.

4 **You may need to generate a new platform-appropriate state file ID if you meet either of the following criteria:**

- You already have a state file you generated for a previous version or patch of Instant Messaging.
- You already have a state file generated for a previous version and have applied a patch that contains a new or modified version of `config.class`.

In either case, the old state file ID will no longer be valid. Complete the following to generate a new ID and replace the old one as follows:

a. **Run the `configure` utility again, but this time with the `--id` option as follows:**

`configure --id`

The command generates an encrypted identifier.

b. Copy the identifier and paste the value into the state file as the value for the *STATE_BEGIN* and *STATE_DONE* parameters.

For information on using the state file to configure a different installation of Instant Messaging, see "Configuring Instant Messaging After Installing or Upgrading" on page 39.

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 would be as follows:

Solaris: `/var/opt/SUNWiim/default`

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`:

Solaris: **`mkdir /var/opt/SUNWiim/xyz`**

Linux: **`mkdir /var/opt/sun/im/xyz`**

2 Create a log directory for the new instance:

For example, to create a new log directory for instance `xyz`:

Solaris: **`mkdir /var/opt/SUNWiim/xyz/log`**

Linux: **`mkdir /var/opt/sun/im/xyz/log`**

3. **If you are using a file-based property store for user data, you need to create a database directory (*im-db-base*) for the new instance:**

   For example, to create a new database directory for instance `xyz`:

   Solaris: **`mkdir /var/opt/SUNWiim/xyz/db`**

   Linux: **`mkdir /var/opt/sun/im/xyz/db`**

4. **Copy the contents of the *im-svr-base* directory and all of its subdirectories into the newly created directories:**

   For example:

   Solaris: **`cp -r /etc/opt/SUNWiim/default /etc/opt/SUNWiim/xyz`**

   Linux: **`cp -r /etc/opt/sun/im/default /etc/opt/sun/im/xyz`**

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:

   Solaris: `/etc/opt/SUNWiim/xyz/imadmin`

   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 Solaris, change `/etc/opt/SUNWiim/default/config/iim.conf` to `/etc/opt/SUNWiim/xyz/config/iim.conf`.

   On 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:

   Solaris: `/etc/opt/SUNWiim/xyz/config/iim.conf`

   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*) – 45222

For more information about these parameters, see Appendix A, "Instant Messaging Configuration Parameters in `iim.conf`."

10 Modify *iim.instancedir* **to point to** *im-svr-base*.

See "Instant Messaging Server Directory Structure" on page 53 for information on *im-svr-base*.

11 Modify *iim.instancevardir* **to point to the runtime directory for the new instance.**

For example:

On Solaris, change `/var/opt/SUNWiim/default` to `/var/opt/SUNWiim/xyz`.

On 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 *im-svr-base*`/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" on page 180 for instructions.

16 Start the new instance:

Solaris: **`/etc/opt/SUNWiim/xyz/imadmin start`**

Linux: **`/etc/opt/sun/im/xyz/imadmin start`**

◆ ◆ ◆ CHAPTER 2

Setting up and Launching Instant Messenger

This chapter contains information about configuring the web container and client systems to support Instant Messenger in the following sections:

- "Enabling Java Web Start" on page 47
- "Configuring Client Systems for Instant Messaging" on page 48
- "Launching Instant Messenger" on page 49

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. For instructions on installing Java Web Start, go to `http://java.sun.com/products/javawebstart`.

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 Sun Java System Web Server Enterprise Edition" on page 47
- "To Add the MIME Type to Apache Web Container" on page 48

▼ To Add the MIME Type to Sun Java System Web Server Enterprise Edition

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

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

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

Sun Java System 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

- **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 v7 as well as the recent versions of the Mozilla browser include the latest version of Java, while Internet Explorer does not. See the Sun Java System Instant Messaging 7 2006Q4 Release Notes for version requirements.

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 `http://www.java.sun.com/j2se`.

You can download and install Java Web Start from `http://www.java.sun.com/products/javawebstart`.

Launching Instant Messenger

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

- "Running Instant Messenger From a Web Browser" on page 49
- "Running Instant Messenger as a Standalone Application" on page 50

Running Instant Messenger From a Web Browser

Follow these instructions to run Instant Messenger as an applet within a web browser.

▼ To Run Instant Messenger as an Applet Within a Web Browser:

1. **Start the web browser.**

   For information on supported browsers, see the Sun Java System Instant Messaging 7 2006Q4 Release Notes.

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 running 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.

Running Instant Messenger as a Standalone Application

Follow these instructions to run Instant Messenger as a standalone application.

▼ To Run Instant Messenger as a Standalone Application

1 Start the web browser.

For information on supported browsers, see the Sun Java System Instant Messaging 7 2006Q4 Release Notes.

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 running 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" on page 160 for information on customizing the resource pages.

PART II

Administering Instant Messaging

- Chapter 3, "Instant Messaging Configuration File and Directory Structure Overview," provides information about the configuration files you use to administer Instant Messaging.
- Chapter 4, "Configuring Instant Messaging for High Availability (Solaris Only)," describes configuring Instant Messaging in a Sun Cluster environment.
- Chapter 5, "Enabling Single Sign-On (SSO) for Instant Messaging," describes SSO and how to configure it for Instant Messaging.
- Chapter 6, "Scaling an Instant Messaging Deployment Using Server Pooling," gives instructions on creating a server pool for a single domain to increase horizontal scalability.
- Chapter 7, "Optimizing an Instant Messaging Server Pool Using the Redirect Server," describes using the redirect server to optimize performance in an Instant Messaging server pool.
- Chapter 8, "Federating Deployment of Multiple Instant Messaging Servers," details how to support multiple domains in your Instant Messaging deployment.
- Chapter 9, "Administering Instant Messaging Components," describes how to administer the Instant Messaging server, multiplexor, Calendar agent, cluster agent, and watchdog.
- Chapter 10, "Using the Instant Messaging XMPP/HTTP Gateway," provides instructions on setting up and using the gateway.

- Chapter 11, "Managing Instant Messaging's LDAP Access Configuration," contains information on configuring LDAP for use with Instant Messaging.
- Chapter 12, "Securing Instant Messaging Using TLS and Legacy SSL," provides information about and how Instant Messaging uses SSL and TLS to ensure security.
- Chapter 13, "Managing Logging for Instant Messaging," describes configuring logging for Instant Messaging components and XMPP.
- Chapter 14, "Administering Instant Messaging End Users," provides instructions for disabling end user access to Instant Messenger, registering new users, using LDAP to store user properties, and assigning Instant Messaging and presence services to end users.
- Chapter 15, "Managing Instant Messenger," describes how to customize and administer Instant Messenger.
- Chapter 16, "Using Calendar Pop-up Reminders," describes how to configure the Instant Messaging server, Calendar agent, Calendar Server, and Instant Messenger to enable Calendar pop-up reminders.
- Chapter 17, "Managing Instant Messaging and Presence Policies," describes how to manage administrator and end user privileges, especially with policies set in Sun Java™ System Access Manager.
- Chapter 18, "Managing Archiving for Instant Messaging," explains how to manage and configure the Instant Messaging archive.
- Chapter 19, "Troubleshooting and Monitoring Instant Messaging," lists the common problems that might occur during installation and deployment of Instant Messaging and provides instructions for using the monitoring agent.

◆ ◆ ◆ CHAPTER 3

Instant Messaging Configuration File and Directory Structure Overview

This chapter provides information about 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 chapter describes the Instant Messaging server directory structure and the properties files used to store Instant Messaging operational data and configuration information in the following sections:

- "Instant Messaging Server Directory Structure" on page 53
- "Instant Messaging Server Configuration File" on page 55
- "Instant Messaging Data" on page 55

Instant Messaging Server Directory Structure

"Instant Messaging Server Directory Structure" on page 53 shows the platform-specific directory structure for the Instant Messaging server.

TABLE 3–1 Instant Messaging server directories

Description	Solaris Location	Linux Location
Program Files 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.	Instant Messaging Installation Directory (*im-svr-base*) The default value for the Installation Directory is: `/opt/SUNWiim`	Instant Messaging Installation Directory (*im-svr-base*) The default value for the Installation Directory is: `/opt/sun/im`

TABLE 3–1 Instant Messaging server directories *(Continued)*

Description	Solaris Location	Linux Location
Server Configuration files 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.	Instant Messaging Configuration Directory (*im-cfg-base*) The default value for the Configuration Directory is: `/etc/opt/SUNWiim /default/config` For convenience, the installer creates a symbolic link from `/etc/opt/SUNWiim /default/config` to `/opt/SUNWiim/config`. 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" on page 44 for more information.	Instant Messaging Configuration Directory (*im-cfg-base*) The default value for the Configuration Directory is: `/etc/opt/sun/im /default/config` For convenience, the installer creates a symbolic link from `/etc/opt/sun/im /default/config` to `/opt/sun/im/config`. 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" on page 44 for more information.
Runtime Directory Contains Instant Messaging server data. These files include the configurable directory for the files generated by the server at runtime. It includes the end user data in the database directory. It also contains the server, multiplexor, Calendar agent, and XMPP service log files, in the `log` directory.	Instant Messaging Runtime Directory (*im-runtime-base*) The default value for the Runtime Directory is: `/var/opt/SUNWiim/default` 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" on page 44 for more information.	Instant Messaging Runtime Directory (*im-runtime-base*) The default value for the Runtime Directory is: `/var/opt/sun/im/default` 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" on page 44 for more information.

TABLE 3–1 Instant Messaging server directories *(Continued)*

Description	Solaris Location	Linux Location
Database 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.	Instant Messaging Database Directory (*im-db-base*) The default value for the Database Directory is: `/var/opt/SUNWiim/default/db` 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" on page 44 for more information.	Instant Messaging Database Directory (*im-db-base*) The default value for the Database Directory is: `/var/opt/sun/im/default/db` 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" on page 44 for more information.
Instant Messenger resources. 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.	Instant Messaging Resource directory (*im-svr-base*`/html`) The default value for the Resource Directory is: `/opt/SUNWiim/html`	Instant Messaging Resource directory (*im-svr-base*`/html`) The default value for the Resource Directory is: `/opt/sun/im/html`

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 Appendix A, "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 (*im-db-base*), 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 4

Configuring Instant Messaging for High Availability (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 Solaris only. This chapter describes an Instant Messaging HA configuration using Sun Cluster software. See "HA Related Documentation" on page 71 for more information about scalable and failover data services provided by Sun Cluster.

This chapter describes how to configure an Instant Messaging HA service, including:

- "Instant Messaging HA Overview" on page 57
- "Setting Up HA for Instant Messaging" on page 60
- "Stopping, Starting, and Restarting the Instant Messaging HA Service" on page 68
- "Stopping, Starting, and Restarting Instant Messaging Components in a Deployment with Sun Cluster" on page 68
- "Managing the HA RTR File for Instant Messaging" on page 69
- "Removing HA for Instant Messaging" on page 70
- "HA Related Documentation" on page 71

Instant Messaging HA Overview

You use Sun 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" on page 58
- "Instant Messaging HA Configuration Permission Requirements" on page 58
- "Instant Messaging HA Configuration Terms and Checklist" on page 59

Before you begin, you should be familiar with general HA concepts, and Sun Cluster software in particular. For more information, see "HA Related Documentation" on page 71.

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
Solaris 9 OS	All versions of Solaris 9 OS are supported. Solaris 9 OS requires Sun Cluster 3.0 U3 at a minimum. Solaris 9 OS includes Solaris Logical Volume Manager (LVM).
Solaris 10 OS	All versions of Solaris 10 OS are supported.
Sun Cluster 3.1	Sun Cluster software must be installed and configured on all nodes in the cluster. To install Sun Cluster, use the Communications Suite installer by following the installation process in the *Sun Java Communications Suite 5 Installation Guide*. After you install the Sun Cluster software, you must configure the cluster. For information, refer to the *Sun Cluster System Administration Guide for Solaris OS*. For related documentation, see "HA Related Documentation" on page 71. **Sun Cluster Patches** For Solaris 9 and 10, you can download patches from SunSolve Online.
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
/global/im	Global file system mount point used with a cluster file system or HAStoragePlus.	
/local/im	Local directory to use as a mount point for the shared disk if you are using HAStoragePlus.	
im-logical-host	Logical host name	
im-logical-host-ip	Logical host IP numeric address	
im-node–1	Node 1 FQDN	
im-node–2	Node 2 FQDN	
im-resource-group	Instant Messaging resource group	
im-resource-group-store	Instant Messaging storage resource	
im-resource	Instant Messaging resource	
im-runtime-base (Includes *im-runtime-base*`/db` and *im-runtime-base*`/logs`)	For the location of the runtime directory (which includes the database and log subdirectories), select global, shared partitions. For example: ■ Instant Messaging runtime directory (*im-runtime-base*): */global/im*`/var/opt/SUNWiim/default` ■ Database subdirectory (*im-db-base*): */global/im*`/var/opt/SUNWiim/default/db` ■ Log subdirectory: */global/im*`/var/opt/SUNWiim/default/logs` See "Instant Messaging Server Directory Structure" on page 53 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" on page 60
- "Preparing Each Node in the Cluster" on page 60
- "Selecting the Installation Directory (*im-svr-base*)" on page 61
- "Installing Sun Java System Products and Packages" on page 61
- "Configuring the HA Environment" on page 62
- "Configuring the Logical Host" on page 65
- "Registering and Activating the Storage Resource" on page 66
- "Registering the Resource Type and Creating a Resource" on page 66
- "Verifying the Instant Messaging HA Configuration" on page 67
- "Troubleshooting the Instant Messaging HA Configuration" on page 67

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" on page 38 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 *Sun Java Communications Suite 5 Installation Guide*.

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 *n*
Sun 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.

TABLE 4–3 Products and Packages Required for a Multiple Node Instant Messaging HA Configuration *(Continued)*

Sun 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" on page 62
- "To Configure HA on Node *n* Using a Local Disk for Configuration Files and Binaries" on page 63

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" on page 64
- "To Configure HA on Node *n* Using a Shared Disk for Configuration Files and Binaries" on page 65

▼ 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*)" on page 61 for specific instructions on choosing an installation directory.

See Table 4–3 for a list of required products and packages for HA. Refer to the *Sun Java Communications Suite 5 Installation Guide* 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:

a. **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.

b. **Use the `mount` command to mount the disk on */local/im/im-runtime-base*.**

3 **Run the `configure` utility.**

See Chapter 1, "Configuring Instant Messaging After Installation," for instructions.

4 **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.

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.

7 **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" on page 62).

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*)" on page 61 for specific instructions.

   See Table 4–3 for a list of required products and packages for HA. Refer to the *Sun Java Communications Suite 5 Installation Guide* for specific instructions.

2. **Run the `configure` utility.**

   See Chapter 1, "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*)" on page 61 for specific instructions.

   See Table 4–3 for a list of required products and packages for HA. Refer to the*Sun Java Communications Suite 5 Installation Guide* for specific instructions.

2. **Create a soft link from `/etc/opt/SUNWiim` that points to `/`*global*`/`*im*`/etc/opt/SUNWiim`.**

3. **Run the `configure` utility from the global directory where you installed Instant Messaging (*/global/im/im-svr-base*`/configure`).**

   See Chapter 1, "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" on page 64).

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" on page 71 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 Sun Cluster `scswitch` command, refer to the *Sun Cluster Reference Manual for Solaris OS.*

▼ To Start the Instant Messaging HA Service

- **Type the following at the command line:**

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

▼ To Stop the Instant Messaging HA Service

- **Type the following at the command line:**

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

▼ To Restart the Instant Messaging HA Service

- **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 Sun 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.iim`, 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 RTR File Parameters" on page 69
- "Customizing the RTR File for Instant Messaging" on page 70

Instant Messaging RTR File Parameters

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

TABLE 4–4 `SUNW.iim` Extension Properties

Extension Property	Default	Description
`Server_Root`	If you are using a local disk to store configuration files and binaries: *im-svr-base* 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, *im-svr-base* is `/opt/SUNWiim` on 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.

TABLE 4–4 SUNW.iim Extension Properties *(Continued)*

Extension Property	Default	Description
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 (retry_count) is exceeded during the configured retry interval (retry_interval). See the *Sun Cluster Reference Manual for Solaris OS* 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 the *Sun Cluster Reference Manual for Solaris OS* for more information on the contents of resource type registration files and instructions on customizing values for extension properties.

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

- Sun Java Enterprise System 2005Q4 Technical Overview.
- *Sun Java Communications Suite 5 Installation Guide* describes the Communications Suite installer (and uninstaller) and the supported installation scenarios.
- *Sun Java Enterprise System 5 Release Notes for UNIX* provide current information about the Sun Java Enterprise System product.
- *Sun Cluster Concepts Guide for Solaris OS* provides a general background about Sun Cluster software, data services, and terminology resource types, resources, and resource groups.
- *Sun Cluster Data Services Planning and Administration Guide for Solaris OS* provides general information on planning and administration of data services.

- *Sun Cluster System Administration Guide for Solaris OS* provides the software procedures for administering a Sun Cluster configuration.
- *Sun Cluster Reference Manual for Solaris OS* describes the commands and utilities available with the Sun Cluster software, including commands found only in the `SUNWscman` and `SUNWccon` packages.
- *Sun Java Communications Suite 5 Deployment Planning Guide* provides further information about how HA is implemented in Instant Messaging.

◆ ◆ ◆ CHAPTER 5

Enabling Single Sign-On (SSO) for Instant Messaging

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. The Sun Java™ System Access Manager is the official gateway used for SSO for Sun Java System 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 the Sun Java System 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. This chapter describes using Access Manager to enable SSO for Instant Messaging in the following sections:

- "SSO Limitations and Notices" on page 73
- "Configuring Instant Messaging to Support Access Manager-Based SSO and Policies" on page 74
- "Troubleshooting SSO for Instant Messaging" on page 75

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.

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
iim_server.usesso	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 `configure` utility. Otherwise, the default value is `0`.
iim_server.ssoprovider	Specifies the class implementing the `com.sun.im.provider.SSOProvider` interface. If *iim_server.usesso* 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" on page 41 for instructions.

3. **Run the `configure` utility.**

   See "To Configure Instant Messaging After Installation" on page 39 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:**

   a. **Open `iim.conf` and find the *iim.policy.module* parameter.**

   b. **Set the parameter:**

      ```
      iim.policy.module = "identity"
      ```

6. **Restart the Instant Messaging server:**

   ```
   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" on page 41 for information.

Ensure that the *im_server.usesso* parameter is not set to `0` in `iim.conf`. See Table 5–1 for information on this parameter. If it is set to `0`, complete the steps in "To Enable SSO for Instant Messaging" on page 74.

If you are unable to log into Instant Messaging directly, look in `xmppd.log` 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, use 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.

◆ ◆ ◆ CHAPTER 6

Scaling an Instant Messaging Deployment Using Server Pooling

Server pooling allows 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. This chapter provides information on server pooling in the following sections:

- "Overview of Server Pooling for Instant Messaging" on page 77
- "Availability in an Instant Messaging Server Pool" on page 78
- "Configuring Server-to-Server Communication Between Instant Messaging Servers in a Server Pool" on page 78
- "Adding a New Node to an Existing Instant Messaging Deployment" on page 81
- "Securing a Multi-node Deployment" on page 81

For information on the load balancing and the redirect server, see Chapter 7, "Optimizing an Instant Messaging Server Pool Using the Redirect Server." The procedures in this chapter assume you have already installed Instant Messaging on the hosts in your server pool. In addition, 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.

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.

In order 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
- Multiparty conference messages

The following information is not replicated:

- One on one chat messages
- Presence subscriptions and notifications

In addition, 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 Policies Using Access Control Files" on page 193 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.

Table 6–1 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 Appendix A, "Instant Messaging Configuration Parameters in `iim.conf`."

TABLE 6–1 Example Configuration Information for Two Instant Messaging Servers in a Server Pool

Parameter in `iim.conf`	Value for Server A	Value for Server B	Notes
iim_server.serverid	`iimA.siroe.com`	`iimB.siroe.com`	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.
iim_server.password	`secretforiimA`	`secret4iimB`	
iim_server.coservers	`coserver1`	`coserver1`	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 may contain multiple, comma-separated values.
iim_server.domainname	`siroe.com`	`siroe.com`	Peer servers within a server pool share the same default domain.
iim_server.coserver1.host	`iimB.siroe.com:5269`	`iimA.siroe.com:5269`	The hostname and port number of the peer server in the server pool.
iim_server.coserver1.serverid	`iimB.siroe.com`	`iimA.siroe.com`	The server ID (*iim_server.serverid*) of the peer server in the server pool.
iim_server.coserver1.password	`secret4iimB`	`secretforiimA`	The password (*iim_server.password*) of the peer server in the server pool.
iim_server.coserver1.domain	`siroe.com`	`siroe.com`	Peer servers within a server pool share the same default domain.

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

1 **Gather the information listed in Table 6–1.**

2 **Change to *im-cfg-base* on the server `iimA.siroe.com`.**

See "Instant Messaging Server Directory Structure" on page 53 for instructions on locating *im-cfg-base*.

3 **Open `iim.conf`.**

See Appendix A, "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" on page 80 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" on page 81 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" on page 250 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 7

Optimizing an Instant Messaging Server Pool 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" on page 83
- "Configuring an Instant Messaging Server Instance as a Redirect Server" on page 85
- "Administering the Instant Messaging Redirect Server" on page 88
- "Creating and Managing the Instant Messaging Redirect Table Using the `rdadmin` Utility" on page 90
- "Instant Messaging Redirect Server Physical Host Monitoring" on page 91
- "Instant Messaging Redirect Server Best Practices and Troubleshooting" on page 92

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 `rdadmin` 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

TABLE 7–1 Redirect Server Configuration Parameters in `iim.conf` *(Continued)*

iim_server.redirect.provider	None	Comma-separated list of redirect provider names or classes that implement the `com.sun.im.provider.Redirector` interface. Any value for this parameter defines the server instance as a redirect server. Supported values include `db`, `roundrobin`, `regex`, and class names that implement the `com.sun.im.provider.Redirector` interface.
iim_server.redirect.to	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 *iim_server.redirect.to.nodename.host*.
iim_server.redirect.to.nodename.host	None	Where *nodename* is the name of the node as it exists in *iim_server.redirect.to*. This attribute is required for *nodename* to be used by the redirect server.
iim_server.redirect.to.nodename.usessl	False	If true, then *nodename* is configured to use legacy SSL. See "Overview of Using TLS and Legacy SSL in Instant Messaging" on page 123 for more information.
iim_server.redirect.db.users	*im-db-base*`/redirect.db`	Name and location of the redirect database.
iim_server.redirect.db.partitions	*im-cfg-base*`/redirect.partitions`	Name and location of the redirect partitions file.
iim_server.redirect.db.partitionsize	5000	The maximum number of users in a partition.
iim_server.redirect.roundrobin.partitions	*im-cfg-base*`/redirect.partitions`	Name and location of the redirect partitions file.

TABLE 7–1 Redirect Server Configuration Parameters in `iim.conf` *(Continued)*

iim_server.redirect.pollfrequency	The interval between connections made by the redirect server to the hosts defined in the `redirect.hosts` 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

Before You Begin 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 in Table 7–1 above.**

2. **Open `iim.conf`.**

   See Appendix A, "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" on page 88
- "Instant Messaging Redirect Server Logging" on page 88
- "Setting the Partition Size for the Instant Messaging Redirect Server" on page 88
- "Specifying the List of Partitions for the Instant Messaging Redirect Server" on page 89

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" on page 99 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 Chapter 13, "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.partitionsize* 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

EXAMPLE 7–2 Redirect.partitions File Configuration *(Continued)*

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 Appendix A, "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" on page 92
- "Instant Messaging Redirect Server Supported Clients" on page 92
- "Using Redirect Server and Storing User Properties in LDAP" on page 92
- "Determining the Partition Size for the Redirect Database" on page 92
- "Using a Redirect Server as a Partition Host" on page 93

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 8

Federating Deployment of Multiple 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`.

Caution – 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.

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 Appendix A, "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 `iim.conf`	**Value for Server** `iim.company22.com`	**Value for Server** `iim.i-zed.com`
iim_server.serverid	`Iamcompany22`	`iami-zed`
iim_server.password	`secretforcompany22`	`secret4i-zed`
iim_server.coservers	`coserver1`	`coserver1`
iim_server.domainname	`iim.company22.com`	`iim.i-zed.com`
iim_server.coserver1.host	`iim.i-zed.com:5269`	`iim.company22.com:5269`
iim_server.coserver1.serverid	`Iami-zed`	`Iamcompany22`
iim_server.coserver1.password	`secret4i-zed`	`secretforcompany22`
iim_server.coserver1.domain	`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" on page 53 for instructions on locating *im-cfg-base*.

3. **Open `iim.conf`.**

   See "`iim.conf` File Syntax" on page 250 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

Administering Instant Messaging Components

This chapter explains how to administer the Instant Messaging components (server, multiplexor, Calendar agent, cluster agent, and watchdog) and perform other administrative tasks, such as changing configuration parameters and creating backups.

This chapter contains the following sections, which describe the various administrative tasks in Instant Messaging:

- "Stopping, Starting, Refreshing, and Checking Instant Messaging Components" on page 99
- "Changing Instant Messaging Server and Multiplexor Configuration Parameters" on page 104
- "Backing Up Instant Messaging Data" on page 105

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, and watchdog).
- 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 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.

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, and watchdog depending on which components are enabled.

▼ 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
```

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, and watchdog depending on which components are enabled.

▼ 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, and watchdog, terminates all end user connections, and disconnects any inbound and outbound servers configured.

▼ 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, and watchdog, terminates all end user connections, and disconnects any inbound and outbound servers configured.

  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 Appendix A, "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 Appendix A, "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" on page 53 for instructions on locating *im-cfg-base*.

2. **Edit `iim.conf` using a text editor.**

3. **Save your changes.**

4. **Refresh the configuration 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 Chapter 15, "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" on page 105
- "Performing a Backup" on page 105
- "Restoring Backup Information" on page 105

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*). Default paths are described in "Instant Messaging Server Directory Structure" on page 53.

The Instant Messaging data is stored in the database directory (*im-db-base*). Defaults for *im-db-base* are also described in "Instant Messaging Server Directory Structure" on page 53.

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 back up 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" on page 53 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" on page 38 for information on this user. Permissions should be set as follows:

   - Files: `600` (indicating read and write permissions for owner only)
   - 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 10

Using the Instant Messaging 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 don't 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" on page 107
- "Configuring the Instant Messaging XMPP/HTTP Gateway" on page 108
- "Securing Communication Between the XMPP/HTTP Gateway and Instant Messaging Server Using StartTLS" on page 114
- "Managing Logging for the XMPP/HTTP Gateway" on page 115

Instant Messaging XMPP/HTTP Gateway Configuration Files

The XMPP/HTTP Gateway uses the following files for configuration:

- Gateway webapp 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" on page 114.
- Gateway configuration file (typically `httpbind.conf`). See "Configuring the Instant Messaging XMPP/HTTP Gateway" on page 108 for instructions on configuring the gateway. See Appendix B, "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" on page 115 for more information on configuring logging. See "XMPP/HTTP Gateway log4j Log Configuration File Syntax" on page 118 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 Appendix B, "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" on page 109
- "To Configure the Number of Concurrent Requests Handled by the XMPP/HTTP Gateway" on page 109
- "To Set the JEP 124 *hold* Attribute for Client Requests to the XMPP/HTTP Gateway" on page 110
- "To Specify the Allowed Client Inactivity Time for the XMPP/HTTP Gateway" on page 110
- "To Set the `content-type` HTTP Header for the XMPP/HTTP Gateway" on page 111
- "To Set the Round Trip Delay for the XMPP/HTTP Gateway" on page 111
- "To Set the Default Time Within Which the XMPP/HTTP Gateway Will Send a Response to the Client" on page 112
- "To Configure an XMPP/HTTP Gateway in a Instant Messaging Gateway Pool" on page 112
- "To Configure the List of Key IDs for Supported XMPP/HTTP Gateway Domains" on page 113
- "To Configure the Instant Messaging XMPP/HTTP Gateway to Use a Non-default Configuration File" on page 114

For instructions on configuring logging for the gateway, see "Managing Logging for the XMPP/HTTP Gateway" on page 115.

▼ 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. **Run the `configure` utility.**

   b. **Choose to deploy the gateway when prompted.**
      See Chapter 1, "Configuring Instant Messaging After Installation," for more information.

   c. **In `iim.conf`, set the *iim_agent.httpbind.enable* parameter to true.**
      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

Before You Begin 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" on page 275 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" on page 110 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

Before You Begin 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" on page 275 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" on page 275 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" on page 275 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" on page 275 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" on page 275 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" on page 275 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.nodeId* parameter to the full URL of the gateway.

The URL is used as the gateway's nodeId. This nodeId must be unique within the server pool. The gateway uses this nodeId 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 using the tools provided by the web or application server.

▼ To Configure the List of Key IDs for Supported XMPP/HTTP Gateway Domains

1 Open `httpbind.conf`.

See "`httpbind.conf` File Location" on page 275 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*" on page 279 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 Chapter 12, "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" on page 115
- "To Change the Location of the XMPP/HTTP Gateway Log Configuration File" on page 116
- "Linux: To Set the Location of the XMPP/HTTP Gateway Log File After Install or Upgrade" on page 116
- "To Change the Location of the XMPP/HTTP Gateway Log File" on page 117
- "To Use a Non-default Log File Location for the XMPP/HTTP Gateway" on page 117
- "To Set the XMPP/HTTP Gateway Logging Level" on page 117
- "XMPP/HTTP Gateway log4j Log Configuration File Syntax" on page 118

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 `httbind.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.gateway* parameter as follows:**

   `log4j.logger.gateway=OFF`

3 **To enable logging, set the** *log4j.logger.gateway* **parameter to the desired logging level.**

For example:

```
log4j.logger.gateway=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" on page 275 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** *log4.appender.appender_ID.file* **parameter to the location where log files are stored.**

▼ To Change the Location of the XMPP/HTTP Gateway Log File

Before You Begin 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** *log4.appender.appender_ID.file* **parameter to the location where log files are stored.**

▼ To Set the XMPP/HTTP Gateway Logging Level

Before You Begin 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.gateway* **parameter to the desired logging level.**

For example:

```
log4j.logger.gateway=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.gateway=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.gateway=ERROR, A1
# DEFAULT TO RollingFileAppender
log4j.appender.A1=org.apache.log4j.RollingFileAppender
# log4j.appender.A1.file=$(logdir)/gateway.log
log4j.appender.A1.file=/tmp/gatewaylog
log4j.appender.A1.append=true
log4j.appender.A1.maxBackupIndex=7
log4j.appender.A1.maxFileSize=5mb
log4j.appender.A1.layout=org.apache.log4j.PatternLayout
log4j.appender.A1.layout.ConversionPattern=[%d{DATE}] %-5p %c [%t] %m%n
```

◆ ◆ ◆ CHAPTER 11

Managing Instant Messaging's LDAP Access Configuration

This chapter describes how Instant Messaging uses LDAP in deployments with and without Access Manager in the following sections:

- "Overview of how Instant Messaging Uses LDAP" on page 119
- "Searching the Directory Anonymously" on page 120
- "Configuring Instant Messaging to Use LDAP Dynamic Groups" on page 121

Overview of how Instant Messaging Uses LDAP

All deployments of Instant Messaging require a directory server. In a deployment without Sun Java™ System 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 Sun Java System Portal Server, the Instant Messaging server uses the directory used by Sun Java System 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" on page 250.

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" on page 250 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" on page 250 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 Dynamic Groups

In the Sun Java System 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` objectclass.

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.

The following modifications need to be made to `iim.conf`:

▼ To Configure Instant Messaging to Use Dynamic Groups

1 **Open** `iim.conf`.

See "`iim.conf` File Syntax" on page 250 for instructions on locating and modifying `iim.conf`.

2 **Add the following three lines to** `iim.conf`**:**

```
iim_ldap.usergroupbynamesearchfilter=(|(&(|
(objectclass=groupofuniquenames)
(objectclass=groupofurls)))
(cn={0}))(&(objectclass=inetorgperson)
(cn={0})))

iim_ldap.groupbrowsefilter=(|
(objectclass=groupofuniquenames)
(objectclass=groupofurls))

iim_ldap.groupclass=groupOfUniqueNames,groupOfURLs
```

Do not include line breaks within a single line. The attribute and objectclass 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.

◆ ◆ ◆ CHAPTER 12

Securing Instant Messaging Using TLS and Legacy SSL

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

- "Overview of Using TLS and Legacy SSL in Instant Messaging" on page 123
- "Setting Up TLS for the Instant Messaging Server" on page 124
- "Activating TLS on the Instant Messaging Server" on page 125
- "Setting Up Legacy SSL for the Multiplexor and Instant Messenger" on page 128
- "Invoking the Secure Version of Instant Messenger" on page 134

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 "Introduction to Certificates and SSL" in *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" on page 100 for details.

5. Obtaining 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 `iim.conf`.

   For instructions see "Activating TLS on the Instant Messaging Server" on page 125.

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* of the *Sun Java System Application Server Enterprise Edition 8.2 Administration Guide*. Step 6 is described in "Activating TLS on the Instant Messaging Server" on page 125.

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" on page 124. 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 `iim.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
iim_server.sslkeystore	None	Contains the relative path and filename for the server's Java keystore (JKS). For example: `/`*im-cfg-base*`/server-keystore.jks`

TABLE 12–1 Instant Messaging Server TLS Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim_server.keystorepasswordfile	`sslpassword.conf`	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:`*password* Where *password* is the password protecting the keystore.
iim_server.requiressl	`false`	If true, the server will terminate any connection that does not request a TLS connection after the initial stream session is set up.
iim_server.trust_all_cert	`false`	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.

▼ 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.

Before You Begin 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" on page 124.

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=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.requiressl=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 Chapter 8, "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.requiressl=true
   ```

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

   If you set *iim_server.requiressl* 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" on page 134.

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=/var/im/server_keystore.jks
iim_server.keystorepasswordfile=/var/im/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" on page 129.
2. "Installing the Certificate" on page 130.
3. "Enabling Legacy SSL Between the Multiplexor and Instant Messenger" on page 131.
4. "Activating TLS on the Instant Messaging Server" on page 125.

5. "Invoking the Secure Version of Instant Messenger" on page 134.

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 Installation 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 Chapter 9, "Configuring Security," in *Sun Java System Application Server Enterprise Edition 8.2 Administration Guide* for Application Server and Chapter 6, "Certificates and Keys," in *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 Chapter 9, "Configuring Security," in *Sun Java System Application Server Enterprise Edition 8.2 Administration Guide* for Application Server and Chapter 6, "Certificates and Keys," in *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://docs.sun.com`

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 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" on page 53 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" on page 134.

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" on page 44 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.certnickname	*Multiplexor-Cert*	This value should contain the name of the certificate you entered while installing the certificate. The certificate name is case-sensitive.

TABLE 12–2 Instant Messaging Multiplexor SSL Parameters *(Continued)*

Parameter	Default Value	Description
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:`*password* Where *password* 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" on page 250 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 '-server'
! 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.secconfigdir = "/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" on page 125 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.jnlp` from your web browser. If you are using TLS, access the client normally. See "Invoking Instant Messenger" on page 158 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

Managing Logging for Instant Messaging

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" on page 135
- "Instant Messaging Log File Location" on page 136
- "Instant Messaging Component Logging Levels" on page 136
- "Managing Instant Messaging Logging Using Log4j" on page 137
- "Configuring Logging for Instant Messaging Components Using `iim.conf` Parameters" on page 143
- "Administering Logging for Instant Messenger" on page 145

For information on logging for the XMPP/HTTP Gateway, see "Managing Logging for the XMPP/HTTP Gateway" on page 115. In addition, you can collect logging data for Instant Messenger on demand. See "Administering Logging for Instant Messenger" on page 145 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" on page 115. For information on setting up logging for Instant Messenger see "Administering Logging for Instant Messenger" on page 145.

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 Apache Logging Services website (`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 *im-runtime-base*`/log`. See "Instant Messaging Server Directory Structure" on page 53 for information on locating *im-runtime-base*.

If you are using log4j for log file generation in your deploymentthe 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" on page 138
- "Instant Messaging Log4j Log File Syntax" on page 138
- "Log4j Log Levels for Instant Messaging Components" on page 141

- "To Specify the Location of the Log4j Configuration File (`Log4j.conf`)" on page 141
- "To Enable or Disable Log4j Logging for an Instant Messaging Component" on page 142
- "To Set Log4j Log Levels for Instant Messaging" on page 142
- "To Specify the Maximum Log4j Log File Size for Instant Messaging Components" on page 143

The logging levels described in "Instant Messaging Component Logging Levels" on page 136 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 Apache Logging Services website (`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 *iim.log4j.config* parameter in `iim.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 `iim.conf` to generate non-log4j style logs.

See "Instant Messaging Server Directory Structure" on page 53 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" on page 136.
- 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 Jabber Stream Objects website for more information.
`genredirect`	Generates `genredirect.log` which contains information for the redirect database creation tool.

- `A#`, for example A1, are appender IDs.

EXAMPLE 13–1 Log4j Template File

```
log4j.logger.xmppd=INFO, A1
# 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
```

EXAMPLE 13–1 Log4j Template File *(Continued)*

```
# 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=DEBUG, 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
```

EXAMPLE 13–1 Log4j Template File *(Continued)*

```
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 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" on page 136.

▼ To Specify the Location of the Log4j Configuration File (`Log4j.conf`)

1. **Open `iim.conf`.**

   See "`iim.conf` File Location" on page 249 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 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" on page 142 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 following location:

   im-svr-base`/lib`

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.

3. **Save and close `log4j.conf.template`.**

4. **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 following location:

   im-svr-base`/lib`

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	`xmppd.log`	*iim.log.iim_server.severity*
Multiplexor	`mux.log`	*iim.log.iim_mux.severity*
Calendar agent	`agent-calendar.log`	*iim.log.agent-calendar.severity*
Watchdog	`iim_wd.log`	*iim.log.iim_wd.severity*

The configuration parameters can have the following values:

- `fatal`
- `error`
- `warning`
- `info`
- `debug`

See "Instant Messaging Component Logging Levels" on page 136 for information on the details logged for each logging level.

In addition, logging configuration in deployments with Sun Java™ System Access Manager is determined by the *com.iplanet.services.debug.level* property. You set this property in the `AMConfig.properties` file on the Sun Java System 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 Sun Java System Portal Server desktop log file (`desktop.debug`) and archive log files (`IMArchiveSearch.log` and `IMArchiveSubmit.log`) are stored in the following locations:

- Solaris: `/var/opt/SUNWam/debug`
- Linux: `/var/opt/sun/am/debug`

▼ To Set Log Levels for Instant Messaging Components Using `iim.conf` Parameters

- **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" on page 250 for instructions on locating and modifying `iim.conf`. For more information on the watchdog, see "Managing the Watchdog Process" on page 243. For more information on the Calendar agent, see Chapter 16, "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.iplanet.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" on page 146 for details.

5. **Save and close the `im.jnlp` file.**
6. **If you are using Sun Java System Application Server or Sun Java System Web Server, redeploy the resource files as described in "Redeploying Resource Files" on page 180.**
7. **Relaunch Instant Messenger.**
8. **Locate the log file.**

   By default the log file is stored as *usr_home*`/.sunmsgr/messenger.log`.

Next Steps 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" on page 180.

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" on page 145 for instructions on setting the *logconfig* parameter and generating Instant Messenger logs.

TABLE 13–3 Instant Messenger Logging Options for `messenger.log`

logconfig value	`messenger.log` 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.

TABLE 13–3 Instant Messenger Logging Options for `messenger.log` *(Continued)*

`XMPPTRAFFIC`	Client to server communication only.
`CLIENT`	Client application (Instant Messenger) details only.

◆ ◆ ◆ CHAPTER 14

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 Sun Java™ System 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" on page 189, then the Access Manager is used for provisioning Instant Messaging end users. For more information, see the *Sun Java Communications Suite 5 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.

This chapter contains the following sections:

- "Disabling End User Access to Instant Messenger" on page 150
- "Registering New Instant Messaging Users" on page 150
- "Storing Instant Messaging User Properties in LDAP" on page 154

- "Assigning Instant Messaging and Presence Services to End Users" on page 154

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" on page 250 for instructions on locating and modifying `iim.conf`.

2 **Modify the following values as shown:**

```
iim_ldap.useidentityadmin="true"
iim_server.usesso=1      The 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" on page 102 for more information. If you are using Instant Messaging in an HA environment, do not use `imadmin`, instead use the Sun 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 Sun Java System 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" on page 151
- "Customizing Instant Messenger to Allow New User Registration" on page 152
- "Registering as a New Instant Messaging User" on page 153

See Chapter 15, "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
iim.register.enable	If `TRUE`, the server allows new Instant Messaging end users to register themselves (add themselves to the directory) using Instant Messenger.
iim_ldap.register.basedn	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"`
iim_ldap.register.domain	The domain to which new users will be added. For example, `directory.siroe.com`.

▼ To Configure the Instant Messaging Server to Allow New User Registration

1. **Open `iim.conf`.**
   See "`iim.conf` File Syntax" on page 250 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" on page 39 for instructions.

10. **If you are usingSun Java System Application Server or Sun Java System Web Server, redeploy the resource files.**

    See "Redeploying Resource Files" on page 180 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 Sun Java System 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" on page 250 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 Sun Java System 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" on page 250 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

Managing Instant Messenger

This chapter describes how to customize and administer Instant Messenger in the following sections:

- "Configuring Instant Messenger" on page 157
- "Invoking Instant Messenger" on page 158
- "Changing the Codebase" on page 159
- "Changing the Web Container Port" on page 160
- "Customizing Instant Messenger" on page 160
- "Modifying How Client Users Search for Contacts" on page 174
- "Administering Conference Rooms and News Channels" on page 175
- "Modifying Instant Messenger Proxy Settings" on page 176
- "Controlling the Exposed Messenger Feature Set" on page 177
- "Instant Messenger Data Stored in the End User's System" on page 178
- "Redeploying Resource Files" on page 180

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 Chapter 2, "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 Using a Direct URL" on page 158
- "To Invoke Instant Messenger From the Command-Line (Solaris Only)" on page 159
- "To Invoke Instant Messenger Using a Desktop Shortcut" on page 159

▼ To Invoke Instant Messenger Using a Direct URL

- **Enter the following URL in your web browser to invoke Instant Messenger:**

  `http://`*webserver*`:`*webserverport*`/`*path*`/`*filename*

  In this URL,

webserver	Specifies the name of the web container on which you have installed the Instant Messenger resources.
webserverport	(Optional) Specifies the web container port. The default value is `80`.
path	(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.
filename	Specifies the Instant Messenger file to use: `index.html` - This file is provided with the product. The file contains links to `im.jnlp` and `im.html` which launch the Java Web Start and Java Plug-in versions of Instant Messenger respectively. `im.jnlp` - The `.jnlp` file to launch only the Java Web Start version of Instant Messenger. `im.html` - The web page to launch only the Java Plug-in version of Instant Messenger.

▼ To Invoke Instant Messenger From the Command-Line (Solaris Only)

- **Type the following at the command-line:**

  `javaws_cmd` *URL*

  See "To Invoke Instant Messenger Using a Direct URL" on page 158 for information about constructing the URL.

▼ To Invoke Instant Messenger Using a Desktop Shortcut

- **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.

If you want to change the codebase after you have deployed the resource files you will need to:

- Modify the template files to point to the new URL. See "To Change the Codebase in the Resource Templates" on page 160.
- 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" on page 39 for instructions.
- Redeploy the resource files. See "Redeploying Resource Files" on page 180 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" on page 160 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" on page 159 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" on page 180 for redeployment instructions.)

You can customize Instant Messenger to meet your requirements in the following ways:

- "Instant Messenger Resource Files" on page 160
- "Customizing the `index.html` and `im.html` Files" on page 162
- "Launching Instant Messenger Using Sun Java System Access Manager SSO" on page 163
- "Customizing the Application (Java Web Start)" on page 164
- "Rebranding Instant Messenger" on page 171
- "Customizing User Name and Group Name Display" on page 172

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

TABLE 15–1 Instant Messenger Resource Files in *im-svr-base*/html *(Continued)*

File	Description	Customizable?
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 Sun Java™ System 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 Sun Java System 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 imwebex.jar msgrinstall.jar	Instant Messenger Online Help directory.	No

Customizing the index.html and im.html Files

If you are using Instant Messenger in a deployment without Sun Java System 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 Sun Java System Access Manager SSO

To launch the Instant Messenger client using single sign-on (SSO) with Sun Java System Access Manager use `IMLaunch.jsp`. This file is in the resource directory.

Sun Java System Access Manager and Instant Messenger must be configured to use the same web container to enable SSO.

To launchInstant Messenger enter the following in a web browser:

codebase`/IMLaunch.jsp?server=`*multiplexor-hostname*`:`*muliplexor-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 mulitplexor. For example, `http://www.company22.com`.

muliplexor-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`" on page 165.

  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 Sun Java System Instant Messenger</description>
```

EXAMPLE 15–1 Sample `im.jnlp` File *(Continued)*

```
    <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

TABLE 15–3 Emoticons *(Continued)*

File Name	Description
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

TABLE 15–8 Presence Icons - Contact List *(Continued)*

File Name	Description
`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_weave_purple.gif
bgplt_tex_brown.gif	bgplt_tex_weave_ruby.gif
bgplt_tex_bubble_blue.gif	bgplt_tex_white.gif
bgplt_tex_bubble_brown.gif	bg_tex_bubble_blue.gif
bgplt_tex_bubble_green.gif	bg_tex_bubble_brown.gif
bgplt_tex_bubble_grey.gif	bg_tex_bubble_green.gif
bgplt_tex_bubble_orange.gif	bg_tex_bubble_grey.gif
bgplt_tex_bubble_purple.gif	bg_tex_bubble_orange.gif
bgplt_tex_bubble_ruby.gif	bg_tex_bubble_purple.gif
bgplt_tex_crackle_blue.gif	bg_tex_bubble_ruby.gif
bgplt_tex_crackle_green1.gif	bg_tex_crackle_blue.gif
bgplt_tex_crackle_grey.gif	bg_tex_crackle_green1.gif
bgplt_tex_crackle_olive.gif	bg_tex_crackle_grey.gif
bgplt_tex_crackle_orange.gif	bg_tex_crackle_olive.gif
bgplt_tex_crackle_purple.gif	bg_tex_crackle_orange.gif
bgplt_tex_crackle_ruby.gif	bg_tex_crackle_purple.gif
bgplt_tex_gradation_blue.gif	bg_tex_crackle_ruby.gif
bgplt_tex_gradation_brown.gif	bg_tex_gradation_blue.gif
bgplt_tex_gradation_green.gif	bg_tex_gradation_brown.gif
bgplt_tex_gradation_grey.gif	bg_tex_gradation_green.gif
bgplt_tex_gradation_orange.gif	bg_tex_gradation_grey.gif
bgplt_tex_gradation_purple.gif	bg_tex_gradation_orange.gif
bgplt_tex_gradation_ruby.gif	bg_tex_gradation_purple.gif
bgplt_tex_green.gif	bg_tex_gradation_ruby.gif
bgplt_tex_orange.gif	bg_tex_weave_blue.gif
bgplt_tex_pink.gif	bg_tex_weave_brown.gif
bgplt_tex_purple.gif	bg_tex_weave_green.gif
bgplt_tex_weave_blue.gif	bg_tex_weave_grey.gif
bgplt_tex_weave_brown.gif	bg_tex_weave_orange.gif
bgplt_tex_weave_green.gif	bg_tex_weave_purple.gif
bgplt_tex_weave_grey.gif	bg_tex_weave_ruby.gif
bgplt_tex_weave_orange.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" on page 250 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" on page 250 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
   ```

   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" on page 250 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)" on page 164.

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" on page 250 for instructions on locating and modifying `iim.conf`.

2 **Modify the `iim_ldap.usergroupbynamesearchfilter` 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" on page 250 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 Chapter 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
server	`127.0.0.1`	The Instant Messaging server host and port.
debug	`FALSE`	If this parameter is set to true, the applet records all the task performed on java console.

TABLE 15–12 Instant Messenger Applet Parameters *(Continued)*

Parameter	Default Value	Description
uid		This parameter is used for SSO.
token		This parameter contains the SSO token and is used for auto-logon.
secure	FALSE	Indicates to the Instant Messenger that it is run in SRA mode. It displays a security indicator.
usessl	FALSE	Tells Instant Messenger to use legacy SSL when connecting to multiplexor.
allow_alert_only	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.
allow_attachments	TRUE	Allows file attachment and transfer.
enable_moderator	TRUE	If set to true, enables the moderated conference feature.
messenger_bean		This parameter contains a list of messenger beans to be used. You can enter multiple factory class names with each separated by a comma.
domain	null	This parameter is used in multidomain Sun Java System Access Manager deployments. The value of this parameter should be the logical domain name of the organization in which this end user is present.
gateway_url	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 *user-domain*. 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
client.password.encoded	false	Determines whether or not the user password is encoded (for use with SSO). If the value for this parameter is `true`, the encoded password is stored as the value for the *net.password* parameter.
net.nms	`127.0.0.1`	Instant Messaging server host name and port.
net.nmsn (Where the trailing *n* is a digit used to distinguish one entry from another)		The secondary servers' host names and port numbers.
net.user		The default user id.
net.password		The encoded user password that enables auto-logon.

Redeploying Resource Files

If you are using Sun Java System Application Server or Sun Java System 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 Sun Java System Application Server or Sun Java System 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 Chapter 1, "Configuring Instant Messaging After Installation," for more information.

◆ ◆ ◆ CHAPTER 16

Using Calendar Pop-up Reminders

Instant Messaging is integrated with Sun Java™ System 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" on page 181
- "Configuring Instant Messaging Pop-ups" on page 185
- "Configuring Calendar Pop-ups in a Server Pool" on page 187
- "Administering the Calendar Agent" on page 187

Pop-up Reminders Overview

This section contains information about Calendar pop-up reminders in the following topics:

- "Pop-up Reminders Operation" on page 181
- "Pop-up Reminders Architectural Flow" on page 182
- "`iim.conf` Calendar Pop-up Configuration Parameters" on page 182

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 Chapter 1, "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 `iim.conf`	Description and Appropriate Value
JMS Consumers Section	

TABLE 16–1 `iim.conf` Parameters for Configuring Calendar Pop-ups *(Continued)*

Parameter or Section in `iim.conf`	Description and Appropriate Value
jms.consumers	Name of alarm. Set the value to: `cal_reminder`
jms.consumer.cal_reminder.destination	Destination of the alarm. This must be the same as the value of the *caldb.serveralarms.url* configuration parameter in the `ics.conf` file. For example, `enp:///ics/customalarm`
jms.consumer.cal_reminder.provider	The name of the provider. Set to `ens`. This must be the same as the name in the *jms.providers* parameter in the JMS Providers section.
jms.consumer.cal_reminder.type	The type of alarm to set. Set the value to: `topic`
jms.consumer.cal_reminder.param	The alarm parameter. Set the value as follows including the quotes: `"eventtype=calendar.alarm"`
jms.consumer.cal_reminder.factory	A listener that registers itself for the new calendar reminder messages. Set the value to: `com.iplanet.im.server.JMSCalendarMessageListener` Enter the value on a single line.
JMS Providers Section	
jms.providers	The name of the provider. Set value to `ens`. This must be the same as the value listed in the JMS Consumers Section for the *jms.consumer.cal_reminder.provider* parameter.

TABLE 16–1 `iim.conf` Parameters for Configuring Calendar Pop-ups *(Continued)*

Parameter or Section in `iim.conf`	Description and Appropriate Value
jms.provider.ens.broker	Hostname of the ENS and the port number on which the ENS listens for incoming requests. Set to the port specified in the `ics.conf` file parameter *service.ens.port*. The default is 57997. For example: `jms.provider.ens.broker=cal.example.com:57997`
jms.provider.ens.factory	Factory class used for creating the topic connection objects. Set the value to: `com.iplanet.ens.jms.EnsTopicConnFactory`
Instant Messaging General Parameters	
iim_agent.enable	Enables agents for Instant Messaging. By default, this parameter is set to `False`. Set the value as follows including the quotes: `iim_agent.enable="true"`
iim_agent.agent-calendar.enable	Loads a component that enables the Calendar agent. Set the value as follows including the quotes: `iim_agent.agent-calendar.enable="true"`
agent-calendar.jid	The JID of the Calendar agent. Set this value as follows: `agent-calendar.jid=calimbot.`*`server.domain`*
agent-calendar.password	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: `agent-calendar.password=`*`password`*
iim_server.components	Set this value as follows: `iim_server.components=agent-calendar`

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" on page 185
- "To Manually Configure Instant Messaging Server for Calendar Pop-ups" on page 186
- "To Configure Calendar Server for Pop-ups" on page 186
- "To Configure Instant Messenger for Calendar Pop-ups" on page 187

▼ To Configure Instant Messaging Server for Calendar Pop-ups Using the `configure` Utility

1 **Run `configure`.**

See "Completing the Configuration Checklist" on page 29 for more information about the `configure` utility.

2 **On the Calendar Agent configuration screen, select the Enable Calendar Agent checkbox.**

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 Chapter 1, "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" on page 250 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
caldb.serveralarms	Enables calendar alarms to be queued. The default is "`1`" (enabled).
caldb.serveralarms.contenttype	Output format for alarm content. The default is "`text/xml`".
caldb.serveralarms.dispatch	Enables calendar alarms to be dispatched. The default is "`yes`".

Parameter	Description and Default Value
caldb.serveralarms.dispatchtype	The type of server alarm to dispatch. The default is "ens".
caldb.serveralarms.url	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 Sun Java System 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" on page 99 for information on administering the Calendar agent component. Also see Chapter 13, "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" on page 250 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 17

Managing Instant Messaging 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" on page 189
- "Methods for Controlling End User and Administrator Privileges" on page 191
- "Managing Policies Using Access Control Files" on page 193
- "Managing Policies using Sun Java System Access Manager" on page 196

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*.

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 Sun Java™ System Access Manager.

- "Managing Policies Using Access Control Files" on page 193 - 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 Sun Java System Access Manager" on page 196 - 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 Sun Java System 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 Sun Java System Access Manager, you must use the access control file method to manage policies. If you are using Sun Java System 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 Sun Java System 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" on page 250 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.)

     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.

   - `file` (Default, to store user properties in files.)

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 Sun Java System Access Manager can play in Instant Messaging deployments.

TABLE 17–1 Parameters Related to Access Manager in `iim.conf`

Parameter Name	Use	Values
iim.policy.modules	Indicates if Sun Java System Access Manager or the directory is used for policy storage.	`iim_ldap` (default) `identity`
iim.userprops.store	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.	`file` (Default if you chose not to use Access Manager for policy when you ran the `configure` utility.) `ldap` (Default if you chose to use Access Manager for policy when you ran the `configure` utility.)

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
`sysSaveUserSettings.acl`	Defines who can and cannot change their own preferences. Users who do not have this privilege cannot add contacts, create conferences, etc.
`sysTopicsAdd.acl`	Defines who can and cannot create News channels.
`sysRoomsAdd.acl`	Defines who can and cannot create Conference rooms.
`sysSendAlerts.acl`	Defines who can and cannot send alerts. Disabling `sysSendAlerts` also disables polls.
`sysWatch.acl`	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.
`sysAdmin.acl`	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" on page 53 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" on page 195 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" on page 53 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
```

Managing Policies using Sun Java System Access Manager

The Instant Messaging and Presence services in Sun Java System 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 Sun Java System Access Manager, several example policies and roles are created. See the *Sun Java System Access Manager Getting Started Guide* and the *Sun Java System 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*	*sunIMProperties*	*sunIMUserProperties*
	sunIMAllowChatInvite	*sunIMRoster*	*sunIMUserRoster*
	sunIMAllowForumAccess	*sunIMConferenceRoster*	*sunIMUserConferenceRoster*
	sunIMAllowForumManage	*sunIMNewsRoster*	*sunIMUserNewsRoster*
	sunIMAllowForumModerate	*sunIMPrivateSettings*	*sunIMUserPrivateSettings*
	sunIMAllowAlertsAccess		
	sunIMAllowAlertsSend		
	sunIMAllowNewsAccess		
	sunIMAllowNewsManage		
	sunIMAllowFileTransfer		
	sunIMAllowContactListManage		
	sunIMAllowUserSettings		
	sunIMAllowPollingAccess		
	sunIMAllowPollingSend		
sunPresence	*sunPresenceAllowAccess*	*sunPresenceDevices*	*sunPresenceEntityDevices*
	sunPresenceAllowPublish	*sunPresencePrivacy*	*sunPresenceUserPrivacy*
	sunPresenceAllowManage		

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

TABLE 17–4 Access Manager Policy Attributes for Instant Messaging *(Continued)*

Policy Attribute	Admin Console Label	Attribute Description
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
sunPresenceAllowAccess	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
sunPresenceAllowPublish	Ability to Publish Presence	End users can click to select their status (online, offline, busy, etc.) for others to watch
sunPresenceAllowManage	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 theAccess 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 Sun Java System 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	*sunIMUserProperties*	*sunIMProperties*	Contains all the properties for Instant Messenger and corresponds to the `user.properties` 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	*sunIMUserRoster*	*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	*sunIMUserConferenceRoster*	*sunIMConferenceRoster*	Contains conference room subscription information	Merge. Dynamic and user subscriptions are merged, and duplicates are removed.
News Channel Subscriptions	*sunIMUserNewsRoster*	*sunIMNewsRoster*	Contains news channel subscription information	Merge. Dynamic and user subscriptions are merged and duplicates are removed.

TABLE 17–5 Access Manager User and Dynamic Attributes for Instant Messaging *(Continued)*

Admin Console Label	User Attribute	Dynamic Attribute	Attribute Description	Conflict Resolution
Presence Agents	*sunPresenceEntityDevices*	*sunPresenceDevices*	Not used in this release (for future use)	The dynamic information is used.
Privacy	*sunPresenceUserPrivacy*	*sunPresencePrivacy*	Corresponds to the privacy setting in Instant Messenger	Merge. the dynamic value is used if there is a conflict.
Instant Messenger Preferences	*sunIMUserPrivateSettings*	*sunIMPrivateSettings*	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 Sun Java System 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 Sun Java System 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.

TABLE 17–6 Default Policies and Roles for Sun Java System Access Manager *(Continued)*

Policy	Role to Which the Policy Applies	Service to Which the Policy Applies	Policy Description
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
sunIMAllowChat	allow	allow					
sunIMAllowChatInvite	allow	allow					
sunIMAllowForumAccess	allow	allow		allow			
sunIMAllowForumManage	deny	allow		allow			
sunIMAllowForumModerate	deny	allow		allow			
sunIMAllowAlertsAccess	allow	allow				allow	
sunIMAllowAlertsSend	allow	allow				allow	
sunIMAllowNewsAccess	allow	allow	allow				
sunIMAllowNewsManage	deny	allow	allow				
sunIMAllowFileTransfer	allow	allow					
sunIMAllowContact ListManage	allow	allow					

TABLE 17–7 Default Policy Assignments *(Continued)*

	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**
sunIMAllowUserSettings	allow	allow			allow		
sunIMAllowPollingAccess	allow	allow					
sunIMAllowPollingSend	allow	allow					
sunPresenceAllowManage	allow	allow					
sunPresenceAllowAccess	allow	allow					allow
sunPresenceAllowPublish	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 Sun Java System 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 18

Managing Archiving for Instant Messaging

This chapter explains how to configure and manage email, Portal, and custom archiving for Instant Messaging in the following sections:

- "Archiving Overview" on page 211
- "Enabling and Disabling Archiving for Instant Messaging" on page 212
- "Managing the Instant Messaging Email Archive" on page 213
- "Managing the Instant Messaging Portal Archive" on page 219
- "Using a Custom Archive Provider" on page 228

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 Appendix D, "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" on page 213
- "To Enable the Instant Messaging Portal Archive Provider" on page 221
- "To Enable a Custom Archive Provider" on page 229

1. **Open** `iim.conf`**.**

   See "`iim.conf` File Syntax" on page 250 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" on page 214
- "To Disable the Portal Archive Provider" on page 222
- "To Disable a Custom Archive Provider" on page 229

1. **Open** `iim.conf`**.**

   See "`iim.conf` File Syntax" on page 250 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" on page 213
- "Configuring Email Archive Settings" on page 214
- "Email Header Format" on page 216

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 *Sun Java Communications Suite 5 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

Before You Begin Ensure that you have enabled archiving for Instant Messaging as described in "To Enable Instant Messaging Archiving" on page 212.

1 **Open `iim.conf`.**

See "`iim.conf` File Syntax" on page 250 for more information.

2 **Add the following line to `iim.conf` if it does not already exist.**

```
iim_server.msg_archive.provider = com.iplanet.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.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 the Instant Messaging Email Archive Provider

1 **Open** `iim.conf`**.**

See "`iim.conf` File Syntax" on page 250 for more information.

2 **Delete the** *com.iplanet.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
iim_arch.admin.email	Empty String	Comma-separated list of administrator email addresses.

TABLE 18–1 Email Archive Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim_arch.alert.admin.email	None	Comma-separated list of administrator email addresses to which all archived alert messages will be sent. This parameter overrides *iim_arch.admin.email* for alert messages.
iim_arch.chat.admin.email	None	Comma-separated list of administrator email addresses to which all archived chat messages will be sent. This parameter overrides *iim_arch.admin.email* for chat messages.
iim_arch.conference.admin.email	None	Comma-separated list of administrator email addresses to which all archived conference messages will be sent. This parameter overrides *iim_arch.admin.email* for conference messages.
iim_arch.poll.admin.email	None	Comma-separated list of administrator email addresses to which all archived poll messages will be sent. This parameter overrides *iim_arch.admin.email* for poll messages.
iim_arch.news.admin.email	None	Comma-separated list of administrator email addresses to which all archived news messages will be sent. This parameter overrides *iim_arch.admin.email* for news messages.
iim_arch.email.archiveheader.name	None	Name of the extended RFC 822 header.
iim_arch.email.archiveheader.value	all	Value corresponding to the header name for *iim_arch.email.archiveheader.name*.

▼ 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" on page 250 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" on page 216
- "RFC 822 Email Archive Header Fields for Private Conferences" on page 216
- "RFC 822 Email Archive Header Fields for Public Conferences" on page 217
- "RFC 822 Email Archive Header Fields for Poll Questions with Replies" on page 217
- "RFC 822 Email Archive Header Fields for Poll Replies Only" on page 217
- "RFC 822 Email Archive Header Fields for Alerts" on page 218
- "RFC 822 Email Archive Header Fields for News Channel Posts" on page 218

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.
`Cc:`	Chat session initiator.
`Subject:`	First useful message over 50 characters in length.
`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 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 recipients 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 based on the news channel ID.

Managing the Instant Messaging Portal Archive

The following topics describe using the Instant Messaging Portal Archive:

- "Instant Messaging Portal Archive Overview" on page 219
- "Enabling and Disabling the Portal Archive Provider" on page 221
- "Configuring the Instant Messaging Portal Archive Provider" on page 222
- "Managing Archived Data in the Portal Server Search Database" on page 225
- "Changing the Display of Archived Data" on page 227
- "Sample Deployment Scenario for Archive Provider" on page 227

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 illustrates Instant Messaging Portal Archive components.

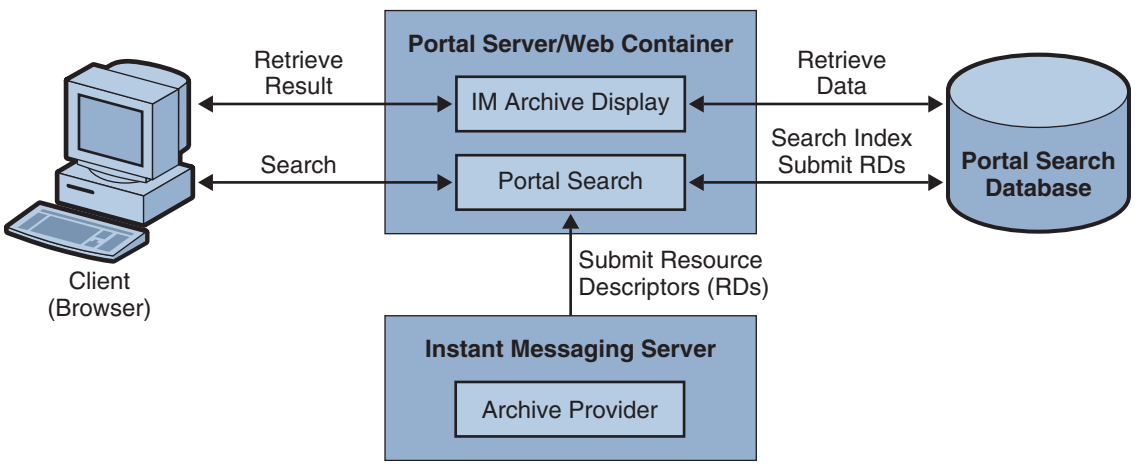

FIGURE 18–1 Instant Messaging Portal Archive Components

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

Before You Begin Ensure that you have enabled archiving for Instant Messaging as described in "To Enable Instant Messaging Archiving" on page 212.

1. **Open `iim.conf`.**

   See "`iim.conf` File Syntax" on page 250 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" on page 250 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
Conference	*RoomName-StartTime*
Chat	Where: *RoomName* - Name of the public or private conference room *StartTime* - Timestamp of the creation of RD
Alert	`Alert-`*messageID* Where: *messageID* - 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	`Poll-`*pollID*
News	`TopicName-`*messageID*

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 Appendix A, "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 Appendix A, "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:setRDMType 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:setRDMType 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","<span class='user'> {0} </span>
<span class='headervalue'> has joined the room.</span>");
....
```

3 **Replace the `headervalue` with the desired text.**

For example:

```
....
ht.put("has_joined_the_room","<span class='user'> {0} </span>
<span class='headervalue'> has entered the room.</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:

EXAMPLE 18–1 Archiving Related Instant Messaging Data Collectively *(Continued)*

```
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" on page 53 for information on locating *im-cfg-base*.

2. **Open `iim.conf`.**

   See "`iim.conf` File Syntax" on page 250 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

- **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

Before You Begin Ensure that you have enabled archiving for Instant Messaging as described in "To Enable Instant Messaging Archiving" on page 212.

1. **Open `iim.conf`.**

   See "`iim.conf` File Syntax" on page 250 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" on page 250 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 19

Troubleshooting and Monitoring Instant Messaging

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" on page 231
- "Problems and Solutions" on page 232
- "Troubleshooting Instant Messaging and LDAP" on page 239
- "Troubleshooting Connectivity Issues in a Multi-Node Deployment (Server Pool)" on page 243
- "Monitoring Instant Messaging" on page 243
- "Managing the Watchdog Process" on page 243

For details and more information on managing server, multiplexor, watchdog, Calendar agent, and client logging, and for default log file locations, see Chapter 13, "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" on page 145 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" on page 233
- "Unable to Log into Instant Messenger through the XMPP/HTTP Gateway" on page 233
- "Messages Not Archived With Sun Java System Portal Server 7 2006Q1 or Later" on page 234
- "Instant Messenger Resource Customizations Lost After `patchrm` and `patchadd`" on page 234
- "Cannot Forward Mail to Offline Users" on page 234
- "Calendar Pop-up Reminders Do Not Work" on page 235
- "Single Sign-on Does Not Work" on page 236
- "Instant Messenger Does Not Load or Start" on page 236
- "Connection Refused or Timed Out" on page 236
- "Authentication Errors" on page 237
- "Instant Messenger Channel Display Error" on page 237
- "Instant Messaging Content is not Archived" on page 238
- "Server-to-Server Communication Fails to Start" on page 238
- "Catastrophic Installation Failure Leaves Server in an Inconsistent State" on page 238
- "Instant Messaging Services Do Not Appear in the Access Manager Console (amconsole)" on page 239

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" on page 250 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 Chapter 16, "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–11 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 `iim.conf` Parameters" on page 145.

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" on page 105 for instructions.

2. **Manually edit the product registry information.**
   For Solaris 9, issue 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:

- Solaris: `/var/sadm/install/productregistry`
- Linux: `/var/tmp/productregistry`

b. Remove the following packages or RPMs if they are still present:

- `SUNWiim`
- `SUNWiimc`
- `SUNWiimd`
- `SUNWiimid`
- `SUNWiimin`
- `SUNWiimjd`
- `SUNWiimm`
- `SUNWiimc-l10n`
- `SUNWiimd-l10n`
- `SUNWiimid-l10n`
- `SUNWiimin-l10n`

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" on page 250 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 byInstant 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" on page 250 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" on page 174 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" on page 253 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" on page 250 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" on page 250 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.user.uidattr* 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" on page 250 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.user.uidattr*.**

   The default value is `uid`.

   For example, to use the *loginname* attribute, set the *iim_ldap.user.uidattr* attribute as follows:

   ```
   iim_ldap.user.uidattr=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" on page 78 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 Appendix C, "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" on page 250 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 Chapter 13, "Managing Logging for Instant Messaging."

PART III

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.

◆ ◆ ◆ APPENDIX A

Instant Messaging 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" on page 249
- "`iim.conf` File Syntax" on page 250
- "General Configuration Parameters" on page 250
- "LDAP and User Registration Configuration Parameters" on page 253
- "Logging Configuration Parameters" on page 255
- "Instant Messaging Server Configuration Parameters" on page 256
- "Multiple Server Configuration Parameters" on page 261
- "Multiplexor Configuration Parameters" on page 263
- "Redirect Server Parameters" on page 264
- "Archive Parameters" on page 266
- "Watchdog Parameters" on page 271
- "Monitoring Parameters" on page 271
- "Agent Parameters" on page 272

`iim.conf` File Location

Instant Messaging stores configuration settings in the `iim.conf` file within the Configuration Directory (*im-cfg-base*).

- On Solaris:

  `/etc/opt/SUNWiim/default/config/iim.conf`

- On 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" on page 44 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 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
iim.comm.modules	`iim_server,iim_mux`	The communication modules used. The possible values are `iim_server` and `iim_mux`. The default value is `iim_server, iim_mux`, which means both the server and multiplexor are used. The `iim_mux` value is useful for multiplexor.
iim.smtpserver	`localhost`	SMTP server to send mail to end users who have set the option for forwarding their messages as emails or to pagers.
iim.instancedir	`/opt`	The installation directory root.
iim.instancevardir	Solaris: `/var/opt/SUNWiim/default` Linux: `/var/opt/sun/im/default`	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 `/default` directory may vary if you created multiple instances of Instant Messaging.
iim.user	`inetuser` for LDAP deployments. `root` for portal deployment.	The end-user name with which the server processes run.
iim.group	`inetgroup` for LDAP deployments. `root` for portal deployment.	The group using which the server processes run.
iim.jvm.maxmemorysize	`256`	The maximum number heap size in MB the JVM running the server is allowed to use. Used to construct the `-mx` argument of the Java command.
iim.mail.charset	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 out for offline alerts. For example: `iim.mail.charset=iso-2022-jp`
iim.jvm.command	`/usr/j2se/bin/java`	The location of the Java Runtime Executable (JRE).
iim.identity.basedir	`/opt`	The default installation directory, also referred to as the base directory, for Sun Java™ System Access Manager.
iim.identity.jre	`/usr/java_1.3.1_04`	The location of the JRE used by the Access Manager to run all its processes.

TABLE A–1 General Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim.portal.deployuri	`/portal`	The URI using which the Portal Server `war` files are deployed in the Access Manager.
iim.portal.host	`imhostname`	The host name of the server on which the Portal Server is running. Specify the port number if a non default port number is used.
iim.portal.protocol	`http`	The protocol used to access the Portal Server.
iim.policy.cache.validity	10	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 `LocalUser` object.
iim.policy.modules	`iim_ldap`	By default, LDAP is used for policy storage. Change the value to `identity` to indicate that Sun Java System Access Manager should be used for policy storage.
iim.policy.resynctime	`720`	Defines the cache validity interval (in minutes) for all end-user information. The Instant Messaging server clears all cached end-user information on a regular basis in order to eliminate old end-user information. This parameter specifies the frequency at which the cached end-user information is cleared.
iim.userprops.store	`file`	By default, user properties are stored in a user properties file if you chose not to use Access Manager for policy when you ran the `configure` utility. If you chose to use Access Manager for policy, the default is `ldap`. Change the value to change the location where user properties are stored. If you change this from `file` to `ldap`, you need to run `imadmin assign_services` to add 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.

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
iim_ldap.host	`localhost:389`	LDAP server name and port used by Instant Messaging server for end-user authentication.
iim_ldap.searchbase	`o=internet`	The string used as base to search for the end users and groups on the LDAP server.
iim_ldap.usergroupbinddn	None (the server performs anonymous searches)	Specifies the DN to use to bind to the LDAP server for searches.
iim_ldap.usergroupbindcred	None (the server performs anonymous searches)	Specifies the password to use with the `iim_ldap.usergroupbinddn` DN for LDAP searches.
iim_ldap.loginfilter	`(&(|(objectclass=inetorgperson)(objectclass=webtopuser))(uid={0}))`	Search filter used during end-user login. The entire filter is entered as one line.
iim_ldap.usergroupbyidsearchfilter	`(|(&(objectclass=groupofuniquenames)(uid={0}))(&(|(objectclass=inetorgperson)(objectclass=webtopuser))(uid={0})))`	The search filter used to search for end users and groups in the directory, under the base specified by ID. The entire filter is entered as one line.
iim_ldap.usergroupbynamesearchfilter	`(|(&(objectclass=groupofuniquenames)(cn={0}))` `(&(|(objectclass=inetorgperson)(objectclass=webtopuser))(cn={0})))`	The search filter used to search for end users and groups in the directory, under the base specified by name.
iim_ldap.allowwildcardinuid	`False`	Determines if wildcards should be enabled for UIDs while performing a search. As most directory installations have UIDs indexed for exact searches only, the default value is `False`. Setting this value to `True` can impact performance unless UIDs are indexed for substring search.
iim_ldap.userclass	`inetOrgPerson,webtopuser`	The LDAP class that indicates that an entry belongs to an end user.
iim_ldap.groupclass	`groupOfUniqueNames`	The LDAP class that indicates that an entry belongs to a group.

TABLE A–2 LDAP, User Registration, and Source Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim_ldap.groupbrowsefilter	`(objectclass=groupofuniquenames)`	The search filter used to browse all groups in the directory, under the specified search base.
iim_ldap.searchlimit	`40`	Maximum number of entries to be returned by a search. A value of `-1` means search is disabled on this server and a value of `0` indicates unlimited search.
iim_ldap.userdisplay	`cn`	LDAP attribute to use for display name of end users.
iim_ldap.groupdisplay	`cn`	LDAP attribute to use for display name of groups.
im_ldap.useruidattr	`uid`	LDAP attribute used as end users' UID.
im_ldap.groupmemberattr	`uniquemember`	LDAP attribute that gives the list of members of a group.
iim_ldap.usermailattr	`mail`	LDAP attribute that should contain end users' provisioned email addresses. Used when the email message is sent to an offline end user.
iim_ldap.userattributes	None	LDAP attribute that contains the list of custom attributes from the LDAP user entry.
iim_ldap.groupattributes	None	LDAP attribute that contains the list of custom attributes from the LDAP group entry.
iim_ldap.groupmemberurlattr	None	The membership attribute of a dynamic group, which contains the LDAP filter or the LDAP URL.
iim_ldap.useidentityadmin	The default value is `true`, if you chose to leverage an Access Manager deployment for policy when you ran the `configure` utility. Otherwise, the default value is `false`.	If the value is `true` then the Access Manager Administrator credentials will be used to bind to the Directory Server.
iim.register.enable	None	If `TRUE`, the server allows new Instant Messaging end users to register themselves (add themselves to the directory) using Instant Messenger.
iim_ldap.register.basedn	None	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"`

TABLE A–2 LDAP, User Registration, and Source Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim_ldap.register.domain	None	The domain to which new users will be added. For example, `directory.siroe.com`.

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
iim.log.iim_server.severity	`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`.
iim.log.iim_server.url	*im-runtime-base*`/log/xmppd.log`	Location of the server log file. This file needs to be periodically trimmed to prevent disk space from filling up.
iim.log.iim_mux.severity	`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`.
iim.log.iim_mux.url	*im-runtime-base*`/log/mux.log`	Location of the multiplexor log file. This file needs to be periodically trimmed to prevent disk space from filling up.
iim.log.iim_mux.maxlogfiles	10	The maximum number of log files to store for the multiplexor. Once this number is exceeded, the oldest multiplexor log file is deleted.

TABLE A–3 Logging Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim.log.iim_mux.maxlogfilesize	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.
iim.log.iim_server.maxlogsize		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.
iim.log.iim_wd.severity	`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`.
iim.log.agent-calendar.severity	`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`.
iim.log4j.config	*im-cfg-base*	Specifies the location and name of the log4j configuration file. If no value exists for this parameter, the logger will look for `log4j.conf` in *im-cfg-base*. If the logger does not find `log4j.conf` in *im-cfg-base*, 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
iim_server.autosubscribe	FALSE	Indicates whether subscriptions are automatically authorized by the server. The possible values are TRUE and FALSE. If TRUE, subscribe requests are automatically followed by a subscribed response generated by the server. The server then sends the modified roster to the subscriber and the user the subscriber added as a contact. The user and the contact must be in the same domain to use this feature.
iim_server.domainname	*host's domain name*	The logical Instant Messaging server domain name you want this server to support. This is the name that is used by other servers in the network to identify this server. It is also the name used by this server to identify its end users to other servers. This is not necessarily the Fully Qualified Domain Name of the system running the Instant Messaging server. For example, if the system `iim.xyz.com` is the only Instant Messaging server for a company `xyz.com`, then the domain name is likely to be `xyz.com`.
iim_server.port	5269	IP address and port for the server to bind to, and to listen for connections from other servers. IP address setting is useful for multi homed machines when you want to use only one particular IP address. If no IP address is listed, this indicates a value of `INADDR_ANY` on `localhost`.
iim_server.useport	TRUE	Indicates whether the server should listen on the server-to-server communication port. The possible values are TRUE and FALSE. If TRUE, the server listens on the port defined by *iim_server.port* or on port 5269, if that is not explicitly defined.

TABLE A–4 General Instant Messaging server Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim_server.clienttimeout	`15`	Specifies the time, in minutes, before the server discards client connections that are no longer active. For example, when a machine is turned off. The minimum accepted value is 5.
iim_server.usesso	The default value is `1`, if you chose to leverage an Access Manager deployment for SSO when you ran the `configure` utility. Otherwise, the default value is `0`.	This parameter tells the server whether or not to depend on the SSO provider during authentication. An SSO provider is a module the server uses to validate a session ID with a SSO service. TheAccess Manager Session API provides the Instant Messaging server with the ability to validate session IDs sent by the client. The value for this parameter can either be `0`, `1`, or `-1`. `0` - do not use the SSO provider. `1` - use the SSO provider first and default to LDAP if the SSO validation fails. `-1`- use SSO provider only without attempting LDAP authentication even when the SSO validation fails. The *iim_server.usesso* parameter is used in conjunction with the *iim_server.ssoprovider* parameter.
iim_server.ssoprovider	None	Specifies the class implementing the `com.sun.im.provider.SSOProvider` interface. If `iim_server.usesso` is not equal to `0` and this option is not set, the server uses the default Access Manager–based SSO Provider.
iim.policy.modules	The default value is `identity`, if you chose to leverage an Access Manager deployment for policy when you ran the `configure` utility. Otherwise, the default value is `iim_ldap`.	If the value is `identity`, indicates that Sun Java System Access Manager is used for policy storage. If the value is `iim_ldap`, directory is used.

TABLE A–4 General Instant Messaging server Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim.userprops.store	`file`	If the value is `file`, indicates that the user properties are stored in a user properties file. If the value is `ldap`, directory is used.
iim_server.msg_archive	`false`	This parameter specifies whether the archive provider should be enabled or disabled. Set this value to `false` to disable all archiving. Set the value to `true` to enable all archiving, including Portal, email, and any custom archive provider you want to use.
iim_server.msg_archive.provider	None	This parameter contains the list of archive providers. This parameter allows multiple values and each value is separated by a comma (`,`). If you are using the Portal Server Search based archive, the value should be `com.iplanet.im.server.IMPSArchive`. If you are using email archiving, the value should be `com.iplanet.im.server.EmailIMArchive`.
iim_server.conversion	`false`	This parameter specifies whether message conversion should be enabled. It specifies whether the configured list of Message Conversion Providers should be used for message conversion.
iim_server.conversion.provider	None	This parameter contains the list of Message Conversion Providers to be used for message conversion. This parameter allows multiple values with each value is separated by a comma (`,`).

TABLE A–4 General Instant Messaging server Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim_server.servertimeout	-1	The server can be configured to automatically close the connection opened by a remote server, if the remote server is inactive. This is performed by periodically measuring the time the last request was made by the remote server to the server. The connection to the remote server is terminated, if the time of the last request made by the remote server exceeds the value of the *iim_server.servertimeout* parameter. The parameter value is in minutes.
iim_server.enable	`true`	This value defines whether or not the Instant Messaging server is enabled. This parameter is set false to enable the Instant Messaging multiplexor.
iim_server.conversion.external.command	None	This parameter contains the external command used for message conversion.
iim_server.stat_frequency	`1`	This parameter contains the frequency at which the server logs the summary of activities to the log file. The server logs the summary of activities to the log file only if the server minimum log severity is set to `INFO` or lower. The value is in minutes.
iim_server.certnickname	`Server-Cert`	This value should contain the name of the certificate you entered while installing the certificate. The certificate name is case-sensitive.
iim_server.sslkeystore	None	Contains the relative path and filename for the server's Java keystore (JKS). For example: */im-cfg-base/*`server-keystore.jks`

TABLE A–4 General Instant Messaging server Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim_server.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:`*password* Where *password* is the password protecting the key database.
iim_server.requiressl	`false`	If true, the server will terminate any connection that does not request a TLS connection after the initial stream session is set up. This includes connections from clients, other servers, and server components, such as the XMPP/HTTP Gateway and agents, except the multiplexor.
iim_server.trust_all_cert	`false`	If this value is true than the server will trust all certificates and will also add the certificate information into the log files.

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
iim_server.serverid	None	String used by this server to identify itself to all other servers.
iim_server.password	None	Password used by this server to authenticate itself to all other servers.

iim_server.coservers	None	Comma separated list containing symbolic names of the servers that can connect to this server. Any meaningful names are allowed, but they must match what you use for the `*.serverid`, `*.password`, and `*.host` parameters. Examples:
		`iim_server.coservers=cose rver1,coserver2`
		or
		`iim_server.coservers=abc,xyz,ntc`

iim_server.coserver1.serverid	None	String that identifies the cooperating server represented by the name, *coserver1* to authenticate to this server. For example, if you used `abc` in the *iim_server.coservers* list, then the corresponding name for its *serverid* would be `iim_server.abc.serverid`.
iim_server.coserver1.password	None	Password used by cooperating server represented by the name, *coserver1* to authenticate to this server. For example, if you used `abc` in the *iim_server.coservers* list, then the corresponding name for its password would be `iim_server.abc.password`.
iim_server.coserver1.host	None	IP address and the port to connect to, for end users on this server to communicate to end users on the server represented by the name `coserver1`. For example, if you used `abc` in the `iim_server.coservers` list, then the corresponding name for its host would be `iim_server.abc.host`. The format is *name*:*port* or *IPaddress*:*port*.
iim_server.coserver1.requiressl	`False`	Indicates if this server should require TLS when communicating with the server identified by *coserver1*. The possible values are `TRUE` and `FALSE`.

Multiplexor Configuration Parameters

Table A–6 lists and describes the multiplexor configuration parameters.

TABLE A–6 Multiplexor Configuration Parameters

Parameter	Default Value	Description
iim_mux.listenport	*multiplexorname or IP address:*`5222`	IP address or FQDN and listening port on which the multiplexor listens for incoming requests from Instant Messenger. The value format is *IPaddress*:*port* or *multiplexorname*:*port*. If no IP address or domain name is listed, `INADDR_ANY` on `localhost` is assumed. If you change this value, also change the `im.html` and `im.jnlp` files so that they match the port value.
iim_mux.serverport	`45222`	The Instant Messaging server and port the multiplexor talks to. The value format is *servername*:*port* or *IPaddress*:*port*.
iim_mux.numinstances	`1`	Number of instances of the multiplexor. This parameter is valid only for Solaris platforms.
iim_mux.maxthreads	`5`	Maximum number of threads per instance of the multiplexor.
iim_mux.maxsessions	`2000`	Maximum number of concurrent connections per multiplexor process.
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	`/etc/opt/SUNWiim /default/config`	This directory contains the key and certificate databases. In addition, it also usually contains the security module database. The name of the `/default` directory may vary if you created multiple instances of Instant Messaging.
iim_mux.keydbprefix	None	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`.

TABLE A–6 Multiplexor Configuration Parameters *(Continued)*

Parameter	Default Value	Description
iim_mux.certdbprefix	None	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.certnickname	*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	`/etc/opt/SUNWiim /default/config /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:`*password* Where *password* is the password protecting the key database. The name of the `/default` directory may vary if you created multiple instances of Instant Messaging.
iim_mux.stat_frequency	`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.
iim_mux.enable	`true`	If the value is `true` then the multiplexor will run for this instance. If the value is `false` then the multiplexor will not run for this instance.

Redirect Server Parameters

Table A–7 lists the parameters you use to administer the Instant Messaging redirect server.

TABLE A–7 Redirect Server Parameters

Parameter	Default Value	Description
iim_server.redirect.provider	None	Comma-separated list of redirect provider names or classes that implement the `com.sun.im.provider.Redirector` interface. Any value for this parameter defines the server instance as a redirect server. Supported values include `db`, `roundrobin`, `regex`, and class names that implement the `com.sun.im.provider.Redirector` interface.
iim_server.redirect.to	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 *iim_server.redirect.to.nodename.host*.
iim_server.redirect.to.nodename.host	None	Where *nodename* is the name of the node as it exists in *iim_server.redirect.to*. This attribute is required for *nodename* to be used by the redirect server.
iim_server.redirect.to.nodename.usessl	False	If true, then *nodename* is configured to use legacy SSL. See "Overview of Using TLS and Legacy SSL in Instant Messaging" on page 123 for more information.
iim_server.redirect.db.users	*im-db-base*`/redirect.db`	Name and location of the redirect database.
iim_server.redirect.db.partitions	*im-cfg-base*`/redirect.partitions`	Name and location of the redirect partitions file.
iim_server.redirect.db.partitionsize	5000	The maximum number of users in a partition.
iim_server.redirect.roundrobin.partitions	*im-cfg-base*`/redirect.partitions`	Name and location of the redirect partitions file.

TABLE A–7 Redirect Server Parameters *(Continued)*

Parameter	Default Value	Description
iim_server.redirect.pollfrequency		The interval between connections made by the redirect server to the hosts defined in the `redirect.hosts` file. The redirect server polls these hosts to determine if they are online and able to accept client connections.

Archive Parameters

Table A–8 lists the parameters you use to manage Instant Messaging archiving.

TABLE A–8 Archive Parameters

Parameter	Default Value	Description
iim_arch.title.attr	`Title`	This parameter contains the name of the field equivalent to the `Title` field in the default schema of the Portal Server Search.
iim_arch.keyword.attr	`Keyword`	This parameter contains the name of the field equivalent to the `Keyword` field in the default schema of the Portal Server Search.
iim_arch.readacl.attr	`ReadACL`	This parameter contains the name of the field equivalent to the `ReadACL` field in the default schema of the Portal ServerSearch.
iim_arch.description.attr	`Description`	This parameter contains the name of the field equivalent to the `Description` field in the default schema of the Portal Server Search.
iim_arch.fulltext.attr	`Full-Text`	This parameter contains the name of the field equivalent to the `Full-Text` field in the default schema of the Portal Server Search.
iim_arch.category.attr	`Category`	This parameter contains the name of the field equivalent to the `Category` field in the default schema of the Portal Server Search.

TABLE A–8 Archive Parameters *(Continued)*

Parameter	Default Value	Description
iim_arch.readacl.admin	None	This parameter contains the administrator's DN. Multiple values should be separated by ";"
iim_arch.readacl.adminonly	`false`	This parameter will contain `true` or `false`. `true` - Only the administrator's DN specified by the parameter *iim_arch.readacl.admin* will be added to the `ReadACL` field overwriting the default behavior of the `ReadACL` field. `false` - The administrator's DN specified by the parameter `iim_arch.readacl.admin` will be added to the `ReadACL` field in addition to the default behavior.

iim_arch.categories	`all`	This parameter contains a list of message types that can be archived.
		The value can be:
		`poll`
		`alert`
		`chat`
		`conference`
		`news`
		Multiple values can be specified separated by commas (`,`).

iim_arch.categoryname	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.
iim_arch.alert.categoryname	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.

TABLE A–8 Archive Parameters *(Continued)*

Parameter	Default Value	Description
iim_arch.poll.categoryname	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.
iim_arch.conference.categoryname	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.
iim_arch.chat.categoryname	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.
iim_arch.news.categoryname	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.
iim_arch.conference.quiettime	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.
iim_arch.poll.maxwaittime	15	This parameter contains the (maximum) time for which poll data is buffered in the server. The value is in minutes.

TABLE A–8 Archive Parameters *(Continued)*

Parameter	Default Value	Description
iim_arch.ignoreexplicitdeny	`true`	This parameter will contain `true` or `false`. `true` - For Poll and Conference category 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 `xmppd.log` file. `false` - For Poll and Conference category 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 then the default access is either `READ` or `WRITE` or `MANAGE`. Some end users can also be granted `NONE` access.
iim_arch.portal.search	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` If this parameter is not present then the Archive Provider determines the value of the Portal Server Search URL based on the `AMConfig.properties` file present on the system.
iim_arch.portal.admindn	None	The value of this parameter should be the DN of the admin user. For example: `uid=amadmin,ou=People,o=internet` This parameter is required when the Document level Security in the Portal Server Server is on.
iim_arch.portal.adminpassword	None	The value of this parameter should be the password of the administrative user as specified by the *iim_arch.portal.admindn* parameter. This parameter is required when the Document level Security in the Portal Search Server is on.

TABLE A–8 Archive Parameters *(Continued)*

Parameter	Default Value	Description
iim_arch.portal.search.database	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.
iim_arch.admin.email	Empty String	Comma-separated list of administrator email addresses.
iim_arch.alert.admin.email	None	Comma-separated list of administrator email addresses to which all archived alert messages will be sent. This parameter overrides *iim_arch.admin.email* for alert messages.
iim_arch.chat.admin.email	None	Comma-separated list of administrator email addresses to which all archived chat messages will be sent. This parameter overrides *iim_arch.admin.email* for chat messages.
iim_arch.conference.admin.email	None	Comma-separated list of administrator email addresses to which all archived conference messages will be sent. This parameter overrides *iim_arch.admin.email* for conference messages.
iim_arch.poll.admin.email	None	Comma-separated list of administrator email addresses to which all archived poll messages will be sent. This parameter overrides *iim_arch.admin.email* for poll messages.
iim_arch.news.admin.email	None	Comma-separated list of administrator email addresses to which all archived news messages will be sent. This parameter overrides *iim_arch.admin.email* for news messages.
iim_arch.email.archiveheader.name	None	Name of the extended RFC 822 header.
iim_arch.email.archiveheader.value	`all`	Value corresponding to the header name for *iim_arch.email.archiveheader.name*.

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" on page 243

Table A–9 lists and describes the watchdog configuration parameters.

TABLE A–9 Watchdog Configuration Parameters

Parameter	Default Value	Description
iim_wd.enable	`true`	Enables the watchdog feature. To reset this parameter or disable the watchdog, set this to `false`. To avoid conflicts, you should disable the watchdog if you are monitoring the Instant Messaging server using the operating system administration console.
iim_wd.period	`300` (seconds)	The watchdog periodically polls the server to check whether it is running. This parameter sets the interval between two status polls.
iim_wd.maxRetries	`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–10 configures how the server interacts with the Sun Java Enterprise System Monitoring Framework.

TABLE A–10 Monitoring Parameters

Parameter	Default Value	Description
iim_server.monitor.enable	`false`	Used by the Sun Java Enterprise System Monitoring Framework. If `true`, configures the server to make its activities available to `mfwk`. Otherwise, the server does not make its activities available.
iim_server.monitor.htmlport	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–11 lists and describes agent configuration parameters.

TABLE A–11 Agent Configuration Parameters

Parameter	Default Value	Description
jms.consumers	None	Used with the Calendar agent. Contains the name of the alarm. The value for this parameter must be set to: `cal_reminder`
jms.consumer.cal_reminder.destination	None	Used with the Calendar agent. Destination of the alarm. This must be the same as the value of the *caldb.serveralarms.url* configuration parameter in the `ics.conf` file. For example, `enp:///ics/customalarm`
jms.consumer.cal_reminder.provider	None	Used with the Calendar agent. The name of the provider. Typically, this is set to `ens`. The value for this parameter must be the same as the name in the *jms.providers* parameter.
jms.consumer.cal_reminder.type	None	Used with the Calendar agent. The type of alarm to set. The value for this parameter must be set to: `topic`
jms.consumer.cal_reminder.param	None	Used with the Calendar agent. The alarm parameter. The value for this parameter must be set as follows including the quotes: `"eventtype=calendar.alarm"`
jms.consumer.cal_reminder.factory	None	Used with the Calendar agent. A listener that registers itself for the new calendar reminder messages. The value for this parameter must be set to: `com.iplanet.im.server.JMSCalendarMessageListener`
jms.providers	None	Used with the Calendar agent. The name of the provider. Typically, you set the value of this parameter to `ens`. This must be the same as the value for the *jms.consumer.cal_reminder.provider* parameter.

TABLE A–11 Agent Configuration Parameters *(Continued)*

Parameter	Default Value	Description
jms.provider.ens.broker	None	Used with the Calendar agent. Hostname of the ENS and the port number on which the ENS listens for incoming requests. Set to the port specified in the `ics.conf` file *service.ens.port* parameter. The default is 57997. For example: `jms.provider.ens.broker=cal.exa mple.com:57997`
jms.provider.ens.factory	None	Used with the Calendar agent. Factory class used for creating the topic connection objects. The value for this parameter must be set as follows. Enter the value on a single line: `com.iplanet.ens.jms.EnsTopicConnFa ctory`
iim_agent.enable	`False`	If `TRUE`, `iim.conf`, enables Instant Messaging agents. Set the value to `FALSE`, or remove the parameter from `iim.conf` to disable all agents.
iim_agent.agent-calendar.enable	None	Used with the Calendar agent. If `TRUE` or absent from `iim.conf`, loads a component that enables the Calendar agent specifically.
agent-calendar.jid	None	The JID of the Calendar agent.
agent-calendar.password	None	Defines the password with which the Calendar agent connects to the Instant Messaging server.
iim_server.components	None	Describes the Calendar agent as a component of the Instant Messaging server. The value of this parameter must be set to: `agent-calendar`

◆ ◆ ◆ APPENDIX B

Instant Messaging XMPP/HTTP Gateway Configuration Parameters in `httpbind.conf`

Instant Messaging stores configuration settings for the XMPP/HTTP Gateway in the `httpbind.conf` file. This appendix describes the configuration parameters and the file in the following sections:

- "`httpbind.conf` File Location" on page 275
- "`httpbind.conf` File Syntax" on page 276
- "Instant Messaging XMPP/HTTP Gateway Configuration Parameters" on page 276
- "Gateway Domain ID Key Parameters for *httpbind.config*" on page 279

Any time you modify the `httpbind.conf` file, you need to restart the XMPP/HTTP Gateway using the tools provided by your web container or application server.

`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:

- On Solaris:

  `/etc/opt/SUNWiim/default/config/httpbind.conf`

- On Linux:

  `/etc/opt/sun/im/default/config/httpbind.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" on page 44 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 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 guide.

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
httpbind.pool.nodeId	N/A	If *httpbind.pool.support* is set to `true`, 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.
httpbind.pool.support	`false`	This parameter defines whether or not the gateway is in a server pool deployment. If no *httpbind.pool.nodeId* is specified, the value for this parameter is set to `false`. The value for this parameter can be: ■ `true` – the gateway is part of a server pool deployment. In addition, `enable`, `on`, `yes`, and `1` are also valid values. If you set this parameter to true, you must provide a value for *httpbind.pool.nodeId*. ■ `false` – (default) the gateway is not part of a server pool deployment. Leaving the value blank (empty string) is also a valid value.
httpbind.config	N/A	Contains a comma-separated list of ID keys, or *gwdomain-id*, 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.
httpbind.content_type	`text/xml; charset=utf-8`	The default value for the `content-type` HTTP header the gateway uses when sending a response back to the client.

TABLE B–1 XMPP/HTTP Gateway Configuration Parameters in `httpbind.conf` *(Continued)*

Parameter	Default Value	Description
httpbind.hold	N/A	Specifies the maximum permissible value for the *hold* attribute in the client request as defined in JEP 124. 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.
httpbind.inactivity	`180`	The maximum time in seconds of client inactivity after which the gateway will terminate the connection to the client.
httpbind.log4j.config	N/A	The location of the `log4j` 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`" (`log4j.logger.httpbind`).
httpbind.polling	`1` (second)	The minimum time, in seconds, a client must wait before sending another request.
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 *httpbind.hold* for more information.
httpbind.round_trip_delay	`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.

TABLE B–1 XMPP/HTTP Gateway Configuration Parameters in `httpbind.conf` *(Continued)*

Parameter	Default Value	Description
httpbind.wait_time	`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
gwdomain-id.domains	Comma-separated list of domains for this ID.
gwdomain-id.hosts	Space-separated list of hosts for this ID. Each of these hosts must be able to service the domains listed in *gwdomain-id.domains*. 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.
gwdomain-id.componentjid	The component JID to use to connect to the host.
gwdomain-id.password	The password to use to connect to the host.

◆ ◆ ◆ APPENDIX C

Instant Messaging `imadmin` Tool Reference

This chapter explains the `imadmin` command used to administer Instant Messaging in the following sections:

- "`imadmin` Overview" on page 281
- "`imadmin` Requirements" on page 281
- "`imadmin` Location" on page 282
- "`imadmin` Commands" on page 282
- "`imadmin` Syntax" on page 283
- "`imadmin` Options" on page 283
- "`imadmin` Actions" on page 283
- "`imadmin` Components" on page 284

`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
imadmin assign_services	If iim.policy.modules is set to identity, 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 iim.policy.modules is set to iim_ldap, and iim.userprops.store is set to ldap, this command adds objectclasses (sunIMUser, and sunPresenceUser) to user entries in the directory. Instant Messaging requires these objectclasses in order to store properties in LDAP.
imadmin status (Previously imadmin check)	Checks to see if the components (server, multiplexor, agent-calendar, and watchdog) are up and running and displays the results. If you don't specify a component, the imadmin utility returns information about all components.
imadmin start	Starts the enabled component(s).
imadmin stop	Stops the enabled component(s).
imadmin refresh	Refreshes the enabled component(s).
imadmin start server	Starts only the server.
imadmin stop server	Stops only the server.
imadmin refresh server	Refreshes only the server.
imadmin start multiplexor	Starts only the multiplexor.
imadmin stop multiplexor	Stops only the multiplexor.
imadmin refresh multiplexor	Refreshes only the multiplexor.
imadmin start agent-calendar	Starts only the Calendar agent.
imadmin stop agent-calendar	Stops only the Calendar agent.
imadmin refresh agent-calendar	Refreshes only the Calendar agent.

TABLE C–1 imadmin Commands and Descriptions *(Continued)*

Command	Description
`imadmin start watchdog`	Starts only the watchdog.
`imadmin stop watchdog`	Stops only the watchdog.
`imadmin refresh watchdog`	Refreshes only the watchdog.
`imadmin version`	Displays the version.

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` *alt-config-file*	Used with the `start` and `refresh` actions, to specify a different configuration file other than `/etc/opt/SUNWiim/config/iim.conf` file
`-h`	Displays help on the `imadmin` 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 *[component]* 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.

◆ ◆ ◆ APPENDIX D

Instant Messaging APIs

This chapter describes the APIs used by Instant Messaging in the following sections:

- "Instant Messaging APIs Overview" on page 285
- "Instant Messaging Service API" on page 285
- "Messenger Beans" on page 286
- "Service Provider Interfaces" on page 286

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" on page 285
- "Messenger Beans" on page 286
- "Service Provider Interfaces" on page 286

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" on page 286
- "Message Conversion API" on page 287

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 Chapter 18, "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.

◆ ◆ ◆ APPENDIX E

Instant Messaging LDAP Schema

This appendix 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 `imadmin assign_services` command.
`sunPresenceUser`	Contains user presence properties. Added to user entries under base DN specified when you run the `imadmin assign_services` command.
`sunIMNews`	Contains news channel properties. If `userprops.store` is set to `ldap`, 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 `sunIMNews` objectclass.
`sunIMConference`	Contains conference room properties. If `userprops.store` is set to `ldap`, 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 `sunIMConference` objectclass.

Index

A

access control, 189-191, 191-193
access control files, 191-193, 193-196
  default privileges, 195-196
  example, 196
  format, 195-196
  location, 193-196
  server pool and, 193-196
Access Manager, 73-76
  disabling users with, 150
  policies, 191-193, 196-209
ACL, *See* access control
actions, used with `imadmin`, 283-284
activating SSL, 125-128
administering
  conference rooms, 175-176
  news channels, 175-176
  redirect server, 88-90
agent
  common container, 243
  mfwk, 243
agent-calendar, `imadmin` command and, 284
allowed client inactivity time
  httpbind.inactivity parameter, 110-111
  setting for gateway, 110-111
API
  archive provider, 286-287
  authentication provider, 287
  for Instant Messaging, 285
  Instant Messaging service, 285-286
  message conversion, 287
  messenger beans, 286
API *(Continued)*
  service provider interfaces overview, 286-287
application server
  disabling gateway for, 109
  enabling gateway for, 109
archive provider API, 286-287
  description, 286-287
assign Instant Messaging services to existing users, 39-41
`assign_services`, assign Instant Messaging services to existing users, 39-41
authentication provider, 287
authentication provider API, 286-287

B

backing up Instant Messaging data, 105-106
BEA Web Container, custom configuration, 39

C

CAC, *See* common agent container
calendar agent
  checking status of, 103
  logging levels, 137
  refreshing, 102
  starting, 100-101
  stopping, 101
certificates, redirect server, 92
changing
  configuration parameters, 104

changing *(Continued)*
user privileges, 194
checking
calendar agent status, 103
component status, 99-103
multiplexor status, 103
redirect server status, 88-90
server status, 103
watchdog status, 103
checklist
configuration, 29-38
for HA configuration, 57-60
client
configuring, 47-48
configuring systems for, 48-49
gateway response wait time and, 112
launching, 49-50
load balancing connections, 77-82
monitoring retries, 91
retry monitoring, 91
see-other-hosts error, 92
standalone application, 50
troubleshooting certificates, 92
web browser and, 49
client inactivity, *See* allowed client inactivity time
client redirect, *See* redirect
codebase, gateway and, 108-114
commands, `iwadmin`, 180
Common Agent Container, 243
component JID, gateway, 113-114
components, used with `imadmin` command, 284
concurrent requests, configuring for gateway, 109-110
Confdir_list, 69-70
Confdir_list RTR parameter, 69-70
conference rooms, administering, 175-176
configuration
checklist, 29-38
Instant Messaging, 39-41
verifying for HA, 67
configuration file, configuring non-default for gateway, 114
configuration files, 53-55, 55
access control files, 191-193
gateway, 107-108
configuration parameters
logging, 255-256
multiple servers, 261-262
multiplexor, 263-264
new user registration, 151-152
server, 256-261
user source, 253-255
configure
BEA Web Container, 39
silent, 42-44
`configure` utility
enabling gateway using, 109
gateway deployment and, 108-114
configuring
Access Manager policies for Instant Messaging, 74-75
after install, 39-41
after upgrade, 39-41
client, 47-48
client systems, 48-49
gateway logging configuration file location, 116
HA, 62-65
HA, 60-67
instance as multiplexor, 31
Instant Messenger for user registration, 152-153
logical host for HA, 65-66
redirect server, 85-87
server as redirect server, 87
server for user registration, 151-152
server pool, 78-81
sever-to-server communication, 78-81
sever-to-server federation, 95-97
SSL, 123-134
SSO for Instant Messaging, 74-75
StartTLS, 123-134
storage resource for HA, 66
TLS, 123-134
conflict resolution for proxies, 176-177, 177
contact lists, including dynamic groups, 121-122
container, agent, 243
`content-type` header
httpbind.content_type parameter, 111
setting for gateway, 111
creating, redirect database, 90-91

customizing `index.html` and `im.html` files, 162-163

D

data, backing up, 105-106
delay, gateway round trip, 111-112
deployment
  multi-node, 77-78
  multiple servers
    multiple domains, 95-97
    single domain, 77-82
directories, *im-svr-base*`/work`, 108-114
directory structure, 53-55
disable, end user access, 150
disable server parameter, 31
disabling
  gateway, 109
  gateway logging, 115-116
  watchdog, 244
document converter API, *See* message conversion API
domain
  configuring support for gateway, 113-114
  httpbind.config parameter, 113-114
domain name, server parameter, 30
dynamic groups
  using in contact lists, 121-122
  using in search results, 121-122

E

enabling
  gateway, 109
  gateway logging, 115-116
  watchdog, 244
end user, *See* user
error, see-other-hosts, 92
example, redirect.partitions, 89
Extensible Messaging and Presence Protocol, *See* XMPP

F

failover
  in multi-node deployment, 78
  in server pool, 78
Failover_enabled, 69-70
Failover_enabled RTR parameter, 69-70
failover service, *See* HA
federating deployment, server-to-server
  communications, 95-97
federating servers, 95-97
files
  access control, 193-196
  gateway configuration file, 107-108
  gateway logging configuration file, 107-108
  gateway webapp configuration file, 107-108
  `messenger.properties`, 177
  messenger.properties, 176-177
  non-default gateway configuration file, 114
  redeploying resource files, 180
  redirect.hosts, 85
  redirect.partitions, 89-90
  resource files, 180
  watchdog log, 244-245
  `web.xml`, 114
firewall, accessing XMPP traffic using HTTP
  gateway, 107-118

G

gateway, 107-118, 118
  changing logging configuration file location, 116
  codebase and, 108-114
  component JID, 113-114
  configuration files, 107-108, 108
  `configure` utility and, 108-114
  configuring, 108-114, 114
    allowed client inactivity time, 110-111
    `content-type` HTTP header, 111
    disabling, 109
    enabling, 109
    gateway pool, 112-113
    Key IDs, 113-114
    logging configuration file location, 116
    non-default configuration file, 114

gateway, configuring *(Continued)*
number of concurrent requests, 109-110
response wait time, 112
round trip delay, 111-112
setting JEP 124 *hold* parameter, 110
supported domains, 113-114
configuring for network latency, 111-112
deploying on web container, 107-118
disabling, 109
disabling logging, 115-116
enabling, 109
enabling logging, 115-116
`httpbind.conf` configuration file, 107-108
`httpbind_log4j.conf` configuration file, 107-108
logging, 115-118, 118
configuration file location, 116
disabling, 115-116
enabling, 115-116
logging configuration file, 116
logging configuration file location, 116
logging levels, 115-116
non-default configuration file, 114
password, 113-114
resource files, 107-118
support for SSL, 114
support for StartTLS, 114
URL, 112-113
URL and nodeId, 112-113
`web.xml` configuration file, 107-108
gateway configuration file, using non-default, 108-114
gateway domain ID, *See* Key ID
generating, redirect database, 90-91
globally available disk, 60
granting users privilege to create conference rooms and news channels, 175-176
group ID
creating, 38
for HA, 60-61
UNIX system group, 38
gwdomain-id, *See* Key ID
gwdomain-id.componentjid parameter, 113-114
gwdomain-id.domain parameter, 113-114
gwdomain-id.hosts parameter, 113-114
gwdomain-id.password parameter, 113-114

H

HA
checklist, 57-60
choosing local disk, 60
choosing shared disk, 60
HA
configuration steps, 60-67
HA
configuring, 62-65
HA
configuring, 57-72
configuring storage resource, 66
HA
installation directory, 61
HA
logical host, 30
HA
overview, 57-60
permissions, 57-60
HA
refreshing components, 102
registering resource, 66-67
registering `SUNWiimsc`, 66-67
related documentation, 71-72
resource group, 65-66
restarting service, 68
`scswitch`, 68
HA
software requirements, 57-60
starting components, 100-101
HA
starting service, 68
stopping components, 101
stopping service, 68
troubleshooting configuration, 67
HA
user and group ID, 60-61
HA
verifying configuration, 67
HAStoragePlus, registering storage resource, 66
header, gateway HTTP `content-type`, 111
help, for `imadmin` command, 283
High Availability
See HA

host name parameter, 30
HTTP, gateway to XMPP, 107-118
HTTP, setting gateway `content-type` header, 111
HTTP/XMPP Gateway, *See* gateway
`httpbind.conf`
  changing gateway logging configuration file location using, 116
  gateway configuration file, 107-108
  gwdomain-id.componentjid parameter, 113-114
  gwdomain-id.domain parameter, 113-114
  gwdomain-id.hosts parameter, 113-114
  gwdomain-id.password parameter, 113-114
  httpbind.config parameter, 113-114
  httpbind.content_type parameter, 111
  httpbind.hold parameter, 110
  httpbind.inactivity parameter, 110-111
  httpbind.pool.nodeId parameter, 112-113
  httpbind.pool.support parameter, 112-113
  httpbind.requests parameter, 109-110
  httpbind.round_trip_delay parameter, 111-112
  httpbind.wait_time parameter, 112
  setting concurrent requests with, 109-110
  setting *httpbind.config* parameter, 113-114
  setting *httpbind.content_type* parameter, 111
  setting *httpbind.inactivity* parameter, 110-111
  setting *httpbind.nodeId* parameter, 112-113
  setting *httpbind.pool.support* parameter, 112-113
  setting *httpbind.round_trip_delay* parameter, 111-112
  setting *httpbind.wait_time* parameter, 112
  setting JEP 124 *hold* attribute with, 110
httpbind.config
  configuring gateway Key IDs, 113-114
  configuring gateway supported domains, 113-114
httpbind.config.file, configuring non-default gateway configuration file, 114
httpbind.content_type, configuring gateway `content-type` HTTP header, 111
httpbind.inactivity, configuring for allowed client inactivity time, 110-111
`httpbind_log4j.conf`
  enabling gateway using, 115-116
  gateway logging configuration file, 107-108, 115-116, 116
`httpbind_log4j.conf` *(Continued)*
  location, 115-116, 116
httpbind.log4j.config gateway parameter, 115-116, 116
httpbind.nodeId, configuring gateway pool, 112-113
httpbind.pool.support, configuring gateway pool, 112-113
httpbind.round_trip_delay, configuring gateway round trip delay, 111-112
httpbind.wait_time, configuring gateway response wait time, 112

I

ID, for state file, 42-44
iim_agent.httpbind.enable, parameter, 109
`iim.conf`, enabling gateway using, 109
iim.conf
  redirect.partitions file and, 89-90
  specifying user partitions using, 89-90
`iim.conf` file, 55, 96-97, 104
  location, 249
  syntax, 250
`iim.conf` parameter issues, 239-243
iim.instancedir
  creating multiple instances with, 44-46
  parameter, 44-46
iim.instancevardir
  creating multiple instances with, 44-46
  parameter, 44-46
iim_server.redirect.db.partitions parameter, 85-87
iim_server.redirect.db.partitionsize parameter, 85-87
iim_server.redirect.db.users parameter, 85-87
iim_server.redirect.pollfrequency parameter, 85-87
iim_server.redirect.provider parameter, 85-87
iim_server.redirect.roundrobin.partitions parameter, 85-87
iim_server.redirect.to.nodename.host parameter, 85-87
iim_server.redirect.to.nodename.host.usessl parameter, 85-87
iim_server.redirect.to parameter, 85-87
iim_server.ssoprovider parameter, 74
iim_server.usesso parameter, 74
`im.jnlp`
  proxy argument, 177

im.jnlp *(Continued)*
proxy_host argument, 177
im-svr-base/work directory, 108-114
imadmin assign_services, 39-41
imadmin check command, *See* imadmin status command
imadmin command
actions, 283-284
checking component status with, 99-103
components, 284
finding, 282
getting help for, 283
invoking, 281
location, 282
options, 283
overview, 281
reference, 281-284
refreshing components with, 99-103
requirements for, 281
starting components with, 99-103
stopping components with, 99-103
syntax, 283
imadmin refresh command, 283-284
imadmin script, 44-46
imadmin script, script for multiple instances, 44-46
imadmin start command, 283-284
imadmin status command, 103, 283-284
watchdog and, 243-245
imadmin stop command, 283-284
imadmin tool, *See* imadmin command
imres.jnlp file, 164
inetgroup, 38
inetuser, 38
install, configuring after, 39-41
installation directory
for HA, 61
parameter, 30
installation directory parameter, 30
instance
creating new for Instant Messaging, 44-46
starting with imadmin command, 44-46
instance list, *See* redirect server
Instant Messaging
access control, 189-191
Instant Messaging *(Continued)*
APIs, 285
backing up, 105-106
configuring, 39-41
custom installation, 39-41
logging overview, 135-147
web browser and, 49
Instant Messaging gateway, *See* gateway
Instant Messaging redirect server, *See* redirect server
Instant Messaging service API, 285-286
Instant Messenger
configuring resource files for multiple instances, 44-46
gateway response wait time and, 112
launching, 49-50
redirect
See redirect
redirecting connections, 77-82
standalone application, 50
iwadmin command, 180

J

Java Web Start, 157
setting proxy with, 176-177
JEP 124 *hold* attribute
concurrent gateway requests and, 109-110
configuring for client requests, 110
configuring for gateway, 110
httpbind.hold parameter and, 110
httpbind.requests parameter and, 109-110
performance degradation and, 109-110
JNLP mime types, 47

K

Key ID
gateway configuration parameters, 113-114
gateway supported domains, 113-114
httpbind.config parameter, 113-114

L

latency, *See* see network latency
launching
  client, 49-50
  Instant Messenger, 49-50
LDAP, schema for Instant Messaging, 289
LDAP directory server, enable server to search as a specific user, 120-121
load balancer, *See* redirect server
load balancing, client connections, 77-82
load director, 77-78
  See redirect server
local disk, for HA, 60
log4j, gateway and, 115-118
log4j.logger.gateway gateway parameter, 115-116
logging
  gateway, 115-118, 118
  log4j, 115-118
  monitoring and trimming log files, 137
  overview, 135-147
  redirect server, 88-90
  setting levels, 145
  watchdog, 244-245
logging configuration file, changing gateway location, 116
logging levels, 136-137
  enabling gateway logging using, 115-116
logical host, 30
  configuring for HA, 65-66
  configuring resource group, 65-66

M

managing, logging, 135-147
map
  network-to-partition, 84
  partition, 85
  redirect database, 84
  user-to-network, 84
  user-to-partition, 84
message content filtering, 287
message conversion API, 286-287
  and content filtering, 287
  and translation engine, 287
message conversion API *(Continued)*
  and virus checking, 287
message converter, *See* message conversion API
messenger beans, 286
`messenger.properties` file
  location, 177
  proxy conflict with, 177
messenger.properties file, proxy conflict with, 176-177
`messenger.properties` file
  user preferences, 177
mfwk agent, 243
mime types file, 47
Monitor_retry_count, 69-70
Monitor_retry_count RTR parameter, 69-70
Monitor_retry_interval, 69-70
Monitor_retry_interval RTR parameter, 69-70
monitoring
  client retries, 91
  physical hosts by redirect server, 91
  redirect server and, 91
multi-node deployment, 77-78
  failover in, 78
multiplexor
  checking status of, 103
  configuring, 31
  creating multiple instances of, 44-46
  `imadmin` command and, 284
  listenport parameter, 165
  `listenport` parameter, 104
  logging levels, 137
  refreshing, 102
  remote server host name and, 31
  starting with watchdog, 100-101
  stopping, 101
multiplexor port number, 31
multiplexor port Number parameter, 31

N

network latency, configuring gateway for, 111-112
network-to-partition map, 84
news channels, administering, 175-176
nodeId, gateway URL and, 112-113

O

objectclass
Instant Messaging schema, 289
sunIMConference, 289
sunIMNews, 289
sunIMuser, 289
sunPresenceUser, 289
options, `imadmin` command, 283
overview
`imadmin` command, 281
server pool, 77-78

P

parameters
See also RTR file parameters
disable server, 31
gwdomain-id.componentjid gateway parameter, 113-114
gwdomain-id.domain gateway parameter, 113-114
gwdomain-id.hosts gateway parameter, 113-114
gwdomain-id.password gateway parameter, 113-114
httpbind.config.file gateway parameter, 114
httpbind.config gateway parameter, 113-114
httpbind.content_type gateway parameter, 111
httpbind.hold gateway parameter, 110
httpbind.inactivity gateway parameter, 110-111
httpbind.log4j.config gateway parameter, 115-116, 116
httpbind.pool.nodeId gateway parameter, 112-113
httpbind.pool.support gateway parameter, 112-113
httpbind.requests gateway parameter, 109-110
httpbind.round_trip_delay gateway parameter, 111-112
httpbind.wait_time gateway parameter, 112
iim_agent.httpbind.enable gateway parameter, 109
iim.instancedir, 44-46
iim.instancevardir, 44-46
iim_server.redirect.db.partitions parameter, 85-87
iim_server.redirect.db.partitionsize parameter, 85-87
iim_server.redirect.db.users parameter, 85-87
iim_server.redirect.pollfrequency parameter, 85-87
iim_server.redirect.provider parameter, 85-87
parameters *(Continued)*
iim_server.redirect.roundrobin.partitions parameter, 85-87
iim_server.redirect.to.nodename.host parameter, 85-87
iim_server.redirect.to.nodename.host.usessl parameter, 85-87
iim_server.redirect.to parameter, 85-87
iim_server.ssoprovider, 74
iim_server.usesso, 74
installation directory, 30
log4j.logger.gateway gateway parameter, 115-116
multiplexor port number, 31
remote server host name, 31
server domain name, 30
server host name, 30
server port number, 30
partition, *See* user partition
partition map, 85
password, gateway, 113-114
peer-to-peer communications, *See* server-to-server communications
performance
gateway round trip delay and, 111-112
JEP 124 *hold* parameter and, 109-110
permissions
redirect server, 92
required for HA, 57-60
policies, 189
configuring Access Manager, 74-75
polling
physical hosts by redirect server, 91
setting frequency for redirect server, 91
pool
See also pooling
support for gateway, 112-113
pooling
configuring for gateway, 112-113
httpbind.pool.nodeId parameter, 112-113
httpbind.pool.support parameter, 112-113
pooling servers, 78-81
port numbers
multiplexor, 31
server, 30

port numbers *(Continued)*
server-to-server, 30
privacy, overview, 189-191
privileges, 189
Probe_timeout RTR parameter, 69-70
proxy
configuring for all clients, 177
configuring for single client, 176-177
conflict resolution, 176-177, 177
setting in `im.jnlp`, 177
setting with Java Web Start, 176-177
proxy argument
configuring proxy with, 177
in `im.jnlp`, 177
proxy_host argument, in `im.jnlp`, 177
proxy settings, 176-177

R

rdadmin
generating redirect database, 90-91
updating redirect database, 90-91
rdadmin utility, 90-91
redeploying resource files, 180
redirect
RFC 3920, 84
see-other-host stream error, 84
server overview, 85
redirect database
generating with rdadmin, 90-91
maps, 84
redirect.hosts, generating instance list from, 85
redirect.hosts file, 85
redirect.partitions file, example, 89
redirect server
administering, 88-90
as partition host, 93
certificate troubleshooting, 92
checking status of, 88-90
client retry monitoring, 91
configuring, 85-87
configuring Instant Messaging as, 87
determining partition size, 92
instance list, 85
redirect server *(Continued)*
LDAP and, 92
logging, 88-90
maximum partition size, 88-90
optimizing server pool using, 83-93
overview, 83-85
partition list, 88-90
physical host monitoring, 91
polling physical hosts, 91
redirect.hosts file, 85
redirect.partitions file, 89-90
refreshing, 88-90
RFC 3920, 92
setting polling frequency, 91
SSL, 85
starting, 88-90
StartTLS, 85
stopping, 88-90
third party clients, 92
troubleshooting, 92-93
user partitioning algorithm, 84
redirect service, 77-82
refresh, `imadmin` command and, 283-284
refreshing
calendar agent, 102
components, 99-103
in an HA environment, 102
redirect server, 88-90
server and multiplexor, 102
watchdog, 102
registration
as new user, 153
configuring Instant Messenger for, 152-153
configuring server for, 151-152
configuring to allow, 150-153
new user, 150-153
remote server host name parameter, 31
requirements, HA, 57-60
resource
creating for HA, 66-67
registering for HA, 66-67
resource files
redeploying with `iwadmin`, 180
`web.xml`, 114

resource group
configuring for HA, 65-66
configuring with logical host, 65-66
resource type, *See* resource
response wait time
httpbind.wait_time parameter, 112
setting for gateway, 112
restarting, HA service, 68
RFC 3920, 84
redirect server, 92
round trip delay
httpbind.round_trip_delay parameter, 111-112
setting for gateway, 111-112
RTR file parameters, 69-70
Probe_timeout, 69-70
Server_Root, 69-70

S

scaling
load balancing, 77-82
redirect service, 77-82
using redirect, 83-85
using server pool, 77-82
schema, for Instant Messaging, 289
`scswitch`, 68
search results, including dynamic groups, 121-122
security, overview, 189-191
see-other-host stream error, 84
server
as redirect server, 85
changing configuration parameters, 104
checking status of, 103
configuration file, 55
creating multiple instances of, 44-46
deploying multiple, 77-82, 95-97
enabling as multiplexor, 31
host name parameter, 30
`imadmin` command and, 284
logging levels, 137
refreshing, 102
remote host name, 31
scaling, 77-82
server pool, 78-81
server *(Continued)*
server-to-server communication, 78-81
server-to-server communications, 95-97
starting with watchdog, 100-101
stopping, 101
server configuration file, location, 249
server domain name parameter, 30
server farm, *See* server pool
server host name parameter, 30
server pool
access control files and, 193-196
configuring server-to-server communication in, 78-81
failover, 78
overview, 77-78
scaling deployment with, 77-82
server port number parameter, 30
Server_root RTR parameter, 69-70
server-to-server communications, 78-81, 95-97
federating deployment, 95-97
setting log file levels, 145
silent configuration, 42-44
single sign-on
See SSO
site policies
Access Manager and, 191-193
overview, 189-191
sizing, user partitions for redirect, 92
SSL
activating, 125-128
configuring, 123-134
gateway and, 114
redirect server and, 85
using in Instant Messaging, 123-134
SSO, 73-76
SSO, configuration parameters, 74-75
SSO
configuring, 74-75
limitations, 73-74
troubleshooting, 75-76
using Access Manager, 287
using authentication provider API, 287
standalone application, Instant Messenger and, 50
start, `imadmin` command and, 283-284

starting
calendar agent, 100-101
components, 99-103
HA service, 68
in an HA environment, 100-101
redirect server, 88-90
server and multiplexor, 100-101
watchdog, 100-101
StartTLS
configuring, 123-134
gateway and, 114
redirect server and, 85
using in Instant Messaging, 123-134
state file
generating, 42-44
ID, 42-44
status
checking for components, 103
`imadmin check` command
See `imadmin status`
`imadmin` command and, 103, 283-284
`imadmin status` command, 103
watchdog, 244
stop, `imadmin` command and, 283-284
stopping
calendar agent, 101
components, 99-103
HA service, 68
in an HA environment, 101
redirect server, 88-90
server and multiplexor, 101
watchdog, 101
storage resource
configuring for HA, 66
configuring for HAStoragePlus, 66
registering, 66
Sun Cluster, 57-60
Sun Java System Instant Messaging
SSL and, 123-134
StartTLS and, 123-134
TLS and, 123-134
Sun Java System Instant Messaging server, directory
structure, 53-55
Sun Java System Instant Messenger
customizing, 160
proxy settings, 176-177
sunIMConference objectclass, 289
sunIMNews objectclass, 289
sunIMuser objectclass, 289
sunPresenceUser objectclass, 289
`SUNWiimsc`, registering for HA, 66-67
supported domains, *See* domain
syntax
`iim.conf`, 250
`imadmin` command, 283
`sysTopicsAdd.acl` file, 196

T

TLS
configuring, 123-134
using in Instant Messaging, 123-134
tools, disabling gateway using, 109
translation engine, 287
troubleshooting
HA configuration, 67
LDAP and redirect server, 92
redirect server, 92-93
redirect server certificates, 92
see-other-hosts stream error, 92
third party clients, 92

U

updating, redirect database, 90-91
upgrade, configuring after, 39-41
URL, gateway, 112-113
user
configuring Instant Messenger for registration, 152-153
configuring server for registration, 151-152
disabling Instant Messaging access, 150
networks, 84
partitions, 84
preferences file `messenger.properties`, 177

user *(Continued)*
privileges
changing, 194
creating conference rooms, 175-176
creating news channels, 175-176
registering as new, 153
registration, 150-153
user-to-network map, 84
user administration, 149-155
user ID
creating, 38
for HA, 60-61
UNIX system user, 38
user network, defined, 84
user partition
algorithm used by redirect server, 84
defined, 84
determining size, 92
list for redirect server, 88-90
redirect server and, 84
redirect server as host, 93
specifying maximum size, 88-90
weak ties, 84
user partitions
redirect.partitions file and, 89-90
redirect.partitions file example, 89
user properties, LDAP and redirect server, 92
user provisioning, 149-155
user-to-network map, defined, 84
user-to-partition map, 84
utilities, rdadmin, 90-91

V

verifying, HA configuration, 67
virus checking, 287

W

wait time, *See* response wait time
watchdog
checking status of, 103
disabling, 244
watchdog *(Continued)*
enabling, 244
`imadmin` command and, 284
`imadmin status` command, 243-245
logging, 244-245
logging levels, 137
refreshing, 102
starting, 100-101
status of, 244
stopping, 101
web, Instant Messaging access through XMPP, 107-118
web browser, Instant Messaging and, 49
web container
disabling gateway for, 109
enabling gateway for, 109
`web.xml`
gateway webapp configuration file, 107-108
httpbind.config.file parameter, 114
`web.xml` file, non-default gateway configuration file and, 108-114
webapp configuration file
gateway, 107-108
`web.xml`, 107-108

X

XMPP, gateway to HTTP, 107-118
XMPP/HTTP Gateway, *See* gateway