Sun Java™ System

Portal Server 6 Administration Guide

2005Q1

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

Part No: 817-7691-10

Contents

List of Procedures

Preface

Who Should Use This Book

You should read this book if you are responsible for installing, administering, and configuring Portal Server at your site.

Before You Read This Book

Before you administer Portal Server, you must be familiar with the following concepts:

- Basic Solaris™ administrative procedures
- LDAP
- Sun Java™ System Directory Server (formerly Sun ONE Directory Server)
- Sun Java™ System Access Manager (formerly Sun ONE Identity Server, and Sun Java System Identity Server)

Depending on the Web container that you are using, you should be familiar with one or more of the following:

- Sun Java™ System Web Server (formerly Sun ONE Web Server)
- Sun Java System Application Server
- BEA WebLogic Server ™ 8.1 SP2
- IBM WebSphere® 5.1

NOTE The Sun™ Java System family of products was previously branded under the Sun™ ONE name.

How This Book Is Organized

This book contains the following chapters and appendices

Table 0-1 List of Chapters and Appendices

Chapter	Description
Preface	(This chapter)
Chapter 1, “Introduction to Administering the Sun Java System Portal Server”	This chapter describes the Portal Server architecture, protocols, and interfaces, and provides an overview of administering and customizing the product.
Chapter 2, “Post Installation Configuration”	This chapter provides instructions for performing post installation configuration tasks.
Chapter 3, “Creating and Deleting Instances of the Server”	This chapter provides instructions for creating and deleting instances of Portal Server.
Chapter 4, “Setting Up the Portal Server to Use Secure External LDAP Directory Server”	This chapter provides instructions for setting up Portal Server to use a secure external LDAP server.
Chapter 5, “Configuring the Portal Server to Run as User Non-Root”	This chapter provides instructions for configuring the Portal Server to run as user non-root.
Chapter 6, “Administering Authentication, Users, and Services”	This chapter describes how to use Sun Java System Identity Server to administer authentication, users, and services.
Chapter 7, “Configuring Delegated Administration”	This chapter describes how to configure delegated administration for Portal Server.
Chapter 8, “Administering the Portal Desktop Service”	This chapter describes how to administer the Portal Server Desktop service.
Chapter 9, “Administering the Web Services for Remote Portlets (WSRP) Service”	This chapter provides information and instructions for using Web Services for Remote Portlets (WSRP).
Chapter 10, “Administering the Display Profile”	This chapter describes how to administer the Portal Server display profile component.
Chapter 11, “Administering the NetMail Service”	This chapter describes how to administer the NetMail service.
Chapter 12, “Administering the Rewriter Service”	This chapter describes how to administer the Rewriter service.

Table 0-1 List of Chapters and Appendices *(Continued)*

Chapter	Description
Chapter 13, “Administering the Search Engine Service”	This chapter describes how to configure and administer the Search Engine service.
Chapter 14, “Administering the Search Engine Robot”	This chapter describes the Search Engine robot and its corresponding configuration files.
Chapter 15, “The Pre-defined Robot Application Functions”	This chapter describes the pre-defined robot application functions. You can use these functions to create and modify filter definitions.
Chapter 16, “Administering the Subscriptions Service”	This chapter describes how to configure and administer the Subscriptions service.
Chapter 17, “Configuring the Communication Channels”	This chapter provides information on the communication channels for Portal Server.
Chapter 18, “Managing the Portal Server System”	This chapter describes the various administrative tasks associated with maintaining the Portal Server system.
Chapter 19, “Tuning the Portal Server”	This chapter provides information about the perftune script that comes with Portal Server.
Appendix A, “SSO Adapter Templates and Configurations”	This appendix provides a reference for the communication channels for Portal Server.

Conventions Used in This Book

The tables in this section describe the conventions used in this book.

Typographic Conventions

The following table describes the typographic changes used in this book.

Table 2 Typographic Conventions

Typeface	Meaning	Examples
AaBbCc123 (Monospace)	API and language elements, HTML tags, web site URLs, command names, file names, directory path names, onscreen computer output, sample code.	Edit `your.login` file. Use `ls -a` to list all files. `% You have mail.`

Table 2 Typographic Conventions *(Continued)*

Typeface	Meaning	Examples
AaBbCc123 (Monospace bold)	What you type, when contrasted with onscreen computer output.	% **su** Password:
AaBbCc123 (Italic)	Book titles, new terms, words to be emphasized. A placeholder in a command or path name to be replaced with a real name or value.	Read Chapter 6 in the *User's Guide*. These are called *class* options. Do *not* save the file. The file is located in the *install-dir*/bin directory.

Symbols

The following table describes the symbol conventions used in this book.

Table 3 Symbol Conventions

Symbol	Description	Example	Meaning
[ ]	Contains optional command options.	ls [-l]	The -l option is not required.
{ | }	Contains a set of choices for a required command option.	-d {y|n}	The -d option requires that you use either the y argument or the n argument.
-	Joins simultaneous multiple keystrokes.	Control-A	Press the Control key while you press the A key.
+	Joins consecutive multiple keystrokes.	Ctrl+A+N	Press the Control key, release it, and then press the subsequent keys.
>	Indicates menu item selection in a graphical user interface.	File > New > Templates	From the File menu, choose New. From the New submenu, choose Templates.

Default Paths and File Names

The following table describes the default paths and file names used in this book.

Table 4 Default Paths and File Names

Term	Description
PortalServer-base	Represents the base installation directory for Portal Server. The Portal Server 2005Q1 default base installation and product directory depends on your specific platform: Solaris™ systems `/opt` Linux systems: `/opt/sun`
AccessManager-base	Represents the base installation directory for Access Manager. The Access Manager 2005Q2 default base installation and product directory depends on your specific platform: Solaris™ systems: `/opt/SUNWam` Linux systems: `/opt/sun/identity`
DirectoryServer-base	Represents the base installation directory for Sun Java System Directory Server. Refer to the product documentation for the specific path name.
ApplicationServer-base	Represents the base installation directory for Sun Java System Application Server. Refer to the product documentation for the specific path name.
WebServer-base	Represents the base installation directory for Sun Java System Web Server, or BEA WEblogic 8.1 SP2, or IBM WebSphere. Refer to the product documentation for the specific path name.

Shell Prompts

The following table describes the shell prompts used in this book.

Table 5 Shell Prompts

Shell	Prompt
C shell on UNIX or Linux	*machine-name*%
C shell superuser on UNIX or Linux	*machine-name*#
Bourne shell and Korn shell on UNIX or Linux	$
Bourne shell and Korn shell superuser on UNIX or Linux	#

Table 5 Shell Prompts

Shell	Prompt
Windows command line	`C:\`

Related Documentation

The http://docs.sun.com℠ web site enables you to access Sun technical documentation online. You can browse the archive or search for a specific book title or subject.

Books in This Documentation Set

The following table summarizes the books included in the Portal Server core documentation set.

.

Book Title	Description
Portal Server Deployment Planning Guide http://docs.sun.com/db/doc/817-6257	Describes how to plan for and deploy Sun Java System Portal Server software.
Portal Server Administration Guide http://docs.sun.com/db/doc/817-7691	Describes how to administer Sun Java System Portal Server 6 using the Access Manager administration console and the command line.
Portal Server Secure Remote Access Administration Guide http://docs.sun.com/db/doc/817-7693	Describes how to administer Sun Java System Portal Server 6 Secure Remote Access.
Portal Server Release Notes http://docs.sun.com/db/doc/817-7699	Available after the product is released. Contains last-minute information, including a description of what is new in this current release, known problems and limitations, installation notes, and how to report issues with the software or the documentation.
Portal Server Technical Reference Guide http://docs.sun.com/db/doc/817-7696	Provides detailed information on the Sun Java System Portal Server technical concepts (such as Display Profile, Rewriter), command line utilities, tag libraries (in the software), and files (such as templates and JSPs). This guide serves as a single source for such essential background information.

Other Portal Server Documentation

Other Portal Server books include:

- *Portal Server Desktop Customization Guide*

  http://docs.sun.com/doc/817-5318

- *Portal Server Developer's Guide*

  http://docs.sun.com/doc/817-5319

- *Portal Server Mobile Access Developer's Guide*

  http://docs.sun.com/doc/817-6258

- *Portal Server Mobile Access Developer's Reference*

  http://docs.sun.com/doc/817-6259

- *Portal Server Mobile Access Deployment Planning Guide*

  http://docs.sun.com/doc/817-6257

- *Portal Server Mobile Access Tag Library Reference*

  http://docs.sun.com/doc/817-6260

Other Server Documentation

For other server documentation, go to the following:

- Directory Server documentation
  http://docs.sun.com/coll/DirectoryServer_04q2
- Web Server documentation
  http://docs.sun.com/coll/S1_websvr61_en
- Application Server documentation
  http://docs.sun.com/coll/s1_asseu3_en
- Web Proxy Server documentation
  http://docs.sun.com/prod/s1.webproxys#hic

Accessing Sun Resources Online

For product downloads, professional services, patches and support, and additional developer information, go to the following:

- Download Center
  http://wwws.sun.com/software/download/
- Professional Services
  http://www.sun.com/service/sunps/sunone/index.html
- Sun Enterprise Services, Solaris Patches, and Support
  http://sunsolve.sun.com/
- Developer Information
  http://developers.sun.com/prodtech/index.html

Contacting Sun Technical Support

If you have technical questions about this product that are not answered in the product documentation, go to http://www.sun.com/service/contacting.

Related Third-Party Web Site References

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.

Sun Welcomes Your Comments

Sun is interested in improving its documentation and welcomes your comments and suggestions.

To share your comments, go to http://docs.sun.com and click Send Comments. In the online form, provide the document title and part number. The part number is a seven-digit or nine-digit number that can be found on the title page of the book or at the top of the document. For example, the title of this book is *Sun Java System Portal Server 2005Q1 Administration Guide*, and the part number is 817-7691.

Chapter 1

Introduction to Administering the Sun Java System Portal Server

Sun Java™ System Portal Server 6 2005Q1 product is a suite of integrated software products that allow enterprises to pull content from a variety of sources, personalize the content for a specific user or group of users, and aggregate content from these multiple sources into a single output format suitable for the specific user's device, such as a web browser.

This chapter provides basic information about the architecture of the product suite, the end user interface to the portal, the services implemented by the Portal Server software and how the are configured, and the tools used to administer the product. This chapter contains the following sections:

Architecture Overview

Portal Server is part of the Sun Java™ System architecture. Within the Sun Java System architecture, the Portal Server provides technologies that locate, connect, aggregate, present, communicate, personalize, notify, and deliver content. The content within Sun Java System is provided by web services. Portal Server does not provide web services itself. Rather, it is the mechanism by which a user interface is associated with web services and by which web services are made useful to people.

The Portal Server product architecture consists of a variety of integratable software products.This allows the Portal Server to leverage functions and services from its internal components as well as external supporting products. The Portal Server itself includes the following internal components: Desktop, NetMail, Rewriter, and Search. External supporting products include the Sun Java™ System Web Server, the Sun™ Java System Directory Server, and Sun Java™ System Access Manager (formerly Sun™ ONE Identity Server). The Portal Server implements the web application container, user, service, and policy management, authentication and single sign-on, administration console, directory schema and data storage, and protocol support from these external products rather than implementing them in the Portal Server product itself. For example, the Portal Server product uses the Sun Java System Web Server as its default web container.

NOTE The Portal Server supports the following web containers: Sun Java System Web Server, Sun Java™ System Application Server, IBM Websphere Application Server, and BEA Weblogic Application Server can also be used.

Sun Java System Portal Server 2005Q1 uses the following component products:

- Java™ 2 SDK (J2SDK™), Standard Edition
- A Web Container
- A web container—The Sun Java System Portal Server can be deployed on the following web containers:
  - Sun Java System Application Server 8.1
  - Sun Java System Web Server 6.1 SP4
  - BEA WebLogic Server™ 5.2 Patch 3
  - IBM WebSphere® Application Server 8.1 SP2
- Sun Java System Directory Server 5.1
- Sun Java System Access Manager 6.3

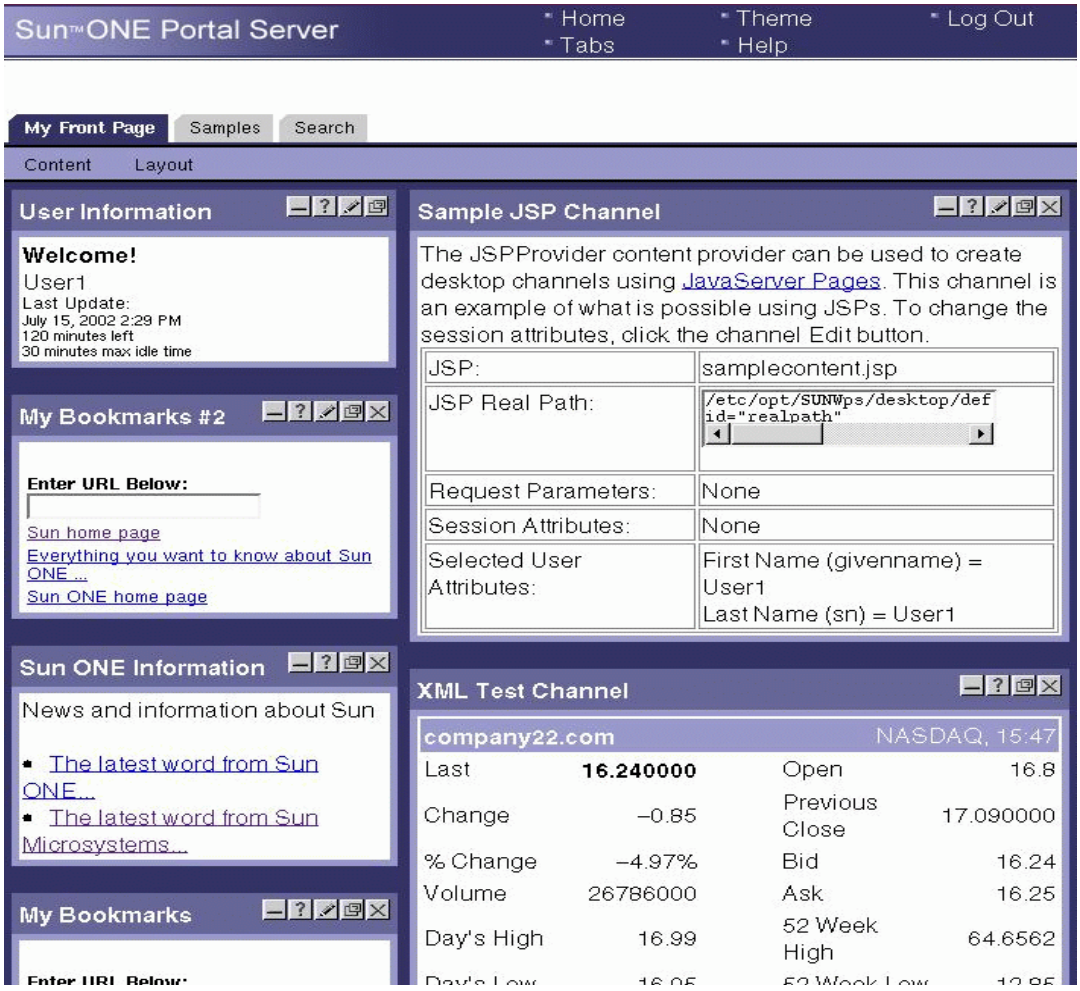
In addition, other Portal Server add-on software can be installed as well (for example, Sun Java™ System Portal Server, Secure Remote Access). Refer to the *Sun Java System Portal Server 6 2005Q1 Deployment Guide* for more information on the Portal Server architecture.

Portal Access Overview

Users typically access portal content through a web browser by requesting the URL for the portal's home page and authenticating through the Sun Java System Access Manager authentication service. Once authenticated, users are directed to the Portal Server Desktop.

Figure 1-1 on page 35 shows a sample Desktop from the Portal Server.

Figure 1-1 Portal Server Sample Desktop



The *Desktop* is the primary interface for the user to portal content. The Desktop service is implemented through a servlet, provider APIs, various channels, and various other support APIs and utilities. The Desktop uses programmatic entities called *providers* to generate content. A single unit of content is called a *channel*. Multiple channels of content can be aggregated together into *container channels* and arranged in a variety of formats such as tables or tabs on the Desktop. When a user accesses the portal, the Desktop references a *display profile* which stores content provider and channel data used to generate the user's content. As confusing as it may sound, the display profile does not actually define the overall layout, display, or organization of what users see on their Desktops. Fundamentally, the display profile exists only to provide property values for channels. Actually, the Desktop uses multiple display profiles stored as LDAP attributes at various levels or nodes in the Sun Java System Directory Server (top-most, organization, role, and user levels) to determine the content for a user. XML documents are used to define the display profile properties for each level and upload the property values into the LDAP node. At runtime, a user's display profile is created by merging the display profile properties defined at each level. Although a display profile document can be defined at each level, you do not need to have a display profile document at each level.

To extend support to store and retrieve specific property values based on a given client type (such as HTML or MAPI), the Portal Server software includes:

- Conditional properties for defining the filtering criteria (see "Conditional Properties" on page 208).
- The `authlessState` property for determining how the client is managed under authless authentication (see "Configuring Anonymous Authentication" on page 127).

Service Configuration Overview

The Portal Server is a Sun Java™ Enterprise System application and, as such, its services are defined and managed using the Access Manager Service Management System (SMS). Service-related data that is not server-specific is defined using an Extensible Markup Language (XML) file that adheres to an SMS Document Type Definition (DTD). Server-specific data can be stored in properties files that are local to the specific server. Each Portal Server service (Desktop, Netmail, Rewriter, and Search) has its own XML and properties files for presenting and modifying service specific data.

Access Manager Services

As explained in Architecture Overview, the Portal Server implements many functions and services using supporting products from the Sun Java System architecture that are external to the Portal Server itself. In particular, while previous versions of the Portal Server implemented many administrative capabilities internally, integration with the Access Manager allows the Portal Server to leverage the following administrative tools and services from the Access Manager product:

- Administration Console
- Service Management
- User Management
- Authentication / Single Sign-On

See Chapter 6, "Administering Authentication, Users, and Services" for information on administering Access Manager services.

Portal Server Services

In addition to the standard Access Manager services, the Portal Server uses the Access Manager administration console to administer its internal services (Desktop, NetMail, Rewriter, and Search).

Desktop

As stated in the previous section, the Desktop provides the primary end-user interface for Portal Server. The Desktop is the mechanism for extensible content aggregation through the Provider Application Programming Interface (PAPI). The Desktop includes a variety of providers that enable container hierarchy and the basic building blocks for building some types of channels. For storing content provider and channel data, the Desktop implements a display profile data storage mechanism on top of an Access Manager service. You can edit the display profile and other Desktop service data through the administration console. Refer to Chapter 8, "Administering the Portal Desktop Service" and Chapter 10, "Administering the Display Profile" for information on administering the Desktop and the display profile.

Rewriter

The Rewriter provides a Java class library for rewriting URL references in various web languages such as HTML, JavaScript™, and WML, and in HTTP Location headers (redirections). The Rewriter defines an Access Manager service for storing rules that define how rewriting is to be done and the data to be rewritten. You can edit Rewriter rules through the administration console. Refer to Chapter 12, "Administering the Rewriter Service" for information on administering Rewriter.

Search Engine

The Search Engine service provides basic and advanced search and browse channels for the Desktop. It uses a robot to create resource descriptions for documents that are available in the intranet, and stores these resource descriptions in an indexed database. Resource descriptions (RDs) can also be imported from another server or from a backup SOIF (Summary Object Interchange Format) file. The Search Engine includes Java and C APIs for submitting resource descriptions and for searching the database. The Search Engine database can also be used for storing other, arbitrary content, for example, a shared content cache for other content providers. You can edit Search Engine service data through the administration console. Refer to Chapter 13, "Administering the Search Engine Service" for information on administering Search.

NetMail

The NetMail service implements the NetMail (Java) and NetMail Lite email clients. These clients work with standard IMAP and SMTP servers. You can edit NetMail service data through the administration console. Refer to Chapter 11, "Administering the NetMail Service" for information on administering NetMail.

WSRP

Web Server Remote Portlets (WSRP) simplifies integration of remote applications and content into portals. You can administer the WSRP producers and consumers through the administration console. Refer to Chapter 9, "Administering the Web Services for Remote Portlets (WSRP) Service" for information on administering WSRP.

SSO Adapter

The Single Sign On (SSO) Adapter service allows end users to use applications, such as a portal server provider or any other web application, to gain authenticated access to various resource servers after signing in once. Refer to Appendix A, "SSO Adapter Templates and Configurations"for information on using SSO with Address book, Mail, and Calendar servers.

Subscriptions

The Subscriptions service enables users to create a profile of interest covering many sources of information. See Chapter 16, “Administering the Subscriptions Service” for information.

Configuration Mechanisms for Portal Server Services

The Portal Server uses a variety of configuration mechanisms to define, store and manage its services. This section contains five tables listing the configuration mechanisms used by each of the Portal Server internal services.

Table 1-1 on page 39 lists the configuration mechanisms for the Desktop service. The table is divided into two columns: Configuration Mechanism and Description. Configuration Mechanism lists the mechanisms and Description describes the purpose of the mechanism.

Table 1-1 Portal Server Desktop Configuration Mechanisms

Configuration Mechanisms	Description
Desktop Service Definition	Defines the Access Manager configuration attributes for the Desktop service. See the *Sun Java System Portal Server 2005Q1 Technical Service Guide* for more XML reference information.
Desktop Display Profile XML DTD	Defines the display configuration for the Desktop by defining provider and channel objects, and their properties. See the *Sun Java System Portal Server 2005Q1 Technical Service Guide* for more XML reference information.
Desktop Administration Console Module	Supplies the means by which you manage Portal Server services in the Access Manager framework. See Chapter 8, “Administering the Portal Desktop Service” for more information on administering the Desktop service configuration attributes. See Chapter 10, “Administering the Display Profile” for more information on administering the display profile.
Desktop CLI	Supplies the `dpadmin` and `par` command utilities for product administration. See the Sun Java System Portal Server 2005Q1 Technical Reference Guide for more information on administering command line utilities.
Desktop Configuration Properties File	Defines the server-specific parameters for the Desktop service. See the Sun Java System Portal Server 2005Q1 Technical Reference Guide for more information on search configuration properties.

Table 1-2 lists the configuration mechanisms for the Search service.The table is divided into two columns: Configuration Mechanism and Description. Configuration Mechanism lists the mechanisms and Description describes the purpose of the mechanism.

Table 1-2 Portal Server Search Configuration Mechanisms

Configuration Mechanisms	Description
Search Service Definition	Defines the Access Manager configuration attributes for the Search service. See the Sun Java System Portal Server 2005Q1 Technical Reference Guide for more information on schema reference.
Search Administration Console Module	Supplies the means by which you manage Portal Server Search service data in the Access Manager framework. See Chapter 13, "Administering the Search Engine Service" for more information.
Search CLI	Supplies the `rdmgr`, `sendrdm`, and `StartRobot` command utilities for product administration. See the Sun Java System Portal Server 2005Q1 Technical Reference Guide for more information on administering command line utilities.
Search Configuration Properties File	Defines the server-specific parameters for the Search service. See the Sun Java System Portal Server 2005Q1 Technical Reference Guide for more information on search configuration properties.
Robot Configuration Files	Define the behavior of the Search Engine robots. There are four robot configuration files. See Chapter 14, "Administering the Search Engine Robot" and Chapter 15, "The Pre-defined Robot Application Functions" for more information.

Table 1-3 lists the configuration mechanisms for the Rewriter service.The table is divided into two columns: Configuration Mechanism and Description. Configuration Mechanism lists the mechanisms and Description describes the purpose of the mechanism.

Table 1-3 Portal Server Rewriter Configuration Mechanisms

Configuration Mechanisms	Description
Rewriter Service Definition	Defines the Access Manager configuration attributes for the Rewriter service. See Sun Java System Portal Server 2005Q1 Technical Reference Guide for more information on schema reference.
Rewriter Rules XML DTD	See the Sun *Java System Portal Server 2005Q1 Technical Reference Guide* for more XML reference information.
Rewriter Administration Console Module	Supplies the means by which you manage Portal Server Rewriter service data in the Access Manager framework. See Chapter 12, "Administering the Rewriter Service" for more information.

Table 1-3 Portal Server Rewriter Configuration Mechanisms

Configuration Mechanisms	Description
Rewriter CLI	Supplies the `rwadmin` command utility for product administration. See the Sun Java System Portal Server 2005Q1 Technical Reference Guide for more information on administering command line utilities.

Table 1-4 lists the configuration mechanisms for the NetMail service.The table is divided into two columns: Configuration Mechanism and Description. Configuration Mechanism lists the mechanisms and Description describes the purpose of the mechanism.

Table 1-4 Portal Server NetMail Configuration Mechanisms

Configuration Mechanisms	Description
NetMail Service Definition	Defines the Access Manager configuration attributes for the NetMail service. See Sun Java System Portal Server 2005Q1 Technical Reference Guide for more information on schema reference.
NetMail Administration Console Module	Supplies the means by which you manage Portal Server NetMail service data in the Access Manager framework. See the Chapter 11, “Administering the NetMail Service” for more information.

Administration Overview

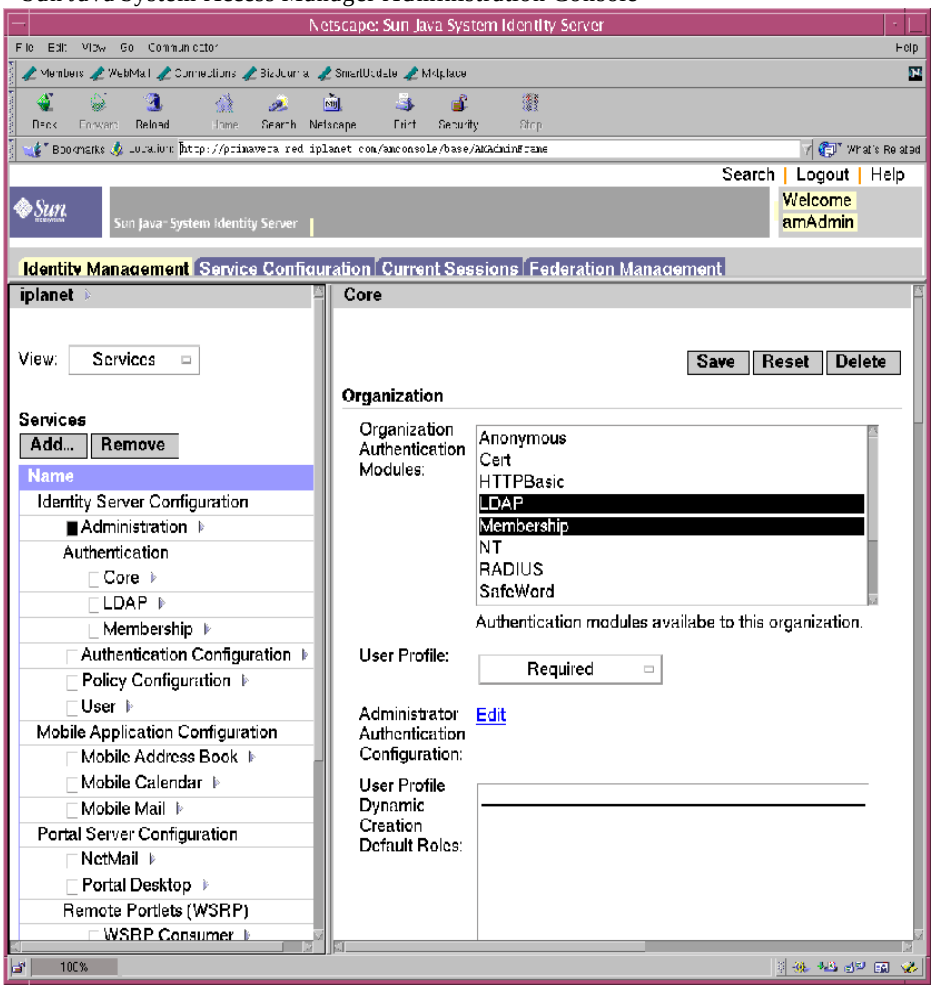
This section provides an overview of administering Portal Server, both from the Access Manager console and the command line.

Using the Access Manager Console

You administer Portal Server and Access Manager services through the HTML-based administration console provided by the Access Manager. Portal Server adds administration modules for Portal Server-specific services to extend the Access Manager console. See the individual chapters in this guide for information on the actual tasks you perform using the console.

The Access Manager console is divided into three sections: the location pane, the navigation pane and the data pane. Using all three panes, the administrator can navigate the directory, perform user and service configurations, and create policies. Figure 1-2 on page 42 shows the administration console.

Figure 1-2 Sun Java System Access Manager Administration Console



Location Pane

The location pane runs along the top of the console. The uppermost View menu allows the administrator to switch between the four different management views:

- Identity Management

- Service Configuration
- Current Session
- Federation Management

The Welcome field displays the name of the user that is currently running the console with a link to their user profile.

The Help link opens a browser window containing an HTML version of Appendixes C, D, E, and F of this documentation, the Attribute Reference Guide.

The Logout link enables the user to log out of the Access Manager console.

Navigation Pane

The navigation pane is the left portion of the console. The Directory Object portion is at the top of the pane and displays the name of the directory object that is currently open and its Properties link. The Show menu lists the directories under the selected directory object. Depending on the number of sub-directories, a paging mechanism is provided.

Data Pane

The data pane is the right portion of the console. Object attributes and their values are displayed and configured here. Entries are selected for their respective group, role or organization in this pane.

Using Command-Line Utilities

The Portal Server command-line interface consists of utilities provided by the Access Manager and Portal Server.

See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for a complete list and syntax of Portal Server command-line utilities. Refer to the Access Manager product documentation for information on its command-line utilities

Part I

Configuring the Sun Java System Portal Server

Chapter 2

Post Installation Configuration

The Portal Server Configurator

When choosing to configure later during installation with the Sun Java™ Enterprise System installer, use the Portal Server configurator to configure the Portal Server component product. The following checklists in this section describe the parameters used to configure the Portal Server component product.

Running the Configurator

To run the configurator:

1. As root in a terminal window, go to the directory that contains the configurator:

   `cd` *PortalServer-base*`/SUNWps/lib`

2. Run the configurator script by typing:

   `./configurator`

NOTE To turn on debugging:

`configurator -DPS_CONFIG_DEBUG=y`

If you turn on debugging, passwords are displayed on the screen as well as the debugging information.

---

3. Follow the instructions on the configuration screens.

Running the Configurator in a Localized Environment

A localized enviroment is one in which the software product has been modified to be compatible with the operating enviroment of a specific region or given language.

To run the configurator in a localized environment on a Solaris Sparc or x86 machine, type the following as one line:

```
/usr/jdk/entsys-j2se/bin/java -DPS_CONFIG_DEBUG=y
-DDEPLOY_ADMIN_PASSWORD=<deploy admin password>
-DDS_DIRMGR_PASSWORD=<LDAP directory manager password>
-DIDSAME_LDAPUSER_PASSWORD=<AM ldap user password>
-DIDSAME_ADMIN_PASSWORD=<AM admin password>
-DBASEDIR=PS-INSTALL-DIR
-cp PS-INSTALL-DIR/SUNWps/lib/configL10N.jar
```

For Secure Remote Access, the command for running the L10N configurator is:

```
/usr/jdk/entsys-j2se/bin/java -DPS_CONFIG_DEBUG=y
-DDEPLOY_ADMIN_PASSWORD=<deploy admin password>
-DDS_DIRMGR_PASSWORD=<LDAP directory manager password>
-DIDSAME_LDAPUSER_PASSWORD=<AM ldap user password>
-DIDSAME_ADMIN_PASSWORD=<AM admin password>
-DBASEDIR=PS-INSTALL-DIR
-cp PS-INSTALL-DIR/SUNWps/lib/configL10N.jar
```

To run the configurator in a localized environment on a Linux machine, type the following as one line:

```
/usr/jdk/entsys-j2se/bin/java -DPS_CONFIG_DEBUG=y
-DDEPLOY_ADMIN_PASSWORD=<deploy admin password>
-DDS_DIRMGR_PASSWORD=<LDAP directory manager password>
-DIDSAME_LDAPUSER_PASSWORD=<AM ldap user password>
-DIDSAME_ADMIN_PASSWORD=<AM admin password>
-DBASEDIR=PS-INSTALL-DIR
```

```
-cp PS-INSTALL-DIR/sun/portal/lib/configL10N.jar
com.sun.portal.config.ConfigureL10N
```

For SRA the command for running L10n Configurator is:

```
/usr/jdk/entsys-j2se/bin/java -DPS_CONFIG_DEBUG=y
-DDEPLOY_ADMIN_PASSWORD=<deploy admin password>
-DDS_DIRMGR_PASSWORD=<LDAP directory manager password>
-DIDSAME_LDAPUSER_PASSWORD=<AM ldap user password>
-DIDSAME_ADMIN_PASSWORD=<AM admin password>
-DBASEDIR=PS-INSTALL-DIR
-cp PS-INSTALL-DIR/sun/portal/lib/configL10N.jar
com.sun.portal.config.ConfigureL10N SRA
```

Configuration Checklists

If you have chosen to configure later during installation, you will need to use the Sun Java ™ System Portal Server configurator to configure your Portal Server installation. The following checklists describe the values that you will need for a post install configuration. Depending on the type of installation you perform, the values that you use might vary.

Portal Server And Secure Remote Access

Table 2-1 is a three column table that lists all the values that you might need when you choose configure later during install. Depending on the type of installation you perform, the values that you use might vary.

Table 2-1 is an example checklist that assumes a Sun Java System Application Server deployment. If you are deploying on Sun Java System Web Server, BEA WebLogic, or IBM WebSphere Application Server, see the section, “Web Container Checklists,” for those web container values.

Table 2-1 Portal Server Installation Checklist

Parameter	Default Value	Description
Installation Directory		

Table 2-1 Portal Server Installation Checklist *(Continued)*

Parameter	Default Value	Description
Directory to install Sun Java System configurator components	`/opt`	This is the base directory in which the Portal Server software is installed.
Deployment Information		
Deployment Type	Sun Java System Application Server	The Portal Server can be deployed on the Sun Java System Web Server, Sun Java System Application Server, BEA WebLogic Server, or IBM WebSphere Application Server.
Deployment URI	`/portal`	The URI is the space on the web server or application server that the Portal Server uses. The value for the deployment URI must have a leading slash and must contain only one slash. However, the deployment URI cannot be a "/" by itself.
Load balancer controlling Portal Server Instances	Unselected	Check this box if you will be using a load balancer with your Portal Server.
Load Balancer URL	`http://`*fully_qualified_domain*`:`*port*`/portal`	
Web Container Information (Sun Java System Application Server)		
Installed Directory	`/opt/SUNWappserver/appserver`	Directory in which the Sun Java System Application Server is installed.
Domain	`/var/opt/SUNWappserver7/domains/domain1`	The Sun Java System Application Server domain contains a set of instances. The domain specified will contain the instance used by the Portal Server. This domain must already be configured.
Instance	server	The name of the Sun Java System Application Server instance to which the Portal Server will be deployed. This instance must already be configured. The instance name should not contain spaces.
Instance Port	8080	The port on which the Sun Java System Application Server instance will run.
Document Root Directory	`/var/opt/SUNWappserver/domains/domain1/docroot`	The directory where static pages are kept.
Administrator	admin	The administrator user ID.
Administration Port	4849	The port number of the administration server.
Administration Password		The administration server password.
Access Manager Information		

Table 2-1 Portal Server Installation Checklist *(Continued)*

Parameter	Default Value	Description
Installed Base Directory	`/opt`	This is the base directory in which the Sun Java System Access Manager software is installed.
Internal LDAP Authentication User Password		The Internal LDAP Authentication User Password chosen during the Sun Java System Access Manager installation.
Administrator (amadmin) Password		The top level administrator (amadmin) password chosen during the Sun Java System Access Manager software installation.
Directory Manager DN	cn=Directory Manager	The LDAP directory manager distinguished name (DN).
Directory Manager Password		The directory manager password chosen during the installation of the Sun Java System Directory Server.
Secure Remote Access Information (for configuring Secure Remote Access Support)		
Gateway Protocol	https	The Protocol used by the gateway. The gateway will communicate using Secure Sockets Layer (SSL).
Portal Server Domain	*portal-server-domain-name*	The domain name for the machine on which the Portal Server is installed.
Gateway Domain	*gateway-domain-name*	The domain name of the gateway machine.
Gateway Port	443	The port on which the gateway listens.
Gateway Profile Name	default	A gateway profile contains all the information related to gateway configuration, such as the port on which gateway listens, SSL options, and proxy options. You can create multiple profiles in the gateway administration console and associate different instances of gateway with different profiles. See “Creating a Gateway Profile” in the *Sun Java System Portal Server, Secure Remote Access 6 2004Q2 Administrator’s Guide*.
Gateway Loggging User Password		This allows administrators with non-root access to look at gateway log files.
Confirm Password		Retype to verify password.

Gateway

Table 2-2 Gateway Installation Checklist

Parameter	Default Value	Description
Protocol	https	The protocol used by the gateway. The gateway will usually communicate using Secure Sockets Layer (SSL).
Host Name	*host*	The host name of the machine on which the gateway is installed.
Subdomain	*gateway-subdomain-name*	The subdomain name of the gateway machine.
Domain	*gateway-domain-name*	The domain name of the gateway machine.
IP Address	*host-ip-address*	The IP Address should be that of the machine where Gateway is installed and not that of the Sun Java System Access Manager.
Access Port	443	The port on which the gateway machine listens.
Gateway Profile Name	default	Specify the same profile name specified when you installed Portal Server or Secure Remote Access support. See “Creating a Gateway Profile” in the *Sun java System Portal Server, Secure Remote Access 6 2004Q2 Administrator’s Guide* for more information
Log User Password		This allows administrators with non-root access to look at gateway log files.
Start the gateway after installation	Selected	The gateway can be started automatically (if this option is checked) or it can be started later. To start the gateway manually use the following command located in *PortalServer-base*/SUNWps/bin: `./gateway -n` *gateway-profile-name* `start`
Certificate Information		
Organization	MyOrganization	The name of your organization.
Division	MyDivision	The name of your division.
City or Locality	MyCity	The name of your city or locality
State or Province	MyState	The name of your state
Two-Letter Country Code	us	The two letter country code for your country.
Certificate Database Password		This can be any password you choose.
Retype Password		Retype the password to verify.

Netlet Proxy

Table 2-3 Netlet Proxy Installation Checklist

Parameter	Default Value	Description
Host Name	*hostname*	The host name of the machine on which Netlet Proxy is installed.
Subdomain	*localhost-subdomain-name*	The sub-domain name of the machine on which the Netlet Proxy is installed.
Domain	*localhost- domain-name*	The domain name of the machine on which the Netlet Proxy is installed.
IP Address	*host-ip-address*	The IP address should be that of the machine where Netlet Proxy is installed and not that of Sun Java System Access Manager.
Access Port	10555	The port on which the Netlet Proxy listens.
Gateway Profile Name	default	Specify the same profile name specified when you installed Portal Server or Secure Remote Access support. See “Creating a Gateway Profile” in the *Sun java System Portal Server, Secure Remote Access 6 2004Q2 Administrator’s Guide* for more information.
Gateway Logging User Password		This allows administrators with non-root access to look at gateway log files.
Start Netlet Proxy after installation	Selected	The Netlet Proxy can be started automatically (if this option is checked) or it can be started later. To start the Netlet Proxy manually use the following command located in *NetletProxy-base*/SUNWps/bin `./netletd -n default start`
Certificate Information		
Organization	MyOrganization	The name of your organization.
Division	MyDivision	The name of your division.
City or Locality	MyCity	The name of your city or locality.
State or Province	MyState	The name of your state or province.
Two-letter Country Code	us	The two-letter country code for your country.
Certificate Database Password		This can be any password you choose.
Retype Password		Retype the password to verify.

Rewriter Proxy

Table 2-4 Rewriter Proxy Installation Checklist

Parameter	Default Value	Description
Host Name	*hostname*	The host name of the machine on which the Rewriter Proxy is installed.
Subdomain	l*ocalhost-subdomain-name*	The sub-domain name of the machine on which the Rewriter Proxy is installed.
Domain	*localhost- domain-name*	The domain name of the machine on which the Rewriter Proxy is installed.
IP Address	*host-ip-address*	The IP address should be that of the machine on which Rewriter Proxy is installed and not that of Sun Java System Access Manager.
Access Port	10443	The port on which the Rewriter Proxy listens.
Gateway Profile Name	default	Specify the same profile name specified when you installed Portal Server or Secure Remote Access support. See “Creating a Gateway Profile” in the *Sun java System Portal Server, Secure Remote Access 6 2004Q2 Administrator’s Guide* for more information.
Gateway Logging User Password		This allows administrators with non-root access to look at gateway log files.
Start the Rewriter Proxy after installation	Selected	The Rewriter Proxy can be started automatically (if this option is checked) or it can be started manually later. To start the Rewriter Proxy manually use the following command located in *RewriterProxy-base*/SUNWps/bin `./rwproxyd -n default start`
Certificate Information		
Organization	MyOrganization	The name of your organization.
Division	MyDivision	The name of your division.
City or Locality	MyCity	The name of your city or locality.
State or Province	MyState	The name of your state or province.
Two-letter Country Code	us	The two-letter country code for your country.
Certificate Database Password		This can be any password you choose.
Confirm Password		Retype the password to verify.

Table 2-5 Netlet and Rewriter Proxy Information

Parameter	Default Value	Description
Work With Portal Server on this Node	Selected	Uncheck this box if Portal server and Sun Java System Secure Remote Access componets are being Installed on seperate machines. Check this box if portal server and other Sun Java System Secure Remote Access components are installed in the same machine. When this box is checked, the following fields are enabled.
Portal Server Protocol	http	Protocol that Portal Server uses to communicate.
Portal Server Host	*fully-qualified-host-name*	Fully qualified host name of Portal Server.
Portal Server Port	80	
Portal Server Deployment URI	/portal	
Organization DN		
Access Manager Service URI	/amserver	
Access Manager password encryption key		The value of the encryption key. The encryption key is located in: `/etc/opt/SUNWam/config AMConfig.properties` as the parameter am.encryption.pwd.

Web Container Checklists

The Portal Server installation has dependencies on some web container parameters. The following checklists describe the parameters that will be needed during the Portal Server installation process. See the checklist that pertains to the web container on which you are deploying the Portal Server product.

- Sun Java System Web Server Checklist
- Sun Java System Application Server Checklist

- BEA WebLogic Server Checklist
- IBM WebSphere Application Server Checklist

For more information about using the supported application servers with the Portal Server, see the appendix in this guide that pertains to your application server deployment.

Sun Java System Web Server Checklist

Table 2-6 Sun Java System Web Server Values Used During Portal Server Installation

Parameter	Default Value	Description
Installed Directory	`/opt/SUNWwbsvr`	The base directory in which the Sun Java System Web Server is installed.
Instance	*host*	The web server instance you want the Portal Server to use. The instance name should not contain spaces.
Instance Port	80	The port for accessing Portal Server.
Secure web container instance port	Unchecked	Check this box if SSL will be running on the instance port.
Document Root Directory	`/opt/SUNWwbsvr/docs`	The directory where static pages are kept.

Sun Java System Application Server Checklist

Table 2-7 Sun Java System Application Server Values Used During Portal Server Installation

Parameter	Default Value	Description
Installed Directory	`/opt/SUNWappserver/appserver`	Directory in which the Sun Java System Application Server is installed.
Domain	`/var/opt/SUNWappserver/domains/domain1`	The Sun Java System Application Server domain contains a set of instances. The domain specified will contain the instance used by the Portal Server. This domain must already be configured.
Instance	server	The name of the Sun Java System Application Server instance to which the Portal Server will be deployed. This instance must already be configured. The instance name should not contain spaces.

Table 2-7 Sun Java System Application Server Values Used During Portal Server Installation *(Continued)*

Parameter	Default Value	Description
Instance Port	8080	The port used to access Portal Server
Secure web container instance port	Unchecked	Check this box if SSL will be running on the instance port.
Document Root Directory	`/var/opt/SUNWappserver/domains/domain1/docroot`	The directory where static pages are kept.
Administrator	admin	The administrator user ID.
Administration Port	4849	The port number of the administration server.
Administration Password		The administration server password.

BEA WebLogic Server Checklist

Table 2-8 BEA WebLogic Server Values Used During Portal Server Installation

Parameter	Default Value	Description
BEA Home Directory	/user/local/bea	The directory in which BEA is installed.
Product Installation Directory	`/usr/local/bea/weblogic81`	The directory in which the BEA WebLogic Server software is installed.
User Project's Directory	user_projects	Use the value you entered during BEA WebLogic installation.
Domain	mydomain	The BEA WebLogic Server domain contains a set of instances. The domain specified will contain the instance used by the Portal Server. This domain must already be configured.
Instance	myserver	The name of the BEA WebLogic Server instance to which the Portal Server will be deployed. This instance must already be configured. The name must not contain a space. If you are installing Portal Server on an administration server instance this will be the name of the administration server instance. Otherwise it will be the name of the managed server instance.
Instance Port	7001	The port for accessing Portal Server
Secure instance port	Unselected	Check this box if SSL will be running on the instance port.

Table 2-8 BEA WebLogic Server Values Used During Portal Server Installation *(Continued)*

Parameter	Default Value	Description
Document Root Directory	`/usr/local/bea/user_projects/domains/mydomain/applications`	The document root value of DefaultWebApp needs to be deployed to the BEA WebLogic Server instance you are running the Portal Server software on. DefaultWebApp is the default web application, from which is served static content in a BEA WebLogic Server. By default it is only deployed to the domain (mydomain) and the server instance defined or created during the BEA WebLogic Server install. This means that if you create your own BEA WebLogic Server or domain, you need to deploy the DefaultWebApp to it, either by copying the directory to the new server's deployment directory, or by using the BEA WebLogic Server administration console. See the BEA WebLogic Server documentation for more detail on how to configure a default web application.
Administrator	system	The administrator's user ID.
Administration Protocol	http	Protocol on which the administration server of BEA WebLogic Server runs on.
Administration Port	7001	Port on which the administration server of BEA WebLogic Server is running. If the Portal Server is installed on the BEA WebLogic Server administration server itself, the port on which Portal Server runs and the administration port of BEA WebLogic Server will be the same.
Administration Password		The system password.

IBM WebSphere Application Server Checklist

Table 2-9 IBM WebSphere Application Server Values Used During Portal Server Installation

Parameter	Default Value	Description
Installed Directory	`/opt/WebSphere/Express51/AppServer`	The directory in which the IBM WebSphere Application Server software is installed.
Virtual Host	default_host	Use the value you entered during IBM WebSphere installation.
Node	*machine-name*	

Table 2-9 IBM WebSphere Application Server Values Used During Portal Server Installation

Parameter	Default Value	Description
Instance	server1	The name of the instance to which the Portal Server will be deployed. This instance must already be configured. Portal Server cannot be installed into an application server instance or domain whose name contains a dash or a space, for example, Default-Server or Default Server.
Document Root Directory	`/opt/IBMHTTPD/htdocs/en_US`	The directory where static pages are kept. This directory is created during the Sun Java System Access Manager installation.

Portal Server Post-Installation Tasks

Post-installation tasks need to be performed for each of the following components:

- Portal Server
- Secure Remote Access
- Gateway
- Netlet and Rewriter Proxy

Portal Server

To access the Portal Server or the Access Manager administration console the directory server and the web container must first be started.

Use the following command to start a local installation of the directory server:

`/var/opt/mps/serverroot/slapd-`*hostname*`/start-slapd`

The following post-installation tasks depend on the type of web container on which you deployed the Portal Server.

- Sun Java System Web Server
- Sun Java System Application Server
- BEA WebLogic Server
- IBM WebSphere Application Server

Sun Java System Web Server

To start the Sun Java System Web Server:

1. Start the admin instance. In a terminal window type:

   `cd` *WebServer-base*`/SUNWwbsrv/https-`*server-instance-name*

   `./start`

2. Access the Sun Java System Web Server administration console.
3. Click Apply Changes to restart the web container.

Sun Java System Application Server

Configuring the Application Server Instance

1. Start the domain. In a terminal window, type:

   ```
   cd $AS8.1_INSTALLDIR/appserver/bin
   ```

   As one line, type:

   ```
   ./asadmin start-domain --user admin --passwordfile passwordfilename
   domain1
   ```

   Starting the domain will start admin server as well as all the other instances.

2. For starting a particular instance, in the terminal window, type:

   ```
   cd $AS8.1_INSTALLDIR/appserver/bin
   ```

   As one line, type:

   ```
   ./asadmin start-instance --user admin_user [--passwordfile filename]
   [--host host_name] [--port port_number] instance_name
   ```

3. For stopping a particular instance, in the terminal window, type:

   ```
   cd $AS8.1_INSTALLDIR/appserver/bin
   ```

   As one line, type:

   ```
   ./asadmin stop-instance --user admin_user [--passwordfile filename]
   [--host host_name] [--port port_number] instance_name
   ```

Stopping and Starting the Sun Java System Application Server

Start the Sun Java System Application Server instance.

In a terminal window, change directories to the application server's instances utilities directory and run the `startserv` script. The following example assumes that the default application server domain and instance have been used.

```
cd /var/opt/SUNWappserver7/domains/domain1/server1/bin
./startserv
```

To stop and start the Sun Java System Application Server using the `asadmin` utility or from the Sun Java System Application Server administration console, consult the Sun Java System Application Server documentation.

BEA WebLogic Server

When deploying the Portal Server on BEA WebLogic Server, perform the following steps following the installation of the Portal Server software.

1. Check the `/var/sadm/install/logs/Java_Enterprise_System_install.B/MMddhhmm` file for errors.

   MM = month

   dd = day

   hh = hour

   mm = minute

2. Stop all BEA WebLogic Server instances (the admin and managed servers).
3. Start the BEA WebLogic admin server instance. If you have installed on a managed instance, start the managed instance too.)
4. From the command line, execute the following:

   PortalServer-base`/SUNWps/bin/deploy`

   Choose the default for the deploy URI and server instance name, and enter the BEA WebLogic Server admin password when prompted.

5. Execute the following commands:

   a. setenv DEPLOY_ADMIN_PASSWORD *BEA-WebLogic-admin-password*

   b. setenv IDSAME_ADMIN_PASSWORD *Identity-Server-admin-password*

a. *PortalServer-base*/SUNWps/lib/postinstall_PortletSamples

Enter the BEA WebLogic Server admin password and the Access Manager admin password when prompted.

This deploys the portletsamples.war file.

6. Restart the BEA WebLogic Server instance into which Portal Server was deployed.See your web container documentation for instructions on starting the web container instance.

NOTE In the case of a managed server installation, the .war files do not get deployed. The .war files should be deployed using the BEA WebLogic Server administration console.

If you will be supporting multiple authentication methods, for example, LDAP, UNIX, Anonymous, you must add each authentication type to the Core authentication service to create an authentication menu. See the *Sun Java System Portal Server 6 2004Q2 Administrator's Guide* for further information.

IBM WebSphere Application Server

1. Check the /var/sadm/install/logs/Java_Enterprise_System_install.B/MMddhhmm file for errors.

2. Stop and restart the application server instance and the application server node. See your web container documentation for instructions on starting the web container instance.

Secure Remote Access

When using the Portal Server with the gateway, the gateway Certificate Authority (CA) certificate must be added to the Portal Server trusted CA list, regardless of whether the Portal Server is running in HTTP or HTTPs mode.

When a user session time out or user session logout action happens, the Sun Java System Access Manager sends a session notification to the gateway. Even when the Sun Java System Access Manager is running in HTTP mode, it will act as an SSL client using HttpsURLConnection to send the notification. Since it is connecting to an SSL server (the gateway), it should have the gateway CA certificate as part of the Trusted CA list or it should have an option to allow self signed certificate.

NOTE The method for adding the CA to the trusted CA list depends on the protocol handler defined.

To create HttpsURLConnection, the Java Virtual Machine (JVM™) property `-Djava.protocol.handler.pkgs` needs to be set.

If Portal Server is running on the Sun Java System Web Server, Sun Java System Application Server, or BEA WebLogic Server, this property is correctly set to `com.iplanet.services.comm` by default. The Sun Java System Access Manager package has the implementation of HttpsURLConnection and it provides an option to accept self-signed certificates from any SSL server by adding the flag `com.iplanet.am.jssproxy.trustAllServerCerts=true` in the `AMConfig.properties file`.

The `-Djava.protocol.handler.pkgs` is not set by default for the IBM WebSphere Application Server. The HttpsURLConnection implementation for supported application servers must use their own default handler (this could be JSSE or custom SSL implementation).

Gateway

1. Start the gateway using the following command:

   Gateway-base`/SUNWps/bin/gateway -n` *new-profile-name* `start`

   `default` is the default name of the gateway profile that is created during installation. You can create your own profiles later, and restart the gateway with the new profile. See Creating a Gateway Profile in Chapter 2 of the *Sun Java System Portal Server, Secure Remote Access 6 2005Q1 Administrator's Guide*.

   If you have multiple gateway instances, use:

   Gateway-base`/SUNWps/bin/gateway start`

NOTE This step is not required if you chose **y** for the Start Gateway after installation option during the gateway installation.

CAUTION Ensure that only the configuration files for the instances that you want to start are in the `/etc/opt/SUNWps` directory.

If you want to stop all the gateway instances that are running on that particular node, use the following command:

Gateway-base/SUNWps/bin/gateway stop

The Netlet and the gateway need Rhino JavaScript™ parser (bundled as SUNWrhino) for PAC file support. This must be installed in the Gateway and Portal Server node. To install, use the following steps:

1. Add this package by running **pkgadd -d . SUNWrhino** from the current directory.
2. Copy package-base/js.jar to ${JAVA_HOME}/jre/lib/ext directory.

Netlet and Rewriter Proxy

Before starting the Netlet Proxy and the Rewriter Proxy, ensure that the gateway profile is updated with the Netlet Proxy and the Rewriter Proxy options.

- If you did not choose the option to start the Netlet Proxy during installation, you can start the Netlet Proxy manually. In the directory, *Portal-proxy-base*/SUNWps/bin, type:

  ```
  ./netletd -n default start
  ```

- If you did not choose the option to start the Rewriter Proxy manually during installation, you can start it manually. In the directory *Portal-proxy-base*/SUNWps/bin, type:

  ```
  ./rwproxyd -n default start
  ```

NOTE Ensure that you enable the Access List service for all users, to allow access through the gateway.

The Sun Java System Portal Server software NetFile needs JCiFS libraries (bundled as SUNWjcifs) for Windows access. This needs to be installed in Portal Server node only. To install, use the following steps.

1. Add this package by running pkgadd -d . SUNWjcifs from the current (this) directory.
2. Run *PortalServer-base*/SUNWps/lib/postinstall_JCIFS
3. Run *PortalServer-base*/SUNWps/bin/undeploy

4. Run *PortalServer-base*/SUNWps/bin/deploy command.
5. Restart the server.

Verifying the Portal Server Installation

Accessing the Portal Server Administration Console and Desktop

To Access the Sun Java System Access Manager Administration Console

1. Open a browser.
2. Type *protocol*://*hostname.domain*:*port*/amconsole

   For example,

   http://example.com:80/amconsole
3. Enter the administrator's name and password to view the administration console.

   This is the name and password you specified at the time of installing the Sun Java System Access Manager software.

To Access the Portal Server Desktop

Verify the Portal Server installation by accessing the Desktop. Use the following URL to access the Desktop:

protocol://*fully-qualified-hostname:port/portal-URI*

For example,

http://example.com:80/portal

When you access the Desktop, the Authless Desktop is displayed. This allows users accessing the Desktop URL to be authenticated automatically and granted access to the Desktop.

If the sample Portal Desktop displays without any exception, then your Portal Server installation is good.

Verifying the Gateway Installation

1. Run the following command to check if the gateway is running on the specified port:

   `netstat -an | grep` *port-number*

   where the default gateway port is 443.

   If the gateway is not running, start the gateway in the debug mode, and view messages that are printed on the console. Use the following command to start the gateway in debug mode:

   PortalServer-base`/SUNWps/bin/gateway -n` *profilename* `start debug`

   Also view the log files after setting the gateway.debug attribute in the `platform.conf.profilename` file to message. See the section Understanding the `platform.conf` File in Chapter 2, “Administering Gateway” in the *Sun Java System Portal Server, Secure Remote Access 6 2005Q1 Administrator’s Guide*, for details.

2. Run the Portal Server in secure mode by typing the gateway URL in your browser:

   `https://`*gateway-machine-name*`:`*portnumber*

   If you have chosen the default port (443) during installation, you need not specify the port number.

3. Login to the Access Manager administration console as administrator using the user name `amadmin`, and using the password specified during installation.

You can now create new organizations, roles, and users and assign required services and attributes in the administration console.

Chapter 3

Creating and Deleting Instances of the Server

An instance is a server that listens on a particular port, bound to either one or more IP addresses. For the Sun Java™ System Portal Server, an instance corresponds to a web server process listening on a port and running a single Java™ Virtual Machine (JVM™).

NOTE Multiple-instances are supported with Sun Java™ System Web Server and Sun Java™ System Application Server.

To Create an Instance of the Server

1. Log in to the server running the Portal Server.
2. Deploy a new Access Manager instance. For instructions on deploying a new Access Manager instance, see the Chapter 1 of the *Sun Java System Access Manager Administration Guide* at: http://docs.sun.com/.
3. Go to the Portal Server utilities directory.

   `cd` *PortalServer-base*`/SUNWps/bin`
4. Run the `multiserverinstance` script.

   `./multiserverinstance`
5. Enter the name of the instance.
6. Enter the port of the new instance.
7. If you have portlets, redeploy them. For instructions to redeploy portlets, consult the *Sun Java System Portal Server 2005Q1 Technical Reference Guide*.

8. After the `multiservinstance` script exits, go to the web server instance directory.

   `cd` *WebServer-base*`/https-`*new-instance-name*

9. Stop the web server instance.

   `./stop`

10. Restart the web server instance.

    `./start`

11. Go to the newly created instance in a browser.

12. Repeat steps Step 9 through Step 11 for each newly created instance.

13. In a browser, enter:

    - `http://`*hostname.domain*`:`*instance-portnumber*`/amconsole` to access the administration console through the new instance
    - `http://`*hostname.domain*`:`*instance-portnumber*`/portal` to access the default URL for the Portal Desktop through the new instance

If you create any additional server instances and you want to run them as non-root or nobody, comment out the following lines for each instance at *AccessManager-base*`/SUNWam/bin/amserver.`*instance-nickname*

```
if [ "$uid" != "0" ];
then
    echo "‘$gettext ’You must be root user to run’‘ $0."
    exit 1
fi
```

To Delete an Instance of the Server

1. Log in to the server running the Portal Server.

2. Change directories to *PortalServer-base*`/SUNWps/bin`.

   `cd` *PortalServer-base*`/SUNWps/bin`

3. If you have portlets, remove them. For instructions, see the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for more information on administering command line utilities.

4. Enter:

   `./multiserverinstance delete -instance` *instance-name*

5. If you are also removing the Sun Java System Access Manager, uninstall the Access Manager instance. For instructions on uninstalling an Access Manager instance, see the Chapter 1 of the *Sun Java System Access Manager Administration Guide* at: `http://docs.sun.com/`.

Setting Up the Portal Server to Use Secure External LDAP Directory Server

In the default install, the Sun Java™ System Portal Server, the Sun Java™ System Access Manager, and the Sun Java™ System Directory Server software are all running on the same host. However, depending on the performance, security, and integration requirements of your deployment, you might want to run the directory server on a separate, external host and have the Portal Server access the directory over a secure connection using Secure Sockets Layer (SSL). In order to access the Directory Server over a secure connection, the Sun Java™ System Web Server or Sun Java™ System Application Server must be configured to trust the certificate authority that signed the directory's certificate.

Setting up the Sun Java System Portal Server to use an external LDAP directory, requires the following procedures:

- Installing the Portal Server.
- Configuring the Directory Server to run SSL.
- Creating a certificate database.
- Installing a root Certificate Authority (CA) certificate.
- Enabling SSL for the Directory Server.

To Configure the Directory Server to Run in SSL

1. Verify that both the Directory Server (`ns-slapd` process) and the administration server (`ns-httpd` process) are started and running.

2. As root, in a terminal window start the directory server console by typing:

   ```
   /var/opt/mps/serverroot/startconsole
   ```

3. In the login window that is displayed, enter admin as the user name and the passphrase for the Directory Server.

4. In the left pane of the console, expand the directory until you see the Directory Server instance under Server Group.

5. Select Directory Server instance and click Open.

6. Select Tasks and then Manage Certificates.

   The first time you perform this task, you'll be asked to create a certificate database by entering a password. Make a note of this password as you will need it later to start up the Directory Server.

7. Click Request.

   The Certificate Request Wizard appears. Follow the wizard and complete the steps to generate a certificate request. The request is sent to a Certificate Management Server (CMS) for approval. The CMS returns the real certificate. Save a copy of the certificate request by copying the request data to a file.

8. After the certificate request is sent to the CMS, have the administrator of the CMS approve the request and send back the approved certificate.

9. Get the generated certificate for the DS and the CMS certificate.

   Since the CMS generated the certificate for DS, the CMS will also have to be trusted by importing its certificate as a root CA.

10. Select Manage Certificates, Server Certificates and then click Install.

    The Certificate Install Wizard appears.

11. Copy and paste the approved certificate data from Step 8 into the text area and follow the steps of the wizard to install the certificate.

    When the certificate is successfully installed, the certificate displays as a line item on the Server Certificates tab.

12. With the Manage Certificates window open, select the CA Certificates tab.

    If the CA from which you got your certificate in Step 9 is in the CA certs list, you do not need to install the certificate in that list.

    If the certificate is not in the list, you need to obtain the root CA certificate from your certificate authority and install it.

a. Click Install.

b. Copy and paste the CMS certificate data into the text area and follow the steps of the wizard to install the certificate.

   The certificate name should appear in the CA certs list.

13. Click Close to close the Manage Certificates window.

14. Select the Configuration tab.

15. Click the Encryption tab, check the Enable SSL for this server and Use the cipher family: RSA check boxes and click Save.

16. On the Network tab verify or specify a valid port number in the Encrypted port field on the and click Save.

    The default port is 636.

17. Restart the Directory Server and supply the certificate database password entered in Step 6.

    Your Directory is now listening on port 636 (default) for SSL connections.

To Create a Trust Database

When you create the trust database, you specify a password that will be used for a key-pair file. You will also need this password to start a server using encrypted communications.

In the certificate database you create and store the public and private keys, referred to as your key-pair file. The key-pair file is used for SSL encryption. You will use the key-pair file when you request and install your server certificate. The certificate is stored in the certificate database after installation.

The procedure for creating a certificate database depends on the type of web container that you are using. The following instructions are for creating a certificate database on the Sun Java System Application Server and can also be found in *Sun Java System Application Server Administration Guide to Security* on `http://docs.sun.com`.

Instructions on creating a certificate database on the Sun Java System Web Server can be found in *Sun Java system Web Server, Enterprise Edition Administration Guide* at `http://docs.sun.com`.

For instructions on To create a certificate database on the Sun Java System Application Server, perform the following steps in the administration interface:

1. Make sure the Application Server instance is started.
2. Access App Server Instances and select the server instance.
3. Access Security.
4. Click Manage Database.
5. Click the Create Database link.

   The Initialize Trust Database page is displayed.
6. Enter a password for the database.
7. Repeat the password.
8. Click OK.
9. Access App Server Instances and your server instance in the left pane, then click Apply Changes.
10. Stop and restart the server for changes to take effect.

To Use the password.conf File

If you want an SSL/TLS-enabled Sun Java System Application Server to be able to restart unattended when configured for SSL, you can save the trust database password in a `password.conf` file.

NOTE Be sure that your system is adequately protected so that this file and the key databases are not compromised.

Further information on the password.conf file can be found in Using the `password.conf` File, in the *Sun Java System Application Server Administrator s Configuration File Reference*.

Normally, you cannot start a Unix SSL-enabled server with the `/etc/rc.local` or the `/etc/inittab` files because the server requires a password before starting. Although you can start an SSL-enabled server automatically if you keep the password in plain text in a file, this is not recommended. The server's `password.conf` file should be owned by root or the user who installed the server, with only the owner having read and write access to them. On Unix, leaving the SSL-enabled server's password in the password.conf file is a large security risk. Anyone who can access the file has access to the SSL-enabled server's password. Consider the security risks before keeping the SSL-enabled server's password in the `password.conf` file.

To Install A Root Certificate Authority (CA) Certificate

The procedure for installing a root CA certificate depends on the type of web container that you are using.

The following procedure describes how to install a root CA on the Sun Java System Application Server and can also be found in *Sun Java System Application Server Administration Guide to Security* on http://docs.sun.com.

Instruction for installing a Root CA on the Sun Java System Web Server can be found in *Sun Java System Web Server, Enterprise Edition Administration Guide* at http://docs.sun.com.

The source that provided your certificate is the same source from which you obtain your root CA certificate.

To install a certificate from a CA, perform the following steps in the Administration interface:

1. Access App Server Instances and select the server instance in the left pane.
2. Access Security.
3. Select Certificate Management.
4. Click the Install link.

   The Install a Server Certificate is displayed.
5. Select Trusted Certificate Authority (CA) for a certificate of a CA that you want to accept as a trusted CA for client authentication
6. Select the cryptographic module from the drop-down list.
7. Enter the password for your key-pair file.
8. Leave the name for the certificate field blank if it will be the only one used for this server instance, unless:

   - Multiple certificates will be used for virtual servers. In this case, enter a certificate name unique within the server instance.

- Cryptographic modules other than internal are used. In this case, enter a certificate name unique across all server instances within a single cryptographic module.

If a name is entered, it will be displayed in the Manage Certificates list, and should be descriptive. For example, United States Postal Service CA is the name of a CA, while VeriSign Class 2 Primary CA describes both a CA and the type of certificate.

NOTE When no certificate name is entered, the default value is applied.

9. Select one:
   - Message is in this file. In this case, enter the full pathname to the saved email.
   - Message text (with headers). In this case, paste the email text. If you copy and paste the text, be sure to include the headers Begin Certificate and End Certificate, including the beginning and ending hyphens.
10. Click OK.
11. Select Add Certificate to install a new certificate.
12. Access App Server Instances and your server instance in the left pane, then click Apply Changes.
13. Stop and restart the server for changes to take effect. The certificate is stored in the server s certificate database. The file name will be `cert8.db`.

To Enable Access Manager to use SSL to Communicate with the Directory Server

To enable SSL for the Directory server, edit the `/etc/opt/SUNWam/config/AMConfig.properties` file. This step is container independent and must be done for Sun Java System Web Server as well as Sun Java System Application Server.

Change the following settings in the `AMConfig.properties` file from:

```
com.iplanet.am.directory.ssl.enabled=false
com.iplanet.am.directory.host=server12.example.com (if it needs to be changed)
com.iplanet.am.directory.port=389
```

to

```
com.iplanet.am.directory.ssl.enabled=true
com.iplanet.am.directory.host=server1.example.com
com.iplanet.am.directory.port=636 (port on which DS uses encryption)
```

Change the connection port and the connection type values in the *AccessManager-base*/SUNWam/config/ums/serverconfig.xml file to change from open mode to SSL.

Edit the serverconfig.XML file and change the following line from:

```
    <Server name="Server1" host="gimli.example.com"
port="389"
 type="SIMPLE" />
```

to:

```
to
<Server name="Server1" host="gimli.example.com"
port="636"
type="SSL" />
```

After making these changes to the serverconfig.xml file restart the web container.

Configuring the Portal Server to Run as User Non-Root

The following optional, post-install procedure describes the steps to configure a Sun Java™ System Portal Server installation that is running as root user to run as a non-root user.

NOTE The Sun Java™ System Enterprise Installer does not support installation of the Sun Java™ System Web Server or the Sun Java™ System Application Server as non root. However it does support the installation of the Sun Java™ System Directory Server as non root. This procedure assumes that the web container and the Sun Java™ System Directory Server are running as non-root user.

Perform all steps as superuser, except as noted. After installing the Sun Java™ System Portal Server software, use the following procedure to configure the Portal Server to run as user non-root.

1. Change the ownership of the following directories from root to *Userid:UserGroup*. That is, enter:
   - `chown -R` *Userid*`:`*UserGroup* `/opt/SUNWps`
   - `chown -R` *Userid*`:`*UserGroup* `/etc/opt/SUNWps`
   - `chown -R` *Userid*`:`*UserGroup* `/var/opt/SUNWps`
   - `chown -R` *Userid*`:`*UserGroup* `/opt/SUNWam`
   - `chown -R` *Userid*`:`*UserGroup* `/etc/opt/SUNWam`
   - `chown -R` *Userid*`:`*UserGroup* `/var/opt/SUNWam`

- `chown -R` *Userid*`:`*UserGroup* *WEBCONTAINER-DIR*

If you did not use the Java Enterprise System installer to install the Sun Java System Identity Server as non-root, consult the Access Manager documentation for information on changing the Access Manager directories.

2. Set the following permissions for the Portal Server directories:
   - `chmod 0755 /opt/SUNWps`
   - `chmod 0755 /etc/opt/SUNWps`
   - `chmod 0755 /var/opt/SUNWps`
   - `chmod 0755 /opt/SUNWam`
   - `chmod 0755 /etc/opt/SUNWam`
   - `chmod 0755 /var/opt/SUNWam`
   - `chmod 0755` *WEBCONTAINER-DIR*
3. Restart the directory server as the non-root user.

   The Java Enterprise System installer installs the Java™ Development Kit (JDK™) in `/usr/jdk/entsys`. Change the ownership and permissions of this directory for the non-root user:
   - `chown -R Userid:UserGroup /usr/jdk`
   - `chmod 0755 /usr/jdk`
4. Stop the web container and Directory Server.
5. Ensure that all of the processes are stopped.

   To verify, type:
   - `ps -aef | grep slapd`
   - `ps -aef | grep httpd`
   - `ps -aef | grep http`
   - `ps -aef | grep admin`
6. Kill off any processes that did not get shutdown.
7. Start Directory Server and the web container.
8. Watch the owner of the directory and web container process. It should be running as non-root user.

NOTE If you are running the Portal Server as non-root user, and you want to apply a patch, the ownership of the Portal Server directories must first be changed from non-root back to superuser (root). After the patch has been successfully applied, you can configure ownership and permissions to run Portal Server as non-root user again.

Part II

Administering the Sun Java System Portal Server

Chapter 17, “Configuring the Communication Channels”

Chapter 6

Administering Authentication, Users, and Services

This chapter describes how to use Sun Java™ System Access Manager to administer authentication, users, and services. This chapter does not attempt to explain all aspects of Access Manager. Instead, it focuses on those aspects that pertain to Sun Java™ System Portal Server. See the Access Manager documentation for more information.

This chapter contains these sections:

Overview of Sun Java System Access Manager

In Sun Java System Portal Server (formerly Sun™ ONE Portal Server) implementations, you administer authentication methods, create domains, roles and users, and manage other data, such as profile attributes and logs, through the product itself. You also use the iPlanet Portal Server 3.0 APIs to develop custom applications.

Now, with Portal Server 6 product, you use Access Manager administrative capabilities and APIs formerly found within iPlanet Portal Server 3.0 itself. Access Manager is a set of tools that leverage the management and security potential of Sun Java™ System Directory Server. The goal of Access Manager is to provide an interface for managing user objects, policies, and services for organizations using the Sun Java System Directory Server.

Access Manager enables:

- Sun Java System Directory Server to perform user authentication and single sign-on, increasing data security.
- Administrators to initiate user entry management based on roles, an entry grouping mechanism which appears as an attribute in a user entry.
- Developers to define and manage the configuration parameters of a multitude of default and custom-made services.

You access all three of these functions through a graphical user interface, the web-based Access Manager admin console. In addition, the command-line interface, `amadmin`, enables you to perform batch administrative tasks on the directory server. For example, you can create, add, and activate new services; and create, delete, and read (get) organizations, people containers, groups, roles, and users.

Summary of Access Manager Features

Access Manager provides the following management components. Previously, these components resided within the Portal Server 3.0 framework itself.

- **User Management**—Creates and manages user-related objects (user, role, group, people container, organization, suborganization, and organizational unit objects). These can be defined, modified, or deleted using either the Access Manager console or the command-line interface.

- **Authentication**—Provides a plug-in solution for user authentication. The criteria needed to authenticate a particular user is based on the authentication service configured for each organization in the Portal Server enterprise. Before being allowed access to a Portal Server session, a user must pass through authentication successfully.
- **Single Sign-On**—Once the user is authenticated, the Access Manager API for Single Sign-On (SSO) takes over. Each time the authenticated user tries to access a protected page, the SSO API determines whether the user has the permissions required based on their authentication credentials. If the user is valid, access to the page is given without additional authentication. If not, the user will be prompted to authenticate again.
- **Service Management**—Specifies configuration parameters for default and custom-made services, including those for the Portal Server product itself (Portal Desktop, Rewriter, Search, and NetMail).
- **Policy Management**—Defines, modifies, or removes the rules that control access to business resources. Collectively, these rules are referred to as policy. Policies can be role-based or organization-based and can offer privileges or define constraints.

Comparison: Portal Server 3.0 and Portal Server 6.2

Table 6-1 on page 88 provides an overview to the major changes that have taken place to the Portal Server product. Many functions and features that previously were part of the Sun ONE Portal Server 3.0 (formerly iPlanet Portal Server 3.0) product are now part of Access Manager. In the table, the first column lists a concept or term, the second column defines the function or feature for that term in the Portal Server 3.0 product, the third column describes the corresponding feature or function in the Portal Server 6.2 product.

NOTE These changes were shipped with the Sun Java System 2003Q4 product and this information is retained for users of that product.

Table 6-1 Portal Server 3.0 to Portal Server 6.2 Comparison

Concept or Term	Portal Server 3.0	Portal Server 6.2
Role tree	A hierarchy you configure within Portal Server 3.0 to organize users and applications. The four levels of the role tree are: • root • domain • role • user	Concept of role tree no longer applies. Instead, because Access Manager leverages the capability of Sun Java System Directory Server, you use the Directory Information Tree (DIT) to organize your users, organizations, suborganizations, and so on.
Domain/ Organization	A top-level grouping of users with common interests, such as employees or customers. Note that this is not a DNS domain, but a means that Portal Server 3.0 uses to group users into logical communities.	Concept of domain no longer applies. Instead, the Access Manager *organization* represents the top level of a hierarchical structure used by an enterprise to manage its departments and resources. Upon installation, Access Manager asks for the root suffix, and the default is derived from the domain name (for example, for the domain sun.com, the default is dc=sun, dc=com). Additional organizations can be created after installation to manage separate enterprises. All created organizations fall beneath the top-level organization. Within these sub organizations other suborganizations can be nested. There is no limitation on the depth to the nested structure.
Role	Divides the members of a domain according to function. The role contains a set of attributes and policies that define a user's Desktop policy.	Contains a privilege or set of privileges that can be granted to a user or users. This includes access and management of identity information stored in Sun Java System Directory Server and access to privileges protected by the Access Manager policy module. A Access Manager role also has associated with it a profile, which is stored in the class-of-service template. Role is defined differently in Access Manager and it includes the ability for a single user to have multiple roles, which was previously not supported. The privileges for a role are defined in access control instructions (ACIs). The Access Manager includes several predefined roles. The Access Manager Console allows you to edit a role's ACI to assign access privileges within the Directory Information Tree.

Table 6-1 Portal Server 3.0 to Portal Server 6.2 Comparison *(Continued)*

Concept or Term	Portal Server 3.0	Portal Server 6.2
Attribute	Supports two types of attributes: global and user-configurable. Global attributes apply to the entire platform and are configured only by the Super Administrator. User-configurable attributes apply to underlying levels of the role tree, as described in the following sections. A delegated Domain Administrator can configure these attributes for the domain, parent role, child role, and user levels. At the user level of the role tree, some attributes can be customized for each user, as needed.	Makes use of Access Manager attributes, which can be one of the following types: • Global— The values applied to the global attributes are applied across the Access Manager configuration and are inherited by every configured organization. • Dynamic—A dynamic attribute can be assigned to an Access Manager configured role or organization. When the role is assigned to a user or a user is created in an organization, the dynamic attribute then becomes a characteristic of the user. • Organization—These attributes are assigned to organizations only. In that respect, they work as dynamic attributes. They differ from dynamic attributes, though, as they are not inherited by entries in the subtrees. • User—These attributes are assigned directly to each user. They are not inherited from a role or an organization and, typically, are different for each user. • Policy—Policy attributes are privilege attributes. Once a policy is configured, they may be assigned to roles or organizations. That is the only difference between dynamic and policy attributes; dynamic attributes are assigned directly to a role or an organization and policy attributes are used to configure policies and then applied to a role or an organization.

Table 6-1 Portal Server 3.0 to Portal Server 6.2 Comparison *(Continued)*

Concept or Term	Portal Server 3.0	Portal Server 6.2
Policy	Configures portal access policies to applications, the Desktop, NetFile, Netlet, and so on.	Rules that define who can do what to which resource. The Access Manager Policy Service allows an organization to set up these rules or policies. In general, policy is created at the organization (or suborganization) level to be used throughout the organization's tree. In order to create a named policy, the specific policy service must first be added to the organization under which the policy will be created. In Sun Java System Identity Server 6.2, the policy service consists only of lists of URLs that are allowed or denied. This is not sufficient for Portal Server to build a policy-based Desktop for content. This is why policy for channel access is built into the display profile for the Desktop. The Portal Server 6 Desktop supports a display profile that allows list of channels to be merged from several roles. If, for example, you have 25 roles, each with a handful of channels associated with that role, users can be configured to have any number of those roles, and the Desktop they get will then provide the aggregation of all those roles. Merge semantics control how channels from the various roles are aggregated or merged. For the purpose of merging display profiles, a hierarchical ordering is imposed on the roles in the Portal Server. The merge begins with the lowest priority document (lowest number) and proceeds in increasing priority number, until it arrives at the user (base), the highest priority profile. See Chapter 10, "Administering the Display Profile" for information on merging display profiles.

Table 6-1 Portal Server 3.0 to Portal Server 6.2 Comparison *(Continued)*

Concept or Term	Portal Server 3.0	Portal Server 6.2
Component/ Service	The four major components of Portal Server 3.0 are the server itself, the profile server, the gateway, and the firewall.	Component has been replaced by Access Manager service, which is group of attributes defined under a common name. The attributes define the parameters that the service provides to an organization. Access Manager is the service framework. Portal Server 6 relies on Access Manager to provide core services, such as authentication, user management, and policy management, as well as for the framework to run Portal Server specific services (Desktop, NetMail, Rewriter, and Search).
Administrative interfaces	Provides its own admin console to administer only Portal Server 3.0 components. The command-line interface is `ipsadmin`.	Uses the Access Manager admin console to administer Access Manager services, users, and policy, as well as Portal Server specific services (Desktop, NetMail, Rewriter, and Search.) The command-line interfaces that replace `ipsadmin` are `amadmin`, `dpadmin`, and `rwadmin`.

Comparison: Portal Server 6.0 and Portal Server 6.2

Table 6-2 on page 92 provides an overview to the changes that have taken place between the Portal Server 6.0 product and Portal Server 6.2 product. In the table, the first column lists a concept or term, the second column defines the function or feature for that term in the Portal Server 6.0 product, the third column describes the corresponding feature or function in the Portal Server 6.2 product.

NOTE These changes were shipped with the Sun Java System 2003Q4 product and this information is retained for users of that product.

Table 6-2 Portal Server 6.0 to Portal Server 6 Comparison

Concept or Term	Sun Java System Portal Server 6.0	Portal Server 6
Policy	Assign a policy to users. Once a policy has been named and created, it can be assigned to the organization or role. Assigning a policy at the organization level makes its attributes available to all entries in the organization. Assigning policy to a role makes its attributes available to all users who contain the role attribute.	Delegate an organization's policy definitions and decisions to another organization. (Alternately, policy decisions for a resource can be delegated to other policy products.) A referral policy controls this policy delegation for both policy creation and evaluation. Create a normal policy to define access permissions. A normal policy can consist of multiple rules, subjects, and conditions.
Authentication menu	The authentication menu configuration feature provided by the Sun ONE Identity Server 5.1 administration console supports a menu of authentication modules selected by the user.	If you need to configure a selectable list of valid authentication modules, use the Sun Java System Identity Server administration console to set each authentication module with the same value in the authentication level attribute. Refer to Chapter 6, "Administering Authentication, Users, and Services" for information on configuring authentication modules.

Access Manager Constraints

When using Access Manager, the following constraints apply:

- The predefined Access Manager roles cannot span multiple parallel organizations; however, a role can be assigned to a user who resides in a child organization of the organization that the role is associated with. In addition, access to resources in multiple domains can also be enabled by creating a custom role and defining the necessary Access Control Instructions (ACIs) to grant the role the privileges required.
- A user must belong to an organization and can only belong to that organization.
- Hierarchical roles are not supported. For example, you cannot create role C as equal to the sum of role A and role B, and have a user with role C have access to the resources in Role A, without being explicitly assigned to role A.
- The access permission for the `RoleAdministratorRole` can only be configured through editing corresponding ACIs directly.

- When role administrators (delegated administrators) log in to the Access Manager admin console, they can see all the roles and their associated services and properties under the same organization even if the role administrators don't have the permission to modify them.

Access Manager Interfaces

Access Manager Admin Console

This browser-based console provides a graphical user interface to manage the Access Manager enterprise, including Portal Server services. The admin console has default administrators with varying degrees of privileges used to create and manage the services, policies and users. (Additional delegated administrators can be created based on roles.) See Chapter 7, "Configuring Delegated Administration" for more information.

The Access Manager admin console is divided into three sections: the location pane, the Navigation pane and the Data pane. By using all three panes you navigate the directory, perform user and service configurations, and create policies.

See Chapter 1, "Introduction to Administering the Sun Java System Portal Server" for more information.

Access Manager Command-Line

The Access Manager command-line interface is `amadmin`, to administer the server. `amadmin` is also used to load XML service files into the directory server and perform batch administrative tasks on the directory tree.The iPlanet™ Portal Server 3.0 command-line interfaces, `ipsadmin` and `ipsserver` are no longer used.

For more information on `amadmin`, see the Access Manager documentation.

Logging In to the Access Manager Admin Console

You can log in to the Access Manager console in two ways:

- Using a Specific URL
- Through HTTPS

When you log in to the admin console, the capabilities that are presented to you depend on your access permissions. Access permissions are determined based on the ACIs or roles assigned to you. For example, the superuser sees all of the admin console's functionality; a delegated administrator might only see a subset of this functionality, perhaps for a suborganization; end users see only the user attributes pertaining to their particular user ID.

Currently, there are two URLs available for logging in to the admin console:

- `http://`*host*`:`*port*`/amconsole/`
- `http://`*host*`:`*port*`/amserver/`

The `/amconsole` URL explicitly requests the HTML pages for the Access Manager admin console. If you log in using `/amconsole`, it brings up the admin console and then you'll see the URL change to `/amserver/UI/login` so the user can authenticate. Regardless of the configuration, this URL can be used to access the admin console.

The `/amserver` URL requests the HTML pages for the Access Manager service. Although the default set up when Portal Server is installed is to redirect this URL to log in to the admin console, because the `/amserver` URL accesses the Access Manager service this URL can be used to make other services besides the console available. For example,

- If a user accesses an application with an invalid session, an application may redirect the `/amserver` URL request to `amserver/UI/login` with the goto parameter. For example, the Portal Server Desktop does this as well as the Access Manager agent.
- A customer may direct users to `amserver/UI/login` as their starting point into some application or portal. Their default redirect URL could then be some portal application or custom application.
- A custom application could directly call the `amserver/UI/login` to authenticate.

To log in to the Access Manager admin console

- Using a specific URL:
  Type `http://`*host*`:`*port*`/amserver/`
  or
  Type `http://`*host*`:`*port*`/amconsole/`
- Using HTTPS:
  Type `https://`*host*`:`*ssl_port*`/amconsole/`

Configuring Log in to the Admin Console Using an IP Address

You cannot log in to the Access Manager admin console by using the server's IP address. This is because of the cookie domain settings in Access Manager.

However, you can add the local host's IP address to the list of Cookie Domains on the admin console.

1. Select Service Configuration from the location pane.
2. Click Platform.
3. Add your local host's IP address to Global.

You should now be able to access the admin console with IP address, rather than the domain name.

Viewing Basic Information

A script is available to enable you to display basic information about the product such as the version, build date of the Portal Server as well as the version and build date for the jar file. The version script is installed in *PortalServer-base*/SUNWps/bin directory where *PortalServer-base* is the base directory in which you installed the Portal Server. The default is /opt.

To view product information:

1. Change directories to the directory where the script is installed. That is:

   cd *PortalServer-base***/SUNWps/bin**

2. To view information about the Portal Server, type

   ./version

3. To view information about the jar file on the Portal Server, type

   ./version *jar-file*

   where *jar-file* is the name of the jar file.

Starting and Stopping Portal Server

This section describes how to stop and start Portal Server. You need to restart the each web container instance using the script for that web container. For example:

- To start the Sun Java System Web Server Instance:

  ws-install-base/`https-`*instancename*/`start`

- To stop the Sun Java System Web Server instance:

  ws-install-base/`https-`*instancename*/`stop`

- To start the Sun Java System Application Server:

  ```
  cd /var/opt/SUNWappserver7/domains/domain1
  ```

  ```
  ./asadmin asadmin> start-domain --user admin domain1 asadmin> exit
  ```

NOTE You do not need to stop the server to restart it. If you start a server that is already running, the server is stopped and restarted.

These instructions may vary with the web container. See your web container documentation for more information.

The Portal Server supports various platform locales. To start the Portal Server with a value other than the installed default see the *Sun Java System Portal Server 6 2005Q1 Developer's Guide.*

Managing Access Manager Services

This section provides an introduction to Access Manager services used by Portal Server. See the Access Manager documentation for complete information.

Installation and Sun Java System Web Server Packaging

- The Portal Server installer executes the Access Manager installer if the Access Manager has not previously been installed.

- Portal Server shares the web container with Access Manager. The web container specifies a runtime environment for Web components including concurrency, deployment, life cycle management, security, transaction, and other services.
- Portal Server uses the JVM™ and other components that Access Manager provides.

User Management

- Portal Server stores its profile information in Access Manager using the Access Manager APIs.
- Portal Server leverages multi-role support in Access Manager.
- Portal Server uses open and non-proprietary standard schema attributes, for example, `givenName`.
- Access Manager provides direct access to the LDAP directory.

Single Sign-On/Authentication

- In Portal Server 6, the authentication is managed by Access Manager.
- Access Manager provides all the authentication modules.
- Portal Server uses Access Manager policy attributes to restrict access.

Service Management

Portal Server 6 defines the following Access Manager services:

- Desktop—Provides the portal front-end and is the primary end user interface to the portal.See Chapter 8, “Administering the Portal Desktop Service” for information on setting up and administering the Portal Desktop.
- NetMail—Accesses the IMAP and SMTP mail servers in the Internet and allows users to access mail through the portal. See Chapter 11, “Administering the NetMail Service” for information on setting up and administering NetMail.
- Rewriter—Implements rules set up by the administrator to rewrite URLs to provide appropriate access. See Chapter 12, “Administering the Rewriter Service” for information on setting up and administering the Rewriter.

- Search—Provides a search capability for the Portal Server including basic and advanced search channels of the available documents. See Chapter 13, "Administering the Search Engine Service" for information on setting up and administering the Search service.

Managing Portal Server Users

The Directory Information Tree (DIT) organizes your users, organizations, suborganizations, and so on into a logical or hierarchical structure that enables you to efficiently administer and assign appropriate access to the users assuming those roles or contained within those organizations. This section provides information to help you plan the directory structure or tree underlying your portal server implementation by providing information about the functions and capabilities of organizations, suborganizations, and roles, and also providing procedures for creating and managing organizations, roles, and users.

NOTE Portal Server 6 supports organizations; previously, Portal Server 3.0 used the concept of domains.

The top of the organization tree in Access Manager is specified at install time. Additional organizations can be created after installation to manage separate enterprises. All created organizations fall beneath the top-level organization. Within these suborganizations other suborganizations can be nested. There is no limitation on the depth to the nested structure.

NOTE The top of the tree does not have to be called `isp`. It can be called anything. But with a tree organized with a generic top, for example, `isp`, then organizations within the tree can share roles.

Roles are a new grouping mechanism that are designed to be more efficient and easier to use for applications. Each role has members, or entries that possess the role. As with groups, you can specify role members either explicitly or dynamically. The roles mechanism automatically generates the `nsRole` attribute containing the DN of all role definitions in which the entry is a member. Each role contains a privilege or set of privileges that can be granted to a user or users. In Portal Server 6, multiple roles can be assigned to a single user. The privileges for a role are defined in Access Control Instructions (ACIs). The Portal Server includes

several predefined roles. The Access Manager console allows you to edit a role's ACI to assign access privileges within the Directory Information Tree. Built-in examples include `Top-level Admin Role` and `Top-level Help Desk Admin Role`. You can create other roles that can be shared across organizations.

Planning Organizations, Suborganizations, and Roles

As you plan your DIT structure, you need to decide whether to use a hierarchical or flat tree structure. As a general rule, you should strive to make your tree as flat as possible. However, as the size of your organization grows, a certain amount of hierarchy is important to facilitate granting and managing user access. The three key structural entities in Access Manager for building your DIT structure are organizations (or suborganizations), roles, and users. Before you plan your structure, you should understand the functions, characteristics, and interrelationships of each of these entities.

Organizations and Suborganizations

- Allow creation of hierarchical relationships that can represent or model your enterprise or organization's hierarchy.
- Can contain users created by its corresponding admin. This provides a method of grouping users together for administration and access control purposes. It is typically easier to administer and control access if users with similar needs are grouped together.
- Can be easily created or removed by an admin in a parent organization or suborganization via the admin console. However, when removed, all subordinate organizations and users are also removed, so not suitable when names or structure likely to change.

Roles

- Allow assignment of a privilege or set of privileges to a user or users. Within an organization, multiple roles can be defined to provide specific privilege sets to users.

- Define permissions via Access Control Instructions (ACI), which must be directly edited. Once defined, can be easily assigned or unassigned to an organization, a suborganization or a user. Unassigning a role from one entity only applies to that entity. Roles will still exist and remain assigned and be available for reassignment to other entities, so are more suited for organizations in which access changes will be frequently required.
- Can control visibility of channels and user's ability to overwrite channels. Settings within the XML Display Profile can make channels in the XML document visible or invisible by default. In addition, the default channels in the XML document can be prevented from being overridden.

Users

- Represent the identity of a person. Can be created within an organization or suborganization by its admin.
- Can be associated with multiple roles, but user must be within the roles' scope. In addition users inherit attributes from the suborganization.
- Belong to only one organization or suborganization; however, users can be easily moved from one organization to another if the admin has the privilege to do it.
- Can personalize visibility of channels.

Scenario 1: Hierarchical Structure with Suborganizations and Roles

Although you should strive for as flat a structure as possible, some hierarchy is useful to provide necessary groupings. The high-level steps to create a hierarchical structure are:

1. Creating a top-level organization.
2. Identifying all the functional or organizational groupings of users in your enterprise and determine for which ones you want to create a DIT structural entity, that is, ones that need to have specific privileges. Typically this should be only the largest subdivisions in your enterprise and the administrators for managing them. Use names that are generic or functional, so reorganizations and name changes will not be problematic.
3. For each DIT entity that has some affiliation with the top-level organization, creating either a *suborganization* (that is, an organization under another organization in the Access Manager world) or a *role* for that entity.

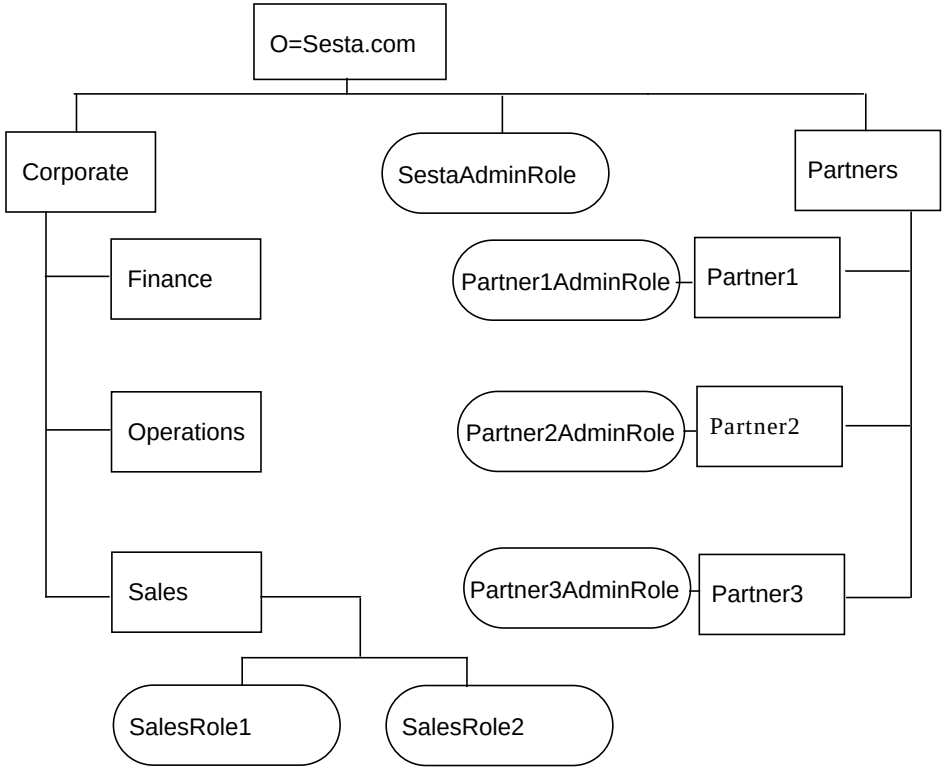
   Use the following guidelines to decide whether to use a suborganization or role:

- Define a suborganization for entities that contain groupings of users with similar access needs. Typically this will be broad functional or organizational entities for which a single set of permissions could be assigned.
- Define a role if it is possible that users in the child organizations need to have this role. All users belong to an organization or suborganization. If they do not have any roles assigned to them, they inherit their permissions from the organization in which they reside. Therefore, if you want a user to have attributes from both the organization they reside in and any parent organizations, you must use the role mechanism and assign them multiple roles.

4. For each role, defining a RoleAdministratorRole to manage the role. Then set the ACIs appropriately (management privileges: add or delete users, modify role attributes, and so on.)
5. Defining the users who will access your enterprise. If users are inheriting their privileges from their organization, place them in the appropriate organization. If users are receiving their privileges through role assignments, they must be placed so that they are within the role's scope, that is, within the organization or a child of the organization in which the role is defined.

Figure 6-1 illustrates a hierarchical directory structure. In this figure, the top-level organization is `Sesta.com`. Directly beneath the top-level is the `SestaAdminRole` to administer the organization and the `Corporate` and `Partners` suborganizations. The `Corporate` organization has three suborganizations: `Finance`, `Operations`, and `Sales`. Because there are multiple types of users within the `Sales` organization, two roles for are defined: `SalesRole1` and `SalesRole2`. Within the `Partners` organization there are three suborganizations: `Partner1`, `Partner2`, and `Partner3`. Each of these organizations, requires its own administrator, so three roles are defined and each one is associated with the appropriate organization.The partner roles are `PartnerAdmin1`, `PartnerAdmin2`, and `PartnerAdmin3`.

Figure 6-1 Hierarchical Directory Structure

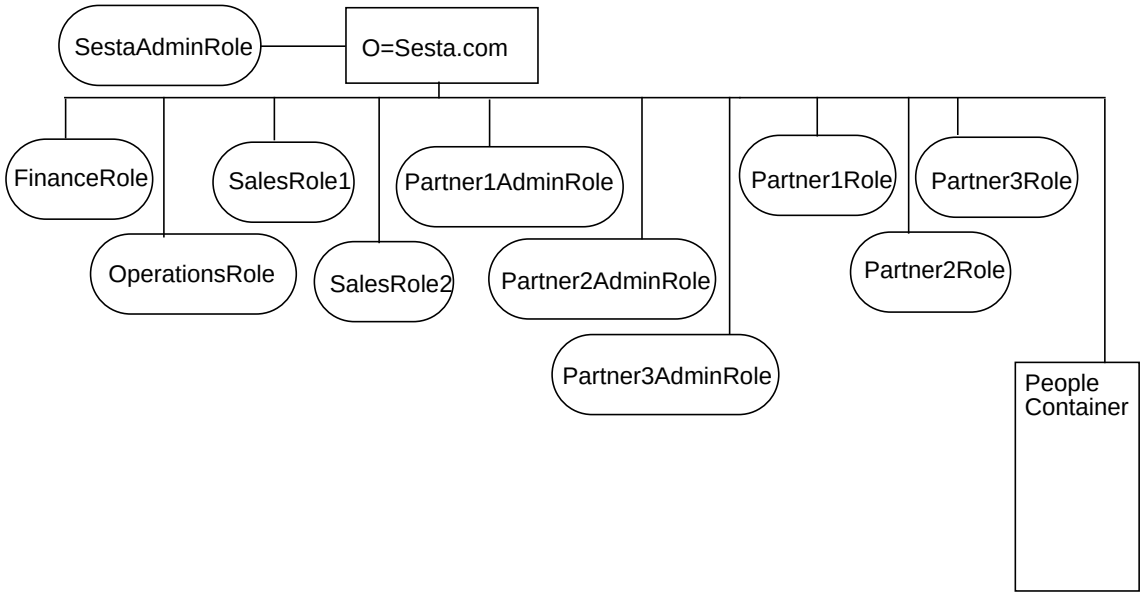


Scenario 2: Flat Tree Structure

If your organization changes often, a flatter or even totally flat tree structure may be appropriate. A structure with one organization, with one People container, and roles all at the same level is often useful if your enterprise changes frequently. With one organization, enterprise changes will not impact your DIT. All access privileges will be defined using roles and since all users are in the single People container and all roles are at the same level, any user can be assigned any role.

Figure 6-2 illustrates a flat directory structure. In this figure, the top-level and only organization is `Sesta.com`. All entities are defined directly beneath this top-level organization. They include the `SestaAdminRole` to administer the organization, four roles for the various corporate functions needed by the Finance, Operations, Sales1 and Sales2 users, and six roles for the user functions required by the partners: `Partner1Role`, `Partner2Role`, `Partner3Role`, `Partner1AdminRole`, `Partner2AdminRole` and `Partner3AdminRole`.

Figure 6-2 Flat Directory Structure



Creating New Organizations and Suborganizations

Organizations and suborganizations allow you to structure and group users for administration and access control purposes. Once you have determined the hierarchy or structure for your enterprise you must create the necessary organizations and suborganizations to implement it. By default, when you create a

new organization or suborganization, there are no services, policies, users, or roles defined for it. Therefore, whenever you create a new organization or suborganization, you need to perform the following high-level steps to configure it:

1. Adding all the services you want available to the organization. See To Add a Service for information. Typically, at a minimum you will want to add the following services:
   - Authentication. The Core authentication service and any authentication service with which users in the organization will use to authenticate (LDAP, anonymous). See Configuring Authentication for further information.
   - URL Policy Agent.
   - User.
   - Portal Server Configuration. Any Portal Server services you want to enable for users in the organization (Portal Desktop and NetMail).
2. Creating templates for each of the added services. See To Create a Template for a Service for more information.
3. Creating the policies needed to grant users within the organization access privileges. See Overview of How Portal Server Uses Policy Management for more information on using policies.
4. Adding users to the organization. See To Add a New User for information.
5. Creating and assigning any roles you want in the organization. See To Create a New Role and To Assign a Role to a User for information.
6. Configuring the services enabled for your organization. To configure the Desktop, see Chapter 8, “Administering the Portal Desktop Service” for information. To configure NetMail, see Chapter 11, “Administering the NetMail Service”.

For a quickstart procedure to create a new organization and configure it to use portal, see Creating a New Portal Organization Quick Start.

To Create a New Organization or Suborganization

See Planning Organizations, Suborganizations, and Roles for recommendations on how to plan your organizations and suborganizations for use with Portal Server.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. If you are creating a suborganization, use the navigation pane to select the organization where the suborganization will be created.

3. Click New in the navigation pane.

   The New Organization page displays in the data pane.

4. Type a value for the name of the organization or suborganization in the New Organization page.

5. Choose a status of `Active` or `Inactive`.

   The default is `Active`. This can be changed at any time during the life of the organization or suborganization by selecting the properties arrow. Choosing `inactive` disables log in to the organization or suborganization.

6. Click OK.

   The new organization or suborganization displays in the navigation pane.

7. Choose Services from the View menu.

8. Click New.

9. Enable the desktop service for the new organization.

   a. Select Identity Management from the location pane.

   b. Select Organizations from the View menu.

   c. Select the newly created organization.

   d. Select Services from the View menu.

   e. Select Portal Desktop

   f. Change the value from DummyChannel to JSPTabContainer (or the name of the op-level container that will be used by the new organization) in Default Channel Name.

g. Change the value from default to sampleportal (or the desktop type that will be used by the new organization) in Portal Desktop Type.

To Add a Service

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization for which you want to add a service.

   Use the View menu in the navigation pane.

3. Choose Services from the View menu.

4. Click New.

5. Select the service or services to add from the data pane and click OK.

To Create a Template for a Service

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization where the added service exists.

   Use the View menu in the navigation pane.

3. Choose Services from the View menu.

4. Click the properties arrow next to the added service.

5. Accept or modify the default attribute values for the service and click Save.

NOTE For the LDAP and POLICY CONFIGURATION services blank password fields are located under the DN for Root User Bind (cn=amldapuser,...) This password needs to be supplied and saved to properly configure policy and ldap configurations. The password is NOT the same as the admin user password. Ask your UNIX administrator for these passwords.

For information on setting Access Manager specific service attributes, see the *Access Manager Administration Guide*. For information on the setting Portal Server specific service attributes, see the appropriate appendix in this guide.

To Add a New User

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

---

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization where the user will be created.

3. Choose Users from the View menu and click New.

   The New User page appears in the data pane.

NOTE If you do not see Users but instead see People Containers in the drop-down menu, then make sure you have set the Show People Containers attribute for your organization, or up at the top level at some point. This is set in the Access Manager Services under Administration.

Users do always go into the People Container, but unless the Show People Containers attribute is selected you will just be able to see and interact with them directly under the organization. Show People Containers is not set by default.

4. Select the services to assign to the user and click Next.

   Typically, at a minimum you will want to add the Portal Desktop, Authentication Configuration, and Subscription services for most users.

5. Enter the user information and click finished.

   The new user appears in the navigation pane.

To Add a Service to a User

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization where the user will be created.

3. Choose Users from the View menu

4. Select the user in the navigation pane and click the Properties arrow.

5. Select Services from the View menu.
6. Click New to choose the services to assign to the users.
7. Check the services and click OK.

   Typically, at a minimum you will want to add the Portal Desktop, and Subscription services for most users.

To Create a New Role

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
2. Navigate to the organization or suborganization where the role will be created.
3. Choose Roles from the View menu and click New.

   The New Role page appears in the data pane.
4. Enter the role information (Name, Description, Role Type, Access Permissions) and click Finish.

   The new role appears in the navigation pane.

NOTE If you are creating a customized role for delegated administration, you must have previously defined the ACI privileges for the role. See Chapter 7, “Configuring Delegated Administration” for information.

To Assign a Role to a User

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
2. Navigate to the organization or suborganization where the role will be created.
3. Choose Users from the View menu.
4. Click the properties arrow next to the user who will be assigned the role.

   The user profile information appears in the data pane.
5. Click Roles from the View menu in the data pane.

   The Add Roles page appears.
6. Check the box next to the roles to assign and click Save.

   The Roles for this User box is updated with the assigned roles.
7. Click Save to save the changes.

Enabling Existing Users to Access the Portal Server

When you install the Portal Server on an existing instance of Access Manager, users are not added to use the Portal Server Desktop. In order to allow users to access the Desktop, you must enable them. Use the following procedures to enable users in the default organization or in another organization.

To Enable Users in the Default Organization

Before you start you will need the to obtain some configuration information. If you do not know all the details of the configuration, the information can be retrieved using a script from the `/var/sadm/pkg/SUNWps/pkginfo` file.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Determine or retrieve the following information from the `/var/sadm/pkg/SUNWps/pkginfo` file:

- The distinguished name for the directory manager (referred to as *DS_DIRMGR_DN/*). Default value is `cn=Directory Manager`.
- The directory manager password (referred to as *DS_DIRMGR_PASSWORD/*).
- The fully qualified domain name of the directory server (referred to as *DS_HOST/*).
- The port on which the directory server runs (referred to as *DS_PORT/*). Default value is 389.
- The root suffix of the directory tree (referred to as *DS_ROOT_SUFFIX/*). Default value is `dc=`*orgname*`,dc=com (such as dc=sun,dc=com)`.
- The default organization of the Portal Server installation (referred to as *DS_DEFAULT_ORG/*). Default value is `o=domain-name`.
- The base directory of the Portal Server installation. Default value is `/opt`.

If you do not know the configuration information, run the following script and refer to the output to obtain the information you will need to complete this procedure.

```
###############################################
# Get configuration from file
###############################################

GrabConfig() {
  GRABCONFIG_KEY=$1
  GRABCONFIG_FILE=$2
  GRABCONFIG_SEPARATOR=$3
  ANSWER_CONFIG=‘$GREP "^$GRABCONFIG_KEY$GRABCONFIG_SEPARATOR"
$GRABCONFIG_FILE | $UNIQ | $SED -e
"s/$GRABCONFIG_KEY$GRABCONFIG_SEPARATOR//" | $SED -e "s/^ //"‘
}

###############################################
# Get PS6 Settings
###############################################

GetPS6Settings() {
  if [ -f $PKGINFO ]; then
    # Ldap Settings

#
    GrabConfig "DS_HOST" $PKGINFO "="
    DS_HOST=$ANSWER_CONFIG
    echo "DS_HOST=$DS_HOST"
    GrabConfig "DS_PORT" $PKGINFO "="
    DS_PORT=$ANSWER_CONFIG
    echo "DS_PORT=$DS_PORT"
```

```
GrabConfig "DS_DIRMGR_DN" $PKGINFO "="
    DS_DIRMGR_DN=$ANSWER_CONFIG
    echo "DS_DIRMGR_DN=$DS_DIRMGR_DN"
    GrabConfig "DS_DIRMGR_PASSWORD" $PKGINFO "="
    DS_DIRMGR_PASSWORD=$ANSWER_CONFIG
    echo "DS_DIRMGR_PASSWORD=$DS_DIRMGR_PASSWORD"

##############################################
# Get PS6 Settings
##############################################

GetPS6Settings() {

  if [ -f $PKGINFO ]; then

 # Ldap Settings
    #
    GrabConfig "DS_HOST" $PKGINFO "="
    DS_HOST=$ANSWER_CONFIG
    echo "DS_HOST=$DS_HOST"
    GrabConfig "DS_PORT" $PKGINFO "="
    DS_PORT=$ANSWER_CONFIG
    echo "DS_PORT=$DS_PORT"
    GrabConfig "DS_DIRMGR_DN" $PKGINFO "="
    DS_DIRMGR_DN=$ANSWER_CONFIG
    echo "DS_DIRMGR_DN=$DS_DIRMGR_DN"
    GrabConfig "DS_DIRMGR_PASSWORD" $PKGINFO "="
    DS_DIRMGR_PASSWORD=$ANSWER_CONFIG
    echo "DS_DIRMGR_PASSWORD=$DS_DIRMGR_PASSWORD"
```

```
# Dsame Settings
    #
    GrabConfig "IDSAME_BASEDIR" $PKGINFO "="
    IDSAME_BASEDIR=$ANSWER_CONFIG
    echo "IDSAME_BASEDIR=$IDSAME_BASEDIR"

AMCONFIG="${IDSAME_BASEDIR}/SUNWam/lib/AMConfig.properties"
    if [ -f $AMCONFIG ]; then
      DS_ROOT_SUFFIX=‘$GREP "^com.iplanet.am.rootsuffix=" $AMCONFIG |

$SED -e "s/com.iplanet.am.rootsuffix=//"‘
      echo "DS_ROOT_SUFFIX=$DS_ROOT_SUFFIX"
      DS_DEFAULT_ORG=‘$GREP "^com.iplanet.am.defaultOrg=" $AMCONFIG | \
                     $SED -e "s/com.iplanet.am.defaultOrg=//"‘
      echo "DS_DEFAULT_ORG=$DS_DEFAULT_ORG"

 else
      print "‘$GETTEXT ’Error - Cannot find DSAME configuration file,
please verify PS6 installation.’‘"
      exit 1
    fi
  else
    print "‘$GETTEXT ’Error - Cannot find SUNWps package information
files, please verify PS6 installation.’‘"
    exit 1
  fi
```

2. Change directories to Access Manager utilities directory. For example, if the base directory is `/opt`, enter:

   `cd /`*`AccessManager-base`***`/SUNWam/bin`**

3. If the root suffix of the directory server and the default organization are not the same, execute the following command:

```
./ldapsearch -h /DS_HOST/ -p /DS_PORT/ -D /DS_DIRMGR_DN/ -w /DS_DIRMGR_PASSWORD/ -b
"ou=People,/DS_DEFAULT_ORG/,/DS_ROOT_SUFFIX/" "(uid=*)" dn | /usr/bin/sed
's/^version.*//' > /tmp/.tmp_ldif_file1
```

4. If the root suffix of the directory server and the default organization are the same, execute the following command:

```
./ldapsearch -h /DS_HOST/ -p /DS_PORT/ -D /DS_DIRMGR_DN/ -w /DS_DIRMGR_PASSWORD/ -b
"ou=People,/DS_ROOT_SUFFIX/" "(uid=*)" dn | /usr/bin/sed 's/^version.*//' >
/tmp/.tmp_ldif_file1
```

5. Execute the following command

```
grep "^dn" /tmp/.tmp_ldif_file1 | awk '{
print $0
print "changetype: modify"
print "add: objectclass"
print "objectclass: sunPortalDesktopPerson"
print "objectclass: sunPortalNetmailPerson\n" }' >
/tmp/.tmp_ldif_file2
```

6. Execute the following command.

```
./ldapmodify -c -h DS_HOST -p DS_PORT \ -D DS_DIRMGR_DN -w
DS_DIRMGR_PASSWORD -f /tmp/.tmp_ldif_file2
```

7. Remove all temporary files.

```
rm /tmp/.tmp_ldif_file1 /tmp/.tmp_ldif_file2
```

To Enable Users in a Non-Default Organization

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Determine or retrieve the following information from the `/var/sadm/pkg/SUNWps/pkginfo` file:

- The distinguished name for the directory manager (referred to as *DS_DIRMGR_DN/*). Default value is `cn=Directory Manager`.

- The directory manager password (referred to as *DS_DIRMGR_PASSWORD/*)
- The fully qualified domain name of the directory server (referred to as *DS_HOST/*)
- The port on which the directory server runs (referred to as *DS_PORT/*). Default value is 389.
- The root suffix of the directory tree (referred to as *DS_ROOT_SUFFIX/*). Default value is `dc=`*orgname*`,dc=com (such as dc=sun,dc=com)`.
- The organization of the Portal Server installation for which you want to update the users (referred to as *DS_ORG_TO_UPDATE/*). Default value is ".
- The base directory of the Portal Server installation. Default value is `/opt`.

2. Add services for the organization or suborganization containing the existing users you want to enable. See To Add a Service for information on the procedure.
3. Create a template for each service you add. See To Create a Template for a Service for information on the procedure.
4. Create and assign policies for each service. See To Add a Policy Service for a Peer or Suborganization, To Create a Referral Policy for a Peer or Suborganization, and To Create a Normal Policy for a Peer or Suborganization for information.
5. Set the URL to which to redirect successfully authenticated users from the organization. See To Redirect Successful Login User to the Portal Desktop URL.
6. Change directories to Access Manager utilities directory. For example, if the base directory is `/opt`, enter

   `cd /`*AccessManager-base*`/SUNWam/bin`**

7. Enable users within the organization or organizations, do one of the following:
   - To enable users only within a particular organization, defined as *DS_ORG_TO_UPDATE/*, then use the following command (type as one line):

`./ldapsearch -h /`*DS_HOST*`/ -p /`***DS_PORT*`/ -D /`***DS_DIRMGR_DN*`/ -w /`***DS_DIRMGR_PASSWORD*`/ -b "ou=People,/`***DS_ORG_TO_UPDATE*`/,/`***DS_ROOT_SUFFIX*`/" "(uid=*)" dn | /usr/bin/sed 's/^version.*//' > /tmp/.tmp_ldif_file1`**

   - To enable users in all organizations, then use the following command (type as one line):

```
./ldapsearch -h /DS_HOST/ -p /DS_PORT/ -D /DS_DIRMGR_DN/ -w /DS_DIRMGR_PASSWORD/
-b "/DS_ROOT_SUFFIX/" "(uid=*)" dn | /usr/bin/sed 's/^version.*//' > /tmp/.tmp_ldif_file1
```

8. Execute the following command:

```
grep "^dn" /tmp/.tmp_ldif_file1 | awk '{
print $0
print "changetype: modify"
print "add: objectclass"
print "objectclass: sunPortalDesktopPerson"
print "objectclass: sunPortalNetmailPerson\n" }' >
/tmp/.tmp_ldif_file2
```

9. Execute the following command:

```
./ldapmodify -c -h DS_HOST -p DS_PORT \ -D "DS_DIRMGR_DN" -w
DS_DIRMGR_PASSWORD -f /tmp/.tmp_ldif_file2
```

10. Remove all temporary files.

```
rm /tmp/.tmp_ldif_file1 /tmp/.tmp_ldif_file2
```

11. Change directory to Portal Server utilities directory.

```
cd /AccessManager-base/SUNWps/bin
```

12. Execute the following to load the display profile for your non-default organization.

```
./dpadmin modify -u
"uid=amadmin,ou=people,DS_DEFAULT_ORG,DS_ROOT_SUFFIX" -w
DS_DIRMGR_PASSWORD -d
"NON_DEFAULT_ORG,DS_DEFAULT_ORG,DS_ROOT_SUFFIX"
AccessManager-base/SUNWps/samples/desktop/dp-org.xml
```

13. To enable users in another organization, repeat steps Step 7 through Step 13.

Creating a New Portal Organization Quick Start

The following task describes the steps to create a new organization and enable it for portal use. By default, when you log in, Identity Management is selected in the location pane, and Organizations is selected in the Navigation pane.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Create the new organization.
   a. Select Organizations from the View menu.
   b. Click New.

      The Create Organization page opens in the data pane.
   c. Type the new organization name. The Organization Status should be Active. Click OK.

      The newly created organization appears in the navigation page.
2. Add services for the new organization.
   a. Select Organizations from the View menu in the navigation pane and click on the newly created organization from the Name list.
   b. Select Services from the View menu.
   c. Click New.

      The Add Services page appears in the data pane. Select the services you want to register for your organization. Minimum services that need to be added are:

      - Core
      - LDAP (or any authentication service that you will be using for this organization.
      - Membership
      - Portal Desktop

        For the purposes of this procedure, the following should also be registered.
      - Policy configuration
      - Subscriptions
      - User Management

      The newly added services appear in the navigation pane.

d. Configure each service by clicking the properties arrow. Click Create to modify the configuration attributes. See the *Sun Java System Access Manager Administration Guide* for a description of attributes that are not specific to Portal Server configuration

NOTE Suborganizations must add their services independently of the parent organization.

3. Create the Desktop referral policy from the parent organization to the new organization.

The referral must define the parent organization as the resource in the rule, and it must contain a SubOrgReferral with the suborganization as the value in the referral.

a. Select Identity Management from the location pane.

b. Select the parent organization.

c. Select Policies from the View menu.

d. Click New to create new policy.

The Create Policy page appears in the data pane.

e. Select Referral for the type of policy.

f. For Name, type SubOrgReferral_Desktop. Then click OK.

The Policy is created and appears under Policies.

g. Click the properties arrow next to SunOrgReferral_Desktop.

h. Click Rules from the View menu in the data pane and click New. Make sure Portal Desktop is selected and click Next.

i. Specify a name for the Portal Desktop rule and click Finish.

j. Click Referrals from the View menu in the data pane and click Add. Make sure that the name of the suborganization is selected for Value in the data pane and click Create to complete the policy's configuration.

4. Create a normal Portal Desktop policy for the new organization.

a. Navigate to the sub organization.

b. Choose Policies from the View menu.

The policies for that organization are displayed.

c. Select New in the navigation pane. The New Policy page opens in the data pane.

d. Make sure you select Normal in Type of Policy.

e. Type a name for the policy.

f. Click OK.

g. Choose Rules from the View menu in the data pane and click New. The Add Rule page opens in the data pane.

h. Specify the name of the rule and select an action under Set Rule Actions. Click Finish.

i. Choose Subjects from the View menu in the data pane and click New. The Add Subject page opens in the data pane.

j. Select a subject that the Portal Desktop policy will be applied and choose Next to complete the subject configuration.

k. Click Finish to complete the policy's configuration.

5. Create a new user in the new organization.

a. Select Identity Management from the location pane.

b. Select Organizations from the View menu.

c. Select the newly created organization.

d. Select Users from the View menu.

e. Click New.

f. Choose the services you want to register for the user.

g. Click Next.

h. Enter the user details in the text fields.

i. Click Finish.

6. Enable the desktop service for the new organization.

a. Select Identity Management from the location pane.

b. Select Organizations from the View menu.

c. Select the newly created organization.

d. Select Services from the View menu.

e. Select Portal Desktop.

f. Change the value from default to sampleportal (or the desktop type that will be used by the new organization) in Portal Desktop Type.

7. Access the new organization's Desktop.

a. Log out of the admin console.

b. Open a browser page and type:

`http://`*server*`:`*port*`/amserver/UI/login?org=`*neworg*

The users's Desktop should appear.

Configuring Authentication

This section describes how to configure Portal Server authentication. Access Manager provides a framework for authentication. Authentication is implemented through plug-in modules that validate the user's identity. Access Manager provides seven different authentication modules as well as a Core authentication module. The Access Manager admin console is used to set the default values, to add authentication services, to create an organization's authentication template, and to enable the service. Because the Core authentication module provides the overall configuration for authentication, the Core authentication module must be added and a template for it created for each organization before you can configure any of the specific authentication modules.

NOTE The authentication menu configuration feature provided by the Sun ONE Identity Server 5.1 administration console is not supported in this release of Sun Java System Access Manager. If you need to configure a selectable list of valid authentication modules, use the Access Manager administration console to set each authentication module with the same value in the authentication level attribute. Refer to To Configure the Authentication Menu for information on configuring authentication modules.

During installation the Core authentication is added and a template is created for it in the default organization. In addition, the installation also adds and creates templates for the following authentication modules:

- LDAP—LDAP authentication allows any valid user within the search base of the directory tree to log in to the Portal Server. This will automatically assign a user to a specific role.
- Membership—Membership authentication allows a user to create an account and personalizes it without the aid of an administrator. With this new account, the user can access it as an add user.

NOTE Although the installation configures a basic authentication implementation consisting of the Core, LDAP and Membership modules, you will need to configure authentication manually if you create new organizations or if you want to set up additional authentication functionality such as the ability to authenticate to an external LDAP directory or identity provider.

The high-level steps to configure an authentication module are as follows:

1. Adding the Core authentication service for each new organization. See To Add a Service for the steps to add a service.
2. Creating a template for the Core authentication service. See To Create a Template for a Service for the steps to create template for a service.
3. Adding the authentication services to support for each organization. See To Add a Service for the steps to adding a service.
4. Creating service templates for the authentication services to support for the organization. See To Create a Template for a Service for the steps to create a template for an authentication service. For information on the setting the service attributes, see the *Access Manager Administration Guide*, Chapter 5, “Authentication Options.”
5. Configuring the authentication menu. See To Configure the Authentication Menu for the steps to configure the authentication order.
6. Configuring the order to use authentication services. See To Configure Authentication Order for the steps to configure the authentication order.

Authentication By Authentication Level

Each authentication module can be associated with an integer value for its authentication level. Authentication levels can be assigned by clicking the authentication module's Properties arrow in Service Configuration, and changing the corresponding value for the module's Authentication Level attribute. Higher authentication levels define a higher level of trust for the user once that user has authenticated to one or more authentication modules.

To Configure the Authentication Menu

Users can access authentication modules with a specific authentication level. For example, a user performs a login as a user with the following syntax:

`http://`*`hostname:port/deploy_uri`*`/UI/Login?authlevel=`*`auth_level_value`*

All modules whose authentication level is larger or equal to *auth_level_value* will displayed as an authentication menu for the user to choose. If only one matching module is found, then the login page for that authentication module will be directly displayed.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

---

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, when you log in, Identity Management is selected in the location pane, and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization that you want to configure authentication for.

   Use the View menu in the navigation pan

3. Choose Services from the View menu and click New.

4. Click the properties arrow next to Core.

5. Enable the appropriate authentication modules by selecting them in the Organization Authentication Modules field of the Organization section.

   By default, Portal Server installation enables LDAP and Membership.

6. Enter a value in the Default Auth Level for each authentication module (default is 0).

   The value for each authentication module must be the same in order to appear in the authentication menu.

7. Click Save.

To Configure Authentication Order

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, when you log in, Identity Management is selected in the location pane, and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization that you want to configure authentication for.

   Use the View menu in the navigation pan

3. Choose Services from the View menu and click New.

4. Click the properties arrow next to Core.

5. Enable the appropriate authentication modules by selecting them in the Organization Authentication Modules field of the Organization section.

   By default, Portal Server installation enables LDAP and Membership.

6. Enter a value in the Default Auth Level for each authentication module (default is 0).

   The value for each authentication module must be the same in order to appear in the authentication menu.

7. Select Edit in Organization Authentication Configuration to specify the attribute information for each authentication module.

   a. Click Add to add an authentication module to the menu.

b. Click Reorder to change the order that the authentication modules will appear in the authentication module.

c. Click Save to save the attribute information.

8. Click Save

9. Use the following URL to verify that the authentication menu appears with the appropriate choices by logging in to the admin server.

`http://`*host*`:`*port*`/amserver/UI/login`

If this is not the default organization, use the following URL to verify the authentication menu for the organization:

`http://`*host*`:`*port*`/amserver/UI/login?org=`*org_name*

To Configure LDAP Authentication to an External Directory

When you install the Portal Server, the installation program configures LDAP authentication to directory instance automatically. The installation program allows you to install an internal instance of the directory on the local server and configure LDAP authentication to that internal directory or to configure LDAP authentication to a pre-existing external instance of the directory. Once you have your initial configuration, there are some scenarios where you might want to configure authentication to an external LDAP directory. For example, you may want to isolate authentication information for particular organization onto a dedicated LDAP server for performance or security reasons.

CAUTION Do not configure authentication to an external LDAP directory for the organization containing the `amadmin` user. This can prevent the `amadmin` user from authenticating and lock you out of the admin console. If you do inadvertently configure the organization containing the `amadmin` user, you will need to log in using the full DN of the `amadmin` and then correct the LDAP template. The `amadmin` DN is listed in the `com.sun.authentication.super.user` property in the `AMConfig.properties` file.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization that you want to configure authentication for.

   Use the View menu in the navigation pane.

3. Choose Services from the View menu.

4. Click the properties arrow next to Core from Access Manager Configuration.

5. Check Dynamically Created from the Dynamic User Profile menu.

6. Click the properties arrow next to LDAP from the Access Manager Configuration menu.

7. Set the appropriate LDAP Attributes for your server. The following example sets up access to the LDAP server `ds-sesta1.sesta.com` on port 389 with a search start point of `ou=people,dc=sesta,dc=com` and using a root user bind to `cn=root,ou=people,dc=sesta,dc=com`:

   Primary LDAP Server and Port: `ds-sesta1.sesta.com`:389
   Secondary LDAP server and port: `ds-sesta1.sesta.com`:389
   DN to Start User Search: `ou=people,dc=sesta,dc=com`
   DN for Root User Bind: `cn=root,ou=people,dc=sesta,dc=com`
   Password for Root User Bind: `root password`
   User Naming Attribute: `uid`
   User Entry Search Attributes: `employeenumber`
   User Search Filter: blank
   Search Scope: `subtree`
   Enable SSL to LDAP Server: `off`
   Return User DN to Auth: `off`
   Authentication Level: `0`

8. Click Save.

Configuring Anonymous Authentication

The Portal Server supports two methods for implementing anonymous authentication:

- Using the Authentication-less User ID attributes. Users accessing the Desktop URL are automatically authenticated and granted access to the Desktop.
- Using an Anonymous user session. Users select Anonymous from the Authentication menu, log in as `anonymous`, and are granted access to the Desktop.

To support anonymous authentication, the Portal Server installation program creates a user account, `authlessanonymous`, and sets up access for this user within the following two Portal Desktop Services global attributes:

- Authorized Authentication-less User IDs
- Default Authentication-less User ID

Portal Server can support both authentication-less and anonymous authentication to be configured at the same in the sense that you can do the following:

1. Configure the Desktop to work in authentication-less mode.
2. Configure the authentication menu so that Anonymous is one of the displayed choices.
3. Access the Desktop with browser A, thereby accessing it in authentication-less mode.
4. Access `http:/`*server*`/amserver/UI/login` with browser B, and select Anonymous, and see the Desktop.

At this point you are using authentication-less mode in browser A and anonymous mode in browser B.

The way in which the Desktop is accessed occurs in two different ways. One, authentication-less access, was through a direct reference to `/portal/dt` and the other (anonymous) was indirectly through `/amserver/UI/login`.

The Access Manager Login menu could be avoided by configuring Access Manager to only have anonymous login in the menu.

Both authentication-less access and anonymous authentication are not supported simultaneously in that when you access `/portal/dt` without an Access Manager session, only one of two things happens:

a. The Desktop will redirect to `/amserver/UI/login`, which may automatically do an Anonymous login and redirect you back to `/portal/dt`.

b. The Desktop will run in authentication-less access mode.

You do not have to disable anonymous authentication to use authentication-less access. But if you want the above item a to work, you have to disable authentication-less access mode.

To Configure Anonymous Authentication (Anonymous User Session Method)

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization that you want to configure authentication for.

   All created organizations are displayed in the navigation pane.

3. Select Service Configuration in the location pane.

4. Click the properties arrow next to the Portal Desktop service.

   The Portal Desktop attributes appear in the data pane.

5. Select the value listed in the Authorized Authentication-less User IDs attribute and click Remove.

6. Select the value listed in the Default Authentication-less User ID attribute and click Remove.

7. Click Save.

8. Choose Identity Management from the location pane.

9. Choose Organizations from the View menu.

   All created organizations are displayed in the navigation pane.

10. Navigate to the organization or suborganization that you want to configure authentication for.

    Use the View menu in the location pane.

11. Choose Services from the Show menu.

12. Add and configure the Anonymous service.

    See To Add a Service and To Create a Template for a Service for information.

13. Add Anonymous to the Authentication menu.

    See To Configure Authentication Order for information.

14. Create an `anonymous` user account.

    See To Add a New User for information.

To Configure Anonymous Authentication (Authentication-less Access)

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. By default, when you log in, Identity Management is selected in the location pane, and Organizations is selected in the Navigation pane.

   All created organizations are displayed in the navigation pane.

3. Navigate to the organization or suborganization that you want to configure authentication for.

   Use the View menu in the navigation pane.

4. Create an `authlessanonymous` user account with the password `authlessanonymous`.

   See To Add a New User for information.

5. Select Service Configuration in the location pane.

6. Select Portal Desktop in the navigation pane.
7. Add the fully distinguished name for the `authlessanonymous` user to the Authorized Authentication-less User IDs attribute. For example:

   `uid=authlessanonymous, ou=People, dc=sesta, dc=com`

8. Specify the fully distinguished name for the `authlessanonymous` user in the Default Authentication-less User ID attribute.
9. Click Save.

You must close and restart your browser to access the Desktop using the newly configured Authentication-less User ID method. The Authentication-less User ID method allows you to specify the UID of the user account in the query string. The auless UID is “desktop.suid” by default. The prefix “desktop” is controlled by the config parameter “cookiePrefix” in the `desktopconfig.properties` file. For example, to access the Desktop from the default organization of sestat.com, use the following URL:

`http://`*`server:port`*`/portal/dt?desktop.suid=uid=authlessanonymous, ou=People,dc=sesta,dc=com`

NOTE If a user logs in a browser with locale that is not the user’s own language, all other users will share the same locale at the login prompt.

There are multiple options to get around this problem.

- Turn off caching by changing the value for `refreshTime` to 0 for `JSPTabContainer` in `dp-anon.xml`.
- You can specify multiple authentication-less users, one authentication-less user per locale and redirect the authentication-less desktop to the right user based on browser’s locale.

Configuring Portal Server for Federated Users

The Sun Java System Portal Server software supports users that have federated identities conforming to the Liberty Alliance specification. A federated user that are Liberty single signed on can access a personalized desktop at a portal server without the need for further authentication.

See the *Sun Java System Access Manager Administration Guide* for more information about Liberty-enabled authentication services. Example configurations with Portal Server acting as a service provider can be found in the following location:

`PortalServer-base``/SUNWps/samples/liberty`

To Configure Federated Users

By default, federated users do not have permission to access the Sun Java System Portal Server acting as a service provider. Portal Server can handle federated users as follows:

- Federated users who are Liberty single signed on can access a personalized portal desktop.
- Federated users that are not Liberty single signed on are redirected to the authentication page of an identity provider

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
2. Select Service Configuration in the location pane.
3. Select Portal Desktop in the navigation pane.
4. Check Enable Federation.
5. Specify the ID of the host provider.
6. Click Save.

To Configure Authentication-less Access for Federated Users

By default, federated users do not have permission to access the authentication-less portal desktop.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

---

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
2. Navigate to the organization or suborganization that you want to configure authentication for.

   Use the View menu in the navigation pane.

3. Select Service Configuration in the location pane.
4. Select Portal Desktop in the navigation pane.
5. Uncheck Disable Authentication-less Access for Federated Users.
6. Click Save.

See To Configure Anonymous Authentication (Authentication-less Access) for more information on authentication-less access.

To Configure UNIX Authentication

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
2. Choose Organizations from the View menu in Identity Management.

   All created organizations are displayed in the navigation pane.
3. Select Service Configuration in the location pane.
4. Click the properties arrow next to UNIX in the navigation pane (under Access Manager Configuration).
5. Set the appropriate UNIX Attributes for your server.
6. Click Save.
7. Navigate to the organization or suborganization that you want to configure authentication for.

   Use the View menu in the navigation pane.
8. Choose Services from the View menu.
9. Click New in the navigation pane.
10. Click Core under Authentication in the data pane.

11. Select Unix from the Organization Authentication Modules menu in the data pane.
12. Click Save.

To Configure UNIX Authentication for the Organization Level

The UNIX authentication documented in To Configure UNIX Authentication is for configuring UNIX globally. This procedure is to configure at the organization level.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator (amadmin) by entering `http://`*fullservername*`:`*port*`/amconsole` in your browser's web address field.
2. At the logon screen, enter amadmin as the user ID and the passphrase you chose during installation.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
3. Choose Organizations from the View menu in Identity Management.

   All created organizations are displayed in the navigation pane.
4. Choose Services from the View menu.
5. Select New.
6. Check UNIX in the right pane and click OK.
7. Select the properties arrow next to UNIX.
8. Select Yes in the Create Service Template (Unix) pane.
9. Set the appropriate UNIX Attributes for your server.
10. Select Save.
11. Select the properties arrow next to Core.
12. Highlight UNIX in Authentication Menu and select Save.

Overview of How Portal Server Uses Policy Management

This section describes how to use Access Manager Policy Management feature. See the Access Manager documentation for procedures to create, modify, and delete policies.

The Access Manager Policy Service enables you to define rules or access to resources. Policies can be role-based or organization-based and can offer privileges or define constraints. Portal Server ships with three policies:

- Ability to execute Portal Server Portal Desktop - Enables users to display the Desktop
- Ability to execute Portal Server NetMail - Enables user to run NetMail

NOTE Chapter 8, "Administering the Portal Desktop Service" and Chapter 11, "Administering the NetMail Service" provide detailed descriptions on assigning their specific policies.

By default, the Policy Configuration service is automatically added to the top-level organization. Suborganizations must add their policy services independently of their parent organization. Any policy service you create must be added to all organizations. The high-level steps to use policies are:

1. Adding the Policy service for an organization. (This will be done automatically for the organization specified at installation.) Suborganizations do not inherit their parent's services, so you need to add a suborganization's Policy service. See To Add a Service for information.
2. Creating a referral policy for a peer or suborganization. You can delegate an organization's policy definitions and decisions to another organization. (Alternately, policy decisions for a resource can be delegated to other policy products.) A referral policy controls this policy delegation for both policy creation and evaluation. It consists of a rule and the referral itself. If the policy service contains actions that do not require resources, referral policies cannot be created for suborganizations. See To Create a Referral Policy for a Peer or Suborganization for information.
3. Creating a normal policy for a peer or suborganization. You create a normal policy to define access permissions. A normal policy can consist of multiple rules, subjects, and conditions. See To Create a Normal Policy for a Peer or Suborganization for information.

To Add a Policy Service for a Peer or Suborganization

Peer or Suborganizations do not inherit their parent's services, so you need to add a peer or suborganization's Policy service.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization that you want to create a referral policy.

   All created organizations are displayed in the navigation pane.

3. Select Organizations from the View menu in the navigation pane and select desired organization from the Name menu.

4. Select Services from the View menu.

5. Click New.

   The Add Services page appears in the data pane. Click the check box for the to the following minimum services, then click OK.

   - LDAP
   - Membership
   - Policy Configuration
   - Portal Desktop
   - NetMail

   The newly added services appear in the navigation pane.

6. Configure each service by clicking the properties arrow. Click Create to modify the configuration attributes. See the *Sun Java System Access Manager Administration Guide* for a description of attributes that are not specific to Portal Server configuration

To Create a Referral Policy for a Peer or Suborganization

You can delegate an organization's policy definitions and decisions to another organization. A referral policy controls this policy delegation for both policy creation and evaluation. It consists of a rule and the referral itself. The referral must define the parent organization as the resource in the rule, and it must contain a SubOrgReferral or PeerOrgReferral with the name of the organization as the value in the referral.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization that you want use to create a referral policy.

   All created organizations are displayed in the navigation pane.

3. Select Policies from the View menu.

4. Click New to create new policy.

   The Create Policy page appears in the data pane.

5. For Name, type either SubOrgReferral_*organization* or either PeerOrgReferral_*organization*. Make sure you select Referral in Type of Policy. Then click OK.

6. Select the type of service in Service and click Next.

7. Click Rules from the View menu in the data pane and click Add. Then click Next.

   The Add Rule template appears in the data pane.

8. Enter the name of the rule in Rule Name and click Finish.

9. Click Referrals from the View menu in the data pane and click Add.

   The Add Referral template appears in the data pane.

10. Enter SubOrgReferralName in Name.

    Make sure that the name of the suborganization is selected for Value in the data pane and click Create to complete the policy's configuration.

11. Click Save in the data pane.

    The message "The policy properties have been saved" is displayed when the data is saved.

To Create a Normal Policy for a Peer or Suborganization

You create a normal policy to define access permissions. A normal policy can consist of multiple rules, subjects, and conditions.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Management Server administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization that you want to assign a policy.

   All created organizations are displayed in the navigation pane.

3. Choose Policies from the View menu.

   The policies for that organization are displayed.

4. Select New in the navigation pane. The New Policy page opens in the data pane.

5. For Name, type either SubOrgNormal_*organization* or either PeerOrgNormal_*organization*.Make sure you select Normal in Type of Policy. Click OK.

6. Select a service from the Service menu and click Next. Enter the name of the rule in Rule Name. Make sure the appropriate checkbox is selected to grant execution privilege to the desired service.
7. Choose Rules from the View menu in the data pane and click Add. The Add Rule page opens in the data pane.
8. Choose Subjects from the View menu in the data pane and click Add. The Add Subject page opens in the data pane.
9. Click Finish to complete the policy's configuration.

   The message "The policy properties have been saved." is displayed when the data is saved.

Logging In to the Portal Server Desktop

If you installed the sample portal, users will be able to log in to the sample Desktop. In addition, the Portal Server supports a variety of other user logins. This section describes some of the other user ways users can log in to the Portal Server.

To Log In to the Sample Portal Desktop

To access the sample Desktop, type the following URL:

http://*server:port***/portal/dt**

To Log In to a Suborganization

If users have access privileges to an organization, they can also log in to suborganizations within the organization. For example, if a user has access to the organization A which has a suborganization B, type the following URL to log in to suborganization B:

http://*server:port***/amserver/UI/login?org=B**

To Log On Using Anonymous Authentication

NOTE You must add the anonymous authentication module to support anonymous authentication. See Configuring Anonymous Authentication for information on adding and enabling anonymous authentication modules.

1. Log on using the following URL:

   http://*server*:*port*/**portal/dt**

2. At the Access Manager authentication page, click Anonymous.
3. The sample Desktop appears.
4. If desired, and if the Membership authentication module has been added, use the Login screen to create and add a user ID.

Managing Logging

Portal Server uses the Access Manager logging and debugging APIs.

By default, the Portal Server log and debug files are located in:

- `/var/opt/SUNWam/logs`
- `/var/opt/SUNWam/debug`

The Access Manager admin console allows you to define the following logging attributes:

- Max Log Size
- Number Of History Files
- Log Location
- Logging Type
- Database User Name
- Database User Password
- Database Driver Name

See the *Sun Java System Access Manager 2005Q1 Administration Guide* for further information.

Chapter 7

Configuring Delegated Administration

This chapter describes how to configure delegated administration for Sun Java™ System Portal Server.

This chapter contains these sections:

Overview of Delegated Administration

As enterprises create larger and more complex portals, a centralized administration model is no longer viable. Delegated administration or Line of Business (LOB) administration addresses this issue by delegating or distributing the administration tasks to the actual portal users.

The Portal Server allows you to delegate administration functions to users by using roles. Role-based administration enables an enterprise to break its business into smaller organizations or lines of business (LOB) and then allows different users to administer the organizations, suborganizations, users, policy, roles, and channels of the LOB based on the user's roles.

Table 7-1 on page 142 lists and defines some important delegated administration terms as they apply in the Portal Server. The table contains two columns: the first column lists the term and the second column gives a brief description.

Table 7-1 Delegated Administration Terms

Term	Description
Privilege	The combination of a single resource and a single action that can be performed upon the resource (for example, view a static web page, view paystubs in a paycheck application, modify W-4 data in the paycheck application, and so on).
Action	Actions are a procedure or operation that can be performed on a resource (for example, read a catalog, write a catalog, get email using POP, get email using IMAP, and so on).
Resource	A resource is something that can be abstractly represented in software and whose access is controlled and protected. In Sun Java System Access Manager, the Resource refers to the URL Access only.
Top-level Admin role	A role that has complete management rights to all policy and identity settings.
Organization admin role	A role that has complete management rights to policy and identity settings for an organization.
Line of Business (LOB)	Line of business capabilities are administration capabilities that can be done by a business analyst or equivalent position. LOB administrators are able to perform administrative tasks that do not require Top-level Admin capabilities to complete. Typically, LOB capabilities, such as adding or removing users to and from roles that grant access to resources, would be available only within their sphere of interest.
Role administrator role	A role administrator role is a role with the access permissions to administer some other specific roles and a certain set of user objects. For example, adding or removing users from a role or editing role level attributes.
Role administrator	Role administrators are users to whom role administrator roles have been assigned.

Delegated Administration Roles

The Sun Java Sytem Access Manager administration console provides role-based delegated administration capabilities to different kinds of administrators to manage organizations, users, policy, roles, and channels based on the given permissions.

Sun Java Sytem Access Manager administration console provides a number of predefined administrator roles for delegating administration functions. They are as follows:

- Top-Level Admin
- Group Admin

- Organization Admin
- Organization Help Desk Admin
- People Container Admin
- Container Admin
- Container Help Desk Admin

For detailed information on these roles, refer to the Sun Java Sytem Access Manager product documentation.

NOTE Sun Java Sytem Access Manager also implements three other roles: Top-level Admin, Top-level Help Desk Admin, and Deny Write Access. These roles are created during installation and only exist at the root of the installation. Any new organizations created will not get these three roles. By default, when a new organization is created, three roles get created with it: Organization Admin, Organization Help Desk Admin, and People Admin.

You can use these predefined administrator roles to set up your delegated administration implementation if their function fits the need. For example, if the directory structure for your model comprises an organization with multiple sub-organizations, you could assign Organization Admin roles to users to create delegated administrators for each of the suborganizations. However, if the organizational structure of your enterprise is more complicated, you might want to create a delegated administration model that targets your specific needs. To do this, the Sun Java Sytem Access Manager administration console allows you to define delegated administrator roles with privileges specific to your business needs.

To implement an enterprise-specific delegated administration model, there are three critical conceptual roles:

- Top-level Admin Role
- Organization Admin Role
- Role Administrator Role

The Top-level Admin Role is created when the system is set up, and the Organization Admin Role is created automatically when a new organization is set up. The Role Administrator Role is a role you create based on the requirements of the delegated administration model. The access permissions for the Role Administrator Role are defined by directly editing the corresponding Access Control Instructions (ACIs).

In a delegated administration, the following principles apply:

- User privileges are granted by the user's role.
- Privileges are granted on a per individual user basis by defining a role with desired privileges and assigning this role to the individual user.
- Sets of users can be grouped together by assigning them a specific role. These users will be granted the set of privileges and inherit the values for dynamic attributes that are defined for that role.
- Users can have multiple or aggregated roles. Users with multiple roles have access to combined features of all their roles. When there is a conflict in the features granted by aggregated roles, conflict resolution is based on the priority configured through Conflict Resolution Level defined for the each of the services for those roles. There are seven conflict resolution settings available ranging from Highest to Lowest. When an attribute conflict occurs as role templates from multiple roles are merged, the attribute on the template set with the highest conflict resolution level is returned.

Developing a Delegated Administration Model

In order to delegate administration functions for the Portal Server appropriately, you should develop a delegated administration model to help determine the administration roles required for you enterprise. Consider the following when developing your model:

- Focus on the business requirements of your enterprise. In general, the proposed solution for the role-based delegated administration should be parallel with the business requirements.
- Develop a directory structure that enables users to be grouped so they can access their required resources and have their administration needs managed by a delegated administrator.

- Try to fit your business entities into a more standard tree structure as much as possible while still addressing all the business requirements. You can use a structure with a hierarchy of organizations and suborganizations or a flat directory tree structure. In a flat directory structure, all the entities are defined immediately beneath the top level organization and all the roles (including Role Administrator Roles) are “parallel” to each other in terms of the organizational hierarchy. For example, all the users who are affiliated with business unit would be created in people containers under the top-level organization. For each of the access roles and administrative roles needed in your model a corresponding role at the top-level would be created.

Configuring Delegated Administration

The high-level steps that you perform to configure a delegated administration implementation for the Portal Server are:

1. Defining the ACI settings for the Role Administrator Roles
2. Creating new Admin Roles for the delegation model
3. Assigning Role Administrator Roles to users
4. Configuring Additional Restrictions on a Role

Defining the ACI Settings for Role Administrator Roles

To configure the appropriate privileges for any of the role administrator roles you identified in your delegation model, you must define the appropriate permissions in an ACI for each unique role in your delegation model. You can define an ACI permission template for a role using the Sun Java Sytem Access Manager administration console or the Directory Server console. You can also define an ACI for a specific role using the `ldapmodify` command.

Use the following format when defining ACI permission templates in the Sun Java Sytem Access Manager administration console or with the Directory Server console:

permission_name | *aci_desc* | *dn:aci ## dn:aci ## dn:aci*

where:

permission_name is the name of the permission.

aci_desc is a text description of the access these ACIs allow.

dn:aci represents pairs of DNs and ACIs separated by ##. Sun Java Sytem Access Manager sets each ACI in the associated DN entry.

This format also supports tags that can be substituted for values that would otherwise have to be specified literally in an ACI: ROLENAME, ORGANIZATION, GROUPNAME, and PCNAME. Using these tags lets you define roles flexible enough to be used as defaults. When a role is created based on one of the default roles, tags in the ACI resolve to values taken from the DN of the new role.

For detailed information setting ACIs, refer to the *Sun Java Sytem Access Manager Programmer's Guide*.

NOTE In these example ACI definitions, the root suffix is assumed to be `dc=sesta,dc=com`.

To Define an ACI Using the Command Line

1. Create a text file containing the ACI settings for use with the `ldapmodify` command. For example, the following file, `acis.ldif`, contains an ACI definition of two roles called `JDCAdmin1` and `JDCAdmin2`.

```
dn:dc=sesta,dc=com
changetype:modify
# aci for JDCAdmin1 role
# This role can add/delete users from JDC role
add:aci
aci: (target= "ldap:///ou=people,dc=sesta,dc=com") (targetattr = "*")(version 3.0; acl
"Allow JDCAdmin1 Role to read and search users"; allow (read,search) roledn =
"ldap:///cn=JDCAdmin1,dc=sesta,dc=com";)
-
add:aci
aci: (target="ldap:///dc=sesta,dc=com")
(targetfilter="(entrydn=cn=JDC,dc=sesta,dc=com)")(targetattr="*")(version 3.0; acl "Allow
JDCAdmin1 Role to read and search JDC Role";allow (read,search)
roledn="ldap:///cn=JDCAdmin1,dc=sesta,dc=com";)
-
add:aci
aci:
(target="ldap:///ou=people,dc=sesta,dc=com")(targetattr="nsroledn")(targetfilter="(!(|(nsro
ledn=cn=Top-level Admin Role,dc=sesta,dc=com)(nsroledn=cn=Top-level Help Desk Admin
Role,dc=sesta,dc=com)(nsroledn=cn=Organization Admin
Role,dc=sesta,dc=com)(nsroledn=cn=Top-level Policy Admin
Role,dc=sesta,dc=com)))")(targattrfilters="add=nsroledn:(nsroledn=cn=JDC,dc=sesta,dc=com),d
el=nsroledn:(nsroledn=cn=JDC,dc=sesta,dc=com)")(version 3.0; acl "Allow JDCAdmin1 Role to
add/remove users to JDC Role"; allow (write)roledn="ldap:///cn=JDCAdmin1,dc=sesta,dc=com";)
-
# aci for JDCAdmin2 role
# This role can add/remove channels from the JDC role's display profile
add:aci
aci:
(target="ldap:///cn=SunPortalDesktopService,dc=sesta,dc=com")(targetfilter=(cn=cn=JDC,dc=se
sta,dc=com))(targetattr="*")(version 3.0; acl "Allow JDCAdmin2 to edit display profile of
JDC Role"; allow (all) roledn="ldap:///cn=JDCAdmin2,dc=sesta,dc=com";)
-
add:aci
aci: (target="ldap:///dc=sesta,dc=com")(targetattr = "*") (version 3.0; acl "Allow
JDCAdmin2 to read and search all"; allow (read,search) roledn =
"ldap:///cn=JDCAdmin2,dc=sesta,dc=com";)
```

2. Change directories to Sun Java Sytem Access Manager utilities directory. For example,

   cd */AccessManager-base*/**SUNWam/bin**

3. Set `LD_LIBRARY_PATH` to include `AccessManager-base/SUNWam/ldaplib/solaris/sparc/ldapsdk`

4. Execute the following command.

   ./ldapmodify -D "*DS_DIRMGR_DN***" -w** *DS_DIRMGR_PASSWORD* **-f /tmp/acis.ldif**

5. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

6. Navigate to the organization or suborganization to create a new role (such as `JDCAdmin1` and `JDCAdmin2`).

   a. Choose Roles from the View menu and click New.

   b. The New Role page appears in the data pane.

   c. Enter the role information (Name, Description, Role Type, Access Permissions) and click Create (for example, a static role JDC with "Type=Service" and "Access Permissions=No Permissions").

      The new role appears in the navigation pane.

7. Create "Desktop" service template for role you created.

   a. Choose Services from the View menu.

   b. Click the properties arrow next to the Desktop service.

   c. Accept or modify the default attribute values for the Desktop service and click Save.

8. Create a tab in the role display profile (for example, the role display profile for JDC).

   a. Navigate to the role where the tab will be created.

   a. Choose Services from the View menu in the navigation pane.

   b. Click the properties arrow next to Desktop in the navigation pane.

   c. The Desktop attributes page appears in the data pane.

d. In the Desktop page, click the Channel and Container Management link.

e. The Channels page appears, with the container path set at the root.

f. Click the Container that you want to add the channel or container to.

g. The top of the page displays the container path where the channel will be added. Defined channels and container, if any, appear in lists.

h. Click Add to add a container channel or channel.

i. To add a container channel, click Add under Container Channel. To add a channel, click Add under Channel.

j. The Add Channel page appears.

k. Type a channel name and select the type of provider from the menu.

l. Click Create.

Refer to Chapter 10, "Administering the Display Profile" for more information.

9. Create a user (such as `admin1` or `admin2`).

a. Navigate to the role where the user will be created.

b. Choose Users from the View menu and click New.

c. The New User page appears in the data pane.

d. Select the services to assign to the user and click Next.

e. Enter the user information and click Create.

f. The new user appears in the navigation pane.

10. Assign a role to a user (such as `JDCadmin1` to `admin1` or `JDCadmin2` to `admin2`).

a. Navigate to the organization or suborganization where the role will be assigned.

b. Choose Users from the View menu.

c. Click the properties arrow next to the user who will be assigned the role.

d. The user profile information appears in the data pane.

e. Click Roles from the View menu in the data pane.

f. The Add Roles page appears.

g. Check the box next to the roles to assign and click Save.

h. The Roles for this User box is updated with the assigned roles.

i. Click Save to save the changes.

11. Logout from the admin console.

To Define an ACI Using the Admin Console

1. Log in to the Sun Java Sytem Access Manager administration console as Top-level Admin.

   By default, Identity Management is selected in the location menu, and Organizations is selected in the navigation pane.

2. Click Service Configuration in the location pane.

3. Click the properties arrow next to the Administration service.

   The administration attributes appear in the data pane.

4. In the Default Role Permissions (ACIs) entry field, type in the ACI definition and click Add. For example, for the `JDCAdmin1 and JDCAdmin1` role defined previously, you would enter the following:

```
JDCAdmin1|Add/delete users from JDC role|dc=sesta,dc=com:aci:
(target= "ldap:///ou=people,dc=sesta,dc=com") (targetattr =
"*")(version 3.0; acl "Allow JDCAdmin1 Role to read and search
users"; allow (read,search) roledn =
"ldap:///cn=JDCAdmin1,dc=sesta,dc=com";)##dc=sesta,dc=com:aci:
(target="ldap:///dc=sesta,dc=com")
(targetfilter="(entrydn=cn=JDC,dc=sesta,dc=com)")(targetattr="*")(v
ersion 3.0; acl "Allow JDCAdmin1 Role to read and search JDC
Role";allow (read,search)
roledn="ldap:///cn=JDCAdmin1,dc=sesta,dc=com";)
##dc=sesta,dc=com:aci:(target="ldap:///ou=people,dc=sesta,dc=com")(
targetattr="nsroledn")(targetfilter="(!(|(nsroledn=cn=Top-level
Admin Role,dc=sesta,dc=com)(nsroledn=cn=Top-level Help Desk Admin
Role,dc=sesta,dc=com)(nsroledn=cn=Organization Admin
Role,dc=sesta,dc=com)(nsroledn=cn=Top-level Policy Admin
Role,dc=sesta,dc=com)))")(targattrfilters="add=nsroledn:(nsroledn=c
```

```
n=JDC,dc=sesta,dc=com),del=nsroledn:(nsroledn=cn=JDC,dc=sesta,dc=co
m)")(version 3.0; acl "Allow JDCAdmin1 Role to add/remove users to
JDC Role"; allow
(write)roledn="ldap:///cn=JDCAdmin1,dc=sesta,dc=com";)


JDCAdmin2|Add/remove channels from the JDC
role|dc=sesta,dc=com:aci:(target="ldap:///cn=SunPortalDesktopServic
e,dc=sesta,dc=com")(targetfilter=(cn=cn=JDC,dc=sesta,dc=com))(targe
tattr="*")(version 3.0; acl "Allow JDCAdmin2 to edit display profile
of JDC Role"; allow (all)
roledn="ldap:///cn=JDCAdmin2,dc=sesta,dc=com";)##dc=sesta,dc=com:ac
i: (target="ldap:///dc=sesta,dc=com")(targetattr = "*") (version
3.0; acl "Allow JDCAdmin2 to read and search all"; allow
(read,search) roledn = "ldap:///cn=JDCAdmin2,dc=sesta,dc=com";)
```

The new ACI appears in the Default Role Permissions (ACIs) list.

5. Click Save.

To Create a New Admin Role for the Delegation Model

Once you have created an ACI defining the permissions for a delegated administration role, you must create a role for using that ACI definition.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java Sytem Access Manager administration console as Top-level Admin or Organization Admin.

   By default, Identity Management is selected in the location menu, and Organizations is selected in the navigation pane.

2. Navigate to the organization or suborganization where the role will be created.

   All created organizations are displayed in the navigation pane.

NOTE If this is a new organization, you must add all the services and create the appropriate templates. See Chapter 6, "Administering Authentication, Users, and Services" for more information.

3. Choose Roles from the View menu and click New.

   The New Role page appears in the data pane.

4. Enter a name, select static role, and click Next.
5. Enter the description and choose Administrative as the type.
6. Select the Access Permissions:
   a. If you created the ACI definition for the role using the Administration Console, select the role you created from the Access Permissions list.
   b. If you created the ACI definition for the role using the command line, select No Permissions as the role name will not be listed in the Access Permissions list.
7. Click Create.

   The new role appears in the navigation pane.

To Assign a Role Administrator Role

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location menu, and Organizations is selected in the navigation pane.

2. Navigate to the organization or suborganization where the role was created.

   All created organizations are displayed in the navigation pane.

3. Choose Roles from the View menu.
4. Click the properties arrow for the role to assign.
5. Choose Users from the View menu in the data pane and click Add.

   The Add Users page appears in the data pane.

6. Specify the values for the fields to find the user to assign and click Filter.

   A list of users displays.

7. Check the box next to the users to which to assign the role or click Select All to choose all the users.

8. Click Submit.

   The list of users for this role box is updated with the assigned users.

To Configure Additional Restrictions on a Role Administrator Role

You can configure a role with a restricted set of capabilities. One common restriction you might want is a role with permissions to modify the display profile and perform content management functions, but that is restricted from viewing the rest of the Desktop attributes.

You can also set up delegated administrators with a start DN view. The start DN view is the directory location below which the delegated administrator can see and modify entities.

To configure additional restrictions on a role:

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location menu, and Organizations is selected in the navigation pane.

2. Navigate to the organization or suborganization where containing the role to configure.

   All created organizations are displayed in the navigation pane.

3. Choose Roles from the View menu.

4. Select the role to configure.

5. Select Services from the View menu.

6. To restrict the role to only display profile or channel management capabilities, do the following:

   a. Click the Edit link for the Desktop service.

b. Create a User service template at this role.

The Desktop page appears in the data pane.

c. Unselect the Show Desktop Attributes checkbox.

d. Specify a DN in Admin DN Starting V.

e. Click Save.

NOTE If the Show Desktop Attributes checkbox is unselected, when users with this role access the Desktop services, they will not be able to see the Desktop attributes; they will only see the Channel and Container Management link. In addition, they will only be able to see the channels and containers defined at the role level.

---

7. To restrict the role to a particular start DN, do the following:

a. Click the Edit link for the User service.

b. Create a User service template for the role.

The User page appears in the data pane.

c. Specify a DN in Admin DN Starting View. For example, `cn=JDC, dc=sesta, dc=com`.

d. Click Save.

Chapter 8

Administering the Portal Desktop Service

This chapter describes how to administer the Sun Java System Portal Server Desktop service.

This chapter contains these sections:

Overview of the Desktop

This section describes the Desktop component, its underlying structure, and how you administer it.

Desktop Glossary

Table 8-1 describes the pertinent Desktop terminology.

The first column of the table lists the term; the second column provides a definition of the term.

Table 8-1 Desktop Glossary

Term	Definition
Desktop	Provides the primary end user interface for Portal Server.
Provider	Adapts the interface of a generic resource for use by the Portal Server. A JSP provider compiles and executes a JSP file to generate a markup. An XML provider translates an XML file to generate a markup The portal server can also query the provider for information to display a markup on a portal page.
Portlet	Pluggable web components that process requests and generate content within the context of a portal. Portlets are managed by the Portlet Container (an implementation of the Portlet Specification as defined by the JSR168 Expert Group). Conceptually they are equivalent to the software Providers.
Channel	Displays content in the Desktop, usually arranged in rows and columns. At runtime, a channel consists of a provider object, configuration, and any data files (JSP, HTML templates, and so on) required to support the channel.
Container or Container Channel	A channel that primarily generates its content by including or aggregating the content of other channels (referred to as child channels).

Portal Desktop Architecture and Container Hierarchy

The Desktop is the primary end-user interface for Portal Server. It is implemented through a servlet and is supported by various APIs and utilities (for example, Sun Java™ System Access Manager APIs, resource bundles, properties files, back-end servers such as mail, and so on).

The Desktop provides a mechanism for extending and aggregating content through the Provider Application Programming Interface (PAPI). Content providers, or providers, enable container hierarchy and the basic building blocks for building some types of channels. Usually, channels are arranged in rows and columns, but they can also be displayed in some other arrangement, depending on the implementation of the container channels. The provider is the programmatic entity responsible for the generation of content, which is displayed in the channel. Generated content can consist of entire pages, frames, or channels; any markup.

As the amount of content on a portal increases, a containment method for referencing or referring to groups of content can facilitate the portal configuration, development, and end-user experience.The Portal Server provides a flexible, extensible set of container providers to aggregate content.

Figure 8-1 provides an example of the Desktop container hierarchy. In this figure, a Tab container is the top-level container. The Tab Container contains two Tab Channels, Tab 1 and Tab 2. Tab 2 is a Table Container and contains five channels.

Figure 8-1 Sample Portal Desktop Container Hierarchy

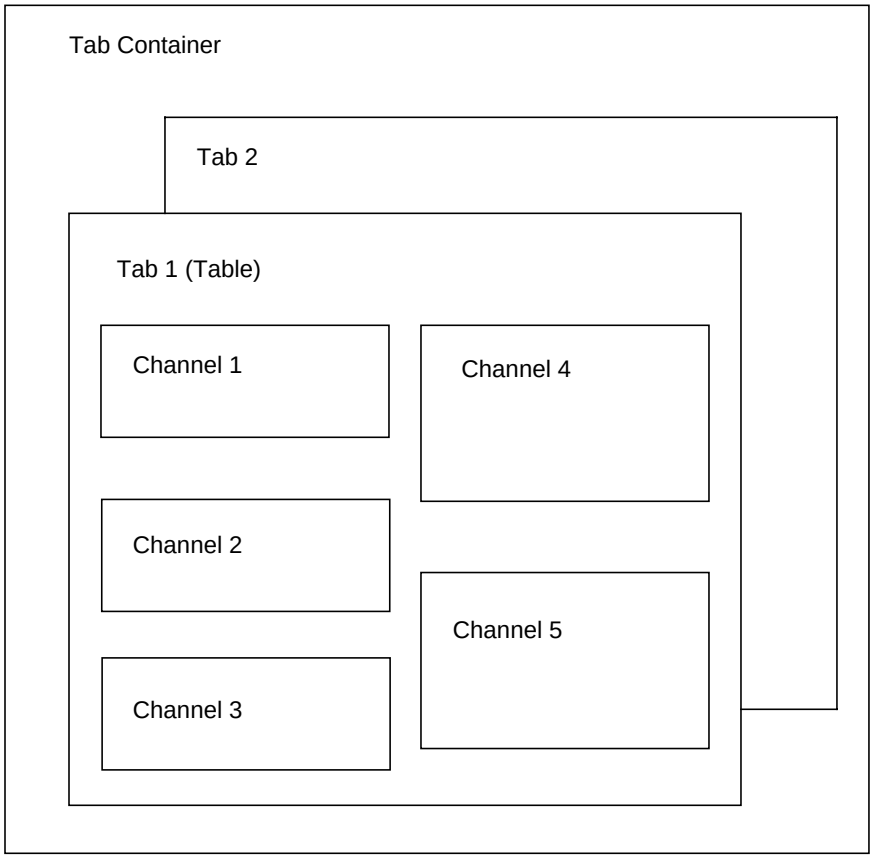


Figure 8-1 illustrates the following containment types:

- **Tab Container** - Contains any number of table, single or tab containers. This container also includes contains the banners, and menu bars for the portal as well.
- **Tab Channel** - Aggregates the output of other channels, providing a tabbed user interface to switch between them. Tab containers configuration are modified at runtime to vary which leaf channel is displayed.
- **Table Container** - Aggregates the content of other channels into rows and columns. This container functions much like the Portal Server 3.0 front provider. It can be thought of as a bucket for the content of other channels.

User Defined Channels

Each tab in a tab container includes a Content link. If you select the Content link, a page where a user can select the channels they would like to appear in the current tab's container is displayed. In this release, an additional link on the top right of this page, Create New Channel link, is included. The Create New Channel link, when selected, presents a page where a user can create a new channel. However, the channels that can created by the user is definable by the administrator.

To create a new channel (from the page shown in), the user must specify the information outlined in Table 8-2 on page 158 in the form presented.

Table 8-2 User Defined Channels

Form Field	Field Type	Field Description
Channel Name	Text field	Channel name may contain only letters (a-z,A-Z) and digits (0-9).
Channel Title	Text field	This is the title that will appear in the Channel titlebar.
Channel Description	Text field	This is the description for the Channel that appears on the Content link page.
Channel Type	Combo box	This is a list of Providers that new Channels can be created from.
Channel Category	Combo box	This is a list of the Categories for the Tab's Container.

Table 8-2 User Defined Channels

Form Field	Field Type	Field Description
Display Channel	Radio buttons with "Yes" and "No"	Select Yes for Display Channel so that the new Channel will automatically be displayed when the Browser is refreshed after selecting the Create button. Select No so that the Channel will not automatically be displayed when the Browser is refreshed after selecting the Create button. Instead, the channel can be displayed in the Browser by selecting the Channel from the Content link. In either case, once the new Channel is selected and displayed in the Browser, it is necessary to update its properties by selecting the Edit button which is available in the newly created Channel's titlebar.
Create	Button	Select Create to create the new Channel.
Cancel	Button	Select Cancel to return the user to their Desktop display.

The Delete A Channel link is displayed on the Content page after a user has created a user-defined channel. When a user clicks on the link, a list of all of the channels that the user created is displayed for possible deletion.

Portal Desktop Providers

Sun Java System Portal Server uses two types of providers:

- Building Block Providers—Extendable providers whose interfaces are public. These providers connects to a generic resource (like a JSP file). These providers can generate more than one channel in the Portal Desktop, thus the relationship between the provider and the channel is one to many.
- Content Providers—Non-extendable providers expects a specific set of data in order to render (for example, a bookmark provider expects a specific template and data). These kind of providers are not building block providers.

The Portal Desktop uses a *display profile* for storing content, provider, portlet, and channel data. See Chapter 10, "Administering the Display Profile" for more information.

Portal Desktop Service

The Desktop service uses Sun Java System Access Manager services to store application and user-specific attributes for each organization or suborganization. You then create a display profile policy and assign it to users. You also use the Sun Java System Access Manager administration console to modify Desktop attributes. See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for more information on desktop attributes.

Sample Desktops

Within the sample Desktops, Portal Server includes the following channels:

- Bookmarks
- Applications
- User Information
- Search
- Notes
- Mail Check
- Login
- Simple Web Service
- Simple Web Service Configurable
- Portlet Samples
- Collaboration

These channels are customized and configured for the sample portal. They may require the modification of the user interface before they are deployed.

Portal Desktop Customization

When deploying Portal Server, one of your major tasks will be to develop, or customize your own portal. You will create and extend providers, channels and container channels, deploy your own online help, come up with a look-and-feel, and so on. If desired, you can use the sample Desktops as a starting point in customizing your site's portal. See the *Sun Java System Portal Server 6 2005Q1 Desktop Customization Guide* for more information on customizing your portal.

Overview of Hot Deployment of Channels

Portal Server enables you to deploy providers and channels on a live system without performing a restart, hence the “hot deployment.” You can do so without interrupting user sessions.

The three technologies that facilitate hot deployment are:

- Provider class loader—Reloads providers and classes used by providers. For the provider class loader to function properly, all classes (or JAR files) must reside in a well-defined directory.
- Display profile refresh—Updates the in-memory Desktop configuration, that is, the display profile, if it has been changed by an external source such as the Sun Java System Access Manager administration console or the `dpadmin` command.
- Portal Desktop template and JSP reloading—Retrieves the appropriate template and JSP files for the Desktop type configured.

Overview of Provider Archives

The `par` utility enables you to package and transport channels, portlets, and providers, and all associated files, in and out of the Portal Server system. The channel, portlet, or provider is stored in the `.par` file format. Files included in the `.par` include:

- Display profile documents
- Class files
- Provider resource bundle files (property files)
- Templates and JSP files
- Static content files, that is, HTML and image files

Administering the Portal Desktop Service

The Desktop merges all of the documents in a user’s display profile merger set and uses the result to configure the user’s desktop. A display profile merger set consists of all the display profile documents associated with a user. Display profiles are defined at different levels in the Sun Java System Access Manager organization

tree. Display profile documents from the various levels of the tree are merged or combined to create the user's display profile. For example, the user's display profile document is merged with the role display profile documents (if any), the organization's display profile document, and the global display profile document to form the user's display profile.

The Desktop display profile and other configuration data are defined as service attributes of the Portal Desktop service under the Sun Java System Access Manager service management framework. When an organization adds for the Portal Desktop service from the Sun Java System Access Manager administration console, all users within the organization inherit the Portal Desktop service attributes in their user profiles. These attributes are queried by the Portal Desktop to determine how information will be aggregated and presented in the Portal Desktop.

By default, the Policy Configuration service is automatically added to the top-level organization. Suborganizations must add their policy services independently of their parent organization. Any policy service you create must be added to all organizations.

The following describes the high-level steps that you perform to configure the Portal Desktop service for users in an Sun Java System Access Manager organization:

1. Adding the Policy service for an organization.
2. Creating a referral policy for a peer or suborganization.
3. Creating a normal policy for a peer or suborganization.
4. Assigning a default redirect URL.
5. Customizing Desktop service attributes.

NOTE If you install the sample portal, the installer installs all the necessary display profile XML files for the sample. You can customize the profiles using the Sun Java System Access Manager console or the command-line interface. See Chapter 10, "Administering the Display Profile" for further information.

By default, the Policy Configuration service is automatically added to the top-level organization. Suborganizations must add their policy services independently of their parent organization. Any policy service you create must be added to all organization. The high-level steps to use policies are:

1. Adding the Policy service for an organization. (This will be done automatically for the organization specified at installation.) Suborganizations do not inherit their parent's services, so you need to add a suborganization's Policy service. See To Add a Policy Service for a Suborganization for information.
2. Creating a referral policy for a peer or suborganization. You can delegate an organization's policy definitions and decisions to another organization. (Alternately, policy decisions for a resource are delegated to other policy products.) A referral policy controls this policy delegation for both policy creation and evaluation. It consists of a rule and the referral itself. If the policy service contains actions that do not require resources, referral policies cannot be created for suborganizations. See To Create a Referral Policy for a Suborganization for information.
3. Creating a normal policy for a peer or suborganization. You create a normal policy to define access permissions. A normal policy can consist of multiple rules, subjects, and conditions. See To Create a Normal Policy for a Suborganization for information.

To Add a Policy Service for a Suborganization

Suborganizations do not inherit their parent's services, so you need to add a suborganization's Policy service.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose the organization for which you would like to add the Desktop service.
3. Choose Services from the View menu in the navigation pane.
4. Click Add in the navigation pane.

   A list of available services displays in the data pane.

5. Select the check box for Portal Desktop under Portal Server Configuration and click OK.

   The Navigation pane is updated with the added Desktop service under Portal Server Configuration.

6. Choose Services from the View menu in the navigation pane.

7. Click the properties arrow next to Desktop in the navigation pane.

8. A question is displayed in a message box in the data pane to confirm if a service template should be created for the Desktop service. Click Create in the message box to create the template.

9. After the page is submitted and the template created, the data pane displays a list of Desktop service attributes and their default values, if any. Modify the values as needed. When done, click Save to store the final values in the service template.

   The display profile of a newly created service template takes on the value entered in the Dynamic section of the Portal Desktop service under Service Management. If those values were blank, the display profile in this new template is also blank.

NOTE The default value for the Conflict Resolution Interval attribute is "Highest." Setting up service templates at different levels (for example, organization and role) with the same priority for a added service could lead to unexpected results.

To Create a Referral Policy for a Suborganization

You can delegate an organization's policy definitions and decisions to another organization. A referral policy controls this policy delegation for both policy creation and evaluation. It consists of a rule and the referral itself. The referral must define the parent organization as the resource in the rule, and it must contain a SubOrgReferral with the name of the organization as the value in the referral.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Select Identity Management from the navigation pane.

3. Select Policies from the View menu.

4. Click New to create new policy.

   The Create Policy page appears in the data pane.

5. For Name, type SubOrgReferral_Desktop. Make sure you select Referral in Type of Policy. Then click Create.

6. Select Desktop in Service and click Next

7. Click Rules from the View menu in the data pane and click New. Make sure Portal Desktop is selected and click Next.

   The New Rule template appears in the data pane.

8. Enter DesktopRule in Rule Name and click Create.

9. Click Referrals from the View menu in the data pane and click New.

   The New Referral template appears in the data pane.

10. Enter SubOrgReferral_Desktop in Name.

    Make sure that the name of the suborganization is selected for Value in the data pane and click Create to complete the policy's configuration.

11. Click Save in the data pane.

    The message "The policy properties have been saved" is displayed when the data is saved.

To Create a Normal Policy for a Suborganization

You create a normal policy to define access permissions. A normal policy can consist of multiple rules, subjects, and conditions.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Navigate to the organization or suborganization that you want to assign a policy.

   All created organizations are displayed in the navigation pane.

3. Choose Policies from the View menu.

   The policies for that organization are displayed.

4. Select New in the navigation pane. The New Policy page opens in the data pane.

5. Enter SubOrgNormal_Desktop in Name. Make sure you select Normal in Type of Policy. Click Create

6. Choose Rules from the View menu in the data pane and click New. The New Rule page opens in the data pane

7. Select Portal Desktop from the Service menu and click Next. Enter DesktopRule in Rule Name. Make sure Has Privilege to Execute NetMail is checked

8. Select Portal Desktop from the Service menu and click Next. Make sure Has Privilege to Execute NetMail is checked.

9. Select the type of subject from the Type menu and click Next to complete subject configuration.

10. Choose Subjects from the View menu in the data pane and click New. The New Subject page opens in the data pane.

11. Click Create to complete the policy configuration.

    The message “The policy properties have been saved.” is displayed when the data is saved.

To Redirect Successful Login User to the Portal Desktop URL

By default, users in an organization receive the Desktop service attributes and values after successfully logging in. These values are queried by the Desktop servlet to determine the Portal Desktop contents of any users in the organization. To instruct Sun Java System Access Manager to invoke the Portal Desktop servlet automatically after a user has successfully logged in, you can change the value of the Default Redirect URL to the Portal Desktop URL.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

To set the default redirect for a specific organization to the Portal Desktop URL:

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Select the organization for which you want to set the Portal Desktop URL.
3. Choose Services from the View menu.
4. Click the properties arrow next to Core in the navigation pane.
5. In the data pane, search for an attribute named Default Success Login URL.
6. Set the value of the User's Default Redirect URL to the URL for the Portal Desktop servlet, for example, `/portal/dt` is the URL for the sample Desktop.
7. Click Save.
8. Verify the default redirect URL by logging in to the Portal Desktop.

To Redirect Successful Login User to the Portal Desktop URL (Global)

The values applied to the global attributes are applied across the Sun Java System Access Manager configuration and will be inherited by every newly created organization.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

To set the Default Redirect URL to the Portal Desktop URL globally:

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Core in the navigation pane.
4. In the data pane, search for an attribute named Default Success Login URL.
5. Set the value of the Default Redirect URL to the URL for the Portal Desktop Servlet, for example, `/portal/dt`.
6. Click Save.

To Modify the Values of Portal Desktop Service Attributes

You can customize the Portal Desktop service by modifying its service attributes.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Select the organization for which you want to modify the Desktop attributes.

3. Click the properties arrow next to Desktop in the navigation pane.

   A list of Portal Desktop service attributes, including the display profile XML, is displayed in the data pane.

4. Modify the service attribute values.

   See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for information on the desktop attributes.

5. When done, click Save.

   The changes will affect only users in this particular suborganization or role.

To Modify the Values of Portal Desktop Service Attributes (Global)

Occasionally, you need to modify the global Desktop service attribute values that affect all organizations that want to add for the Desktop service in the future.

The values applied to the global attributes are applied across the Sun Java System Access Manager configuration and are inherited by every configured organization.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Management in the location pane.

3. Click the properties arrow next to Desktop in the navigation pane.

   A list of global Desktop service attributes, including the display profile XML, is displayed in the data pane.

4. Modify the service attribute values.

   See *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for information on the desktop attributes.

5. When done, click Save.

   The changes affect all organizations that add the Desktop service in the future.

To Access the Sample Portal Desktop

1. Log out from the Sun Java System Access Manager administration console.
2. Log on with a user account (not the `amadmin` user) using the following URL:

   `http://`*`server:port`***`/portal/dt`**

   If you need to create a user account, see Chapter 6, “Administering Authentication, Users, and Services” on page 85 for information.

To Examine the Desktop Logs

Portal Desktop errors on the are logged to debug log files. By default, the location of these log files is as follows.

- `/var/opt/SUNWam/debug/desktop.debug`
- `/var/opt/SUNWam/debug/desktop.dpadmin.debug`

Examine these log files for errors. An example follows. This error indicates that an unauthenticated user attempted to execute the Portal Desktop.

```
06/20/2002 02:36:30:600 PM PDT: Thread[Thread-177,5,main]
ERROR: DesktopServlet.handleException()
com.sun.portal.desktop.DesktopException: DesktopServlet.doGetPost(): no
privilige to execute desktop
        at
com.sun.portal.desktop.DesktopServlet.doGetPost(DesktopServlet.java:456)
        at
com.sun.portal.desktop.DesktopServlet.service(DesktopServlet.java:303)
        at javax.servlet.http.HttpServlet.service(HttpServlet.java:853)
        at
com.sun.server.http.servlet.NSServletRunner.invokeServletService(NSServletR
unner.java:897)
        at
com.sun.server.http.servlet.WebApplication.service(WebApplication.java:1065
)
        at
com.sun.server.http.servlet.NSServletRunner.ServiceWebApp(NSServletRunner.j
ava:959)
```

Administering Portlets

Portlets are administered from the Sun Java™ System Access Manager administration console. The administration console includes pages for creating portlet channels from portlets and changing preferences of portlet channels. The `pdeploy` is a command line tool that can be used to deploy and undeploy the portlet web application into a web container (see the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for more information on administering command line utilities).

NOTE If a client request accesses a portal page which contains at least one session-enabled portlet, it is strongly recommended that all the portlets on that portal page should be packaged within a single portlet application, otherwise the resulting behavior of the session creation may be nondeterministic.

To Create a Channel from a Portlet

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Manager administration console and select your organization.
2. Select Services under Show in the navigation menu.
3. Select the Desktop service from Portal Server Configuration.
4. Select Channel and Container Management link.
5. Select the Add Portlet Channel button under Channels.

   The page to create a portlet channel is displayed.
6. Specify in the Add Channel page,
   - The channel name.
   - Note that channel names can contain only letters (A through Z) and digits (0 through 9) and it is a required field.
   - The Portlet

   Only contains portlets that are deployed in the system are displayed.
7. Select the Create button to create the portlet channel.

To Create a Channel from a Portlet for a Specific Container

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Manager administration console and select your organization.

2. Select Services under Show in the navigation menu.
3. Select the Desktop service under Portal Server Configuration.
4. Select Channel and Container Management.
5. Select the link to the Container where you wish to create a portlet channel.

   The page for managing the container is displayed.
6. Select the Add Portlet Channel button under Channels.

   The page for creating and adding a portlet channel is displayed.
7. Specify, in the Add Channel page:
   - A name for the channel.
   - The Portlet from the pull-down list. The list only contains portlets that are deployed in the system.
8. Whether the channel will be available to end-users or whether it will be available and visible on the Desktop by selecting the appropriate radio button.
9. Select the OK button.

   Note that the channel is added to the list of channels under Channels and under Visible on the Portal Desktop in the Container Management page.

To Add the Portlet Channel to a Container

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console and select your organization.
2. Select Services under Show in the navigation pane.
3. Select Desktop from Portal Server Configuration.
4. Select Channel and Container Management.

5. Select the link to the Container to which you wish to add the newly created portlet channel.

   The page for managing the container is displayed.

6. Select the portlet channel you wish to add from the Ready for Use list.

   Click the Add button located over the Visible on the Portal Desktop.

   This will add the selected portlet channel to the list of channels visible on the selected container.

7. Select Save button under Channel Management to save the new settings.

To Edit a Portlet Channel Preferences and Properties

The portlet preferences are defined in `portlet.xml`

```
<portlet-preferences>
           <preference>
               <name>foo</name>
               <value>apple</value>
           </preference>
           <preference>
               <name>bar</name>
               <value>orange</value>
               <value>grape</value>
               <read-only>true</read-only>
           </preference>
       </portlet-preferences>
```

is mapped to the following display profile:

```
<Collection name="__Portlet__AdditionalPreferences"/>
       <Collection name="__Portlet__PreferenceProperties">
           <Collection name="default">
               <String name="foo" value="|apple"/>
```

```
            <String name="bar" value="|orange|grape"/>
        </Collection>
        <Collection name="isReadOnly">
            <Boolean name="foo" value="false"/>
            <Boolean name="bar" value="true"/>
        </Collection>
    </Collection>
    <String name="__Portlet__foo" value="|apple"/>
    <String name="__Portlet__bar" value="|orange|grape"/>
```

There is an empty collection `__Portlet__AdditionalPreferences` created to hold the preferences added during runtime. The collection `__Portlet__PreferenceProperties` contains two collections, `default` and `isReadOnly`. The `default` collection stores the default values as defined in `portlet.xml`. Similar to the `default` collection, the `isReadOnly` collection stores the read-only flags of the preferences using Boolean properties.

Each preference in the `portlet.xml` has one corresponding `String` property in the `default` collection with the preference name as the property name. The value of the `String` property is to represent the default value defined in `portlet.xml` prepended and delimited by the character "|". Each preference is then represented by a `String` property which stores the current value of the preference. The name of the property is the name of the preference prepended by the string `__Portlet__`. The value of the property is the current preference values prepended and delimited by the character "|".

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

---

1. Log in to the Sun Java System Access Manager administration console and select your domain.
2. Select Services under Show in the navigation pane.
3. Select Desktop from Portal Server Configuration.
4. Select Edit link for the portlet channel you wish to edit.
5. The Edit Channel page is displayed. The channel edit page displays the portlet preferences for the portlet entity.

6. Modify the preferences and select Save to save the modifications.
7. To modify the default values of the preferences, select Edit link for the preference you wish to edit. Properties can be edited in the Edit Channel page.

Administering par Files

The par utility enables you to transfer or move providers or channels from one Portal Server to another. The par utility creates a specialized packaging mechanism called a .par file for transport of channels, portlets, and providers into and out of the server. A .par file is an extended form of the .jar file format, with added manifest information to carry the deployment information and an XML document intended for integration into the Portal Server display profile on the target server.

The par command line utility is used to create, modify, and deploy par files. The export subcommand allows you to create or modify a par file. The import subcommand allows you to import or deploy the provider, channel, or portlet on an Portal Server. The describe subcommand describes the contents of a par file. See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for detailed information on the syntax of the par command.

To use the par utility, you must be logged in as superuser to the Portal Server on which the files you want to export or import are resident. When you export you need to be sure to export all the required files for the channel, portlet, or provider. For example, with channels you must include the static content files and with providers you must include all the class files used by the provider. Because specifying all the data to be included in the par file on the command line can be cumbersome, a simple text file with lines indicating the data is created and this “export file” is called by the par utility. See Chapter 8, “Administering the Portal Desktop Service” for further information.

To Create a New par File

To create a new par file to export a channel, portlet, or provider:

1. Log in to the Portal Server from which to export the channel, portlet, or provider.
2. Change directories to the directory where the script is installed. That is:

   cd *PortalServer-base*/**SUNWps/bin**

3. At the command line, enter the `par export` command and subcommand and include the following arguments: the name of the par file to create, a directory server name argument corresponding to the desired display profile document to export, and any number of (requires at least one) export files or `from` specifications. For example, to export the channel mychannel from `o=sesta.com,o=isp` to the `mychannel.par` file, enter

   `./par export mychannel.par "o=sesta.com,o=isp" from: channel mychannel`

   See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for more syntax information.

To Modify an Existing par File

To modify an existing par file to export a channel, portlet, or provider:

1. Log in to the Portal Server from which to export the channel, portlet, or provider.
2. Change directories to the directory where the script is installed. That is:

   `cd` *PortalServer-base*`/SUNWps/bin`

3. At the command line, `par export` command and subcommand with the `modify` option and include the following arguments: the name of the par file to modify, a directory server name argument corresponding to the desired display profile document to export, and any number of (requires at least one) export tiles or `from` specifications. For example, to modify the `mychannel.par` file to include the static content file `/mycontent.html`, enter

   `./par export --modify mychannel.par "dc=sesta,dc=com" “from= file /mycontent.html”`

To Deploy par Files

To import a par file to a Portal Server to deploy a provider or channel on the system:

1. Copy the par file for the provider or channel to import to the Portal Server on which to deploy the provider or channel.
1. Log in to the Portal Server on which to import the channel, portlet, or provider.

2. Change directories to the directory where the script is installed. That is:

   cd *PortalServer-base***/SUNWps/bin**

3. At the command line, `par import` command and subcommand and include the following arguments: the name of the par file to import, a directory server name argument corresponding to the desired display profile document to export, For example, to import the `mychannel.par` file, enter

   ```
   ./par import --auto myfile.par "do=sesta,dc=com"
   ```

Chapter 9

Administering the Web Services for Remote Portlets (WSRP) Service

This chapter describes how to administer the Sun Java™ System Portal Server Web Services for Remote Portlets (WSRP) service.

This chapter contains these sections:

Overview of the WSRP Standard

WSRP 1.0 is an OASIS standard that simplifies integration of remote applications and content into portals. The WSRP standard defines presentation-oriented, interactive web services with a common, well-defined interface and protocol for processing user interactions and for providing presentation fragments suited for mediation and aggregation by portals as well as conventions for publishing, finding and binding such services.

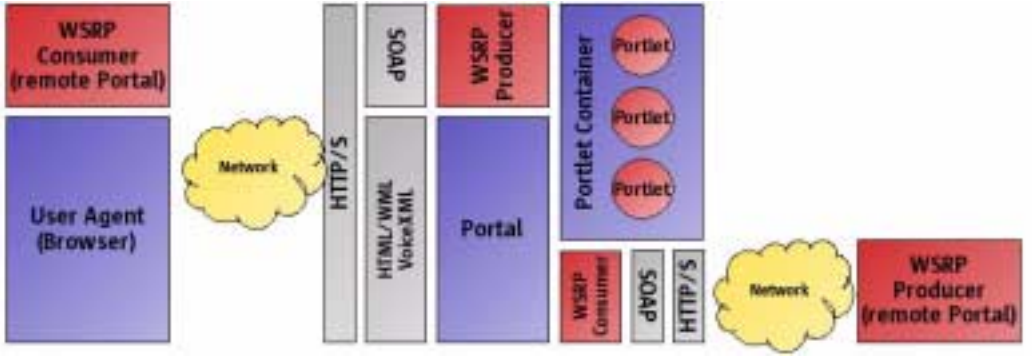
Because the WSRP interfaces are common and well-defined, all web services that implement the WSRP standard plug into all WSRP compliant portals – a single, service-independent adapter on the portal side is sufficient to integrate any WSRP service. As a result, WSRP becomes the means for content and application providers to provide their services to organizations running portals with no programming effort required.

See the WSRP 1.0 standard for more information:

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsrp

The implementation of the WSRP 1.0 standard in Portal Server includes both the WSRP consumer and the WSRP producer. The WSRP producer implementation supports publishing JSR 168 portlets for use by a remote WSRP consumer (see Administering Portlets for more information on JSR 168 portlets).The JSR 168 portlets are deployed locally on a portal server. These portlets can be published by an instance of the WSRP producer. Another portal server, through its WSRP consumer can subscribe to these remote portlets. While local portlets can be expected to provide a large part of the base functionality for portals, remote portlets allow the potential to bind to a variety of remote portlets without any installation effort or code running locally on the consuming portal server.

Figure 9-1 Web Services for Remote Portlets



Administering the WSRP Producer

This section describes the tasks to administer a WSRP producer:

- To Add a WSRP Producer Instance
- To Edit a WSRP Producer Instance
- To Add a WSRP Consumer Registration
- To Edit a WSRP Consumer Registration
- To Disable all WSRP Producers

To Add a WSRP Producer Instance

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the desired organization or suborganization and select Services from the View menu.

   All created organizations are displayed in the navigation pane.

3. Click the properties arrow next to the WSRP Producer under Remote Portlets (WSRP).

   The WSRP Producer attributes appear in the data pane.

NOTE Depending on your requirements and configuration, you may have the ability to create and enable multiple WSRP producers under the same organization or suborganization (or even different organizations or suborganizations). The constraint is that the name of the WSRP producer must be unique for the entire Portal Server system.

4. Click New under Producer to add a producer instance.

   The Specify Basic Parameters page appears.

5. Enter values for the following properties:

   - Name of the producer instance (must be unique for the entire Portal Server)
   - Registration is Required

     When registration is required, all WSRP consumers must register with this producer instance before making requests. Requests from unregistered consumers will be denied.

6. If you select Registration is Required, then the Specify Registration Properties appears. Specify whether:

   - inband registration is supported (when a WSRP consumer can register with the producer instance online).

     Inband registration allows WSRP consumers to register programmatically. Otherwise, out-of-band registration is required with manual contact (such as e-mail or telephone) between the WSRP consumer administrator and the WSRP producer administrator to set up and exchange access to a registration handle.

   - registration property descriptions that a WSRP consumer will provide during registration.

     Registration properties WSRP consumers must provide at the time that they register.

NOTE A registration validator is a Java class that implements the Registration Validator interface. Depending on your needs, you can author your own registration validator class and apply any logic necessary to validate consumer registrations.

7. Click Next.

   The Review Page page appears.

8. Verify the information that you provided and click Finish.

To Edit a WSRP Producer Instance

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the desired organization or suborganization and select Services from the View menu.

   All created organizations are displayed in the navigation pane.

3. Click the properties arrow next to the WSRP Producer under Remote Portlets (WSRP).

   The WSRP Producer attributes appear in the data pane.

4. Click the Edit Properties link beside the producer instance to be modified.

5. Select the General Properties tab and modify the properties as needed:

   - Status of the producer instance—Enabled for an active WSRP producer, Disabled for an inactive producer instance).

     A disabled producer instance will not accept requests from WSRP consumers. By default, a WSRP producer is disabled until one or more portlets are published.

   - Registration is Required—Checked if WSRP consumers need to register.

   - Inband Registration—Supported if a WSRP consumer can register with the producer instance programmatically, Unsupported if the only way a WSRP consumer can register with a WSRP producer is by manual means (out of band).

     Inband registration allows the WSRP producer to be contacted and passed registration data programmatically, and the WSRP consumer receiving a registration handle from the WSRP producer. Out-of-band registration requires manual contact (such as e-mail or telephone) between the WSRP consumer administrator and the WSRP producer administrator to set up and exchange access to a registration handle.

   - Registration Validator Class—Java class used to validate a registration. When a consumer registers with a producer, it passes registration property values for each registration property that the producer has defined in its service description. The registration validator class is used by the producer to validate that the values sent by the consumer are acceptable for this producer instance.

TIP A registrator validator is a Java class that implements the Registrator Validator interface. Depending on your needs, you can author your own registrator validator class and apply any logic necessary to validate consumer registrations.

6. Click Save
7. Select the Portlets Tab to modify the lists of deployed portlets that will be available to WSRP consumers, then click Save.
8. If registration is supported, select the Registration Properties tab to add or delete a property that all WSRP consumers must provide to register with the producer instance, then click Save.

To Add a WSRP Consumer Registration

A consumer registration allows a WSRP consumer to describe its capabilities to a WSRP producer. A WSRP consumer is being added out of band (such as by e-mail or telephone). The information entered when adding a consumer registration must match the capabilities of the WSRP consumer that is given the registration handle. Consumer registrations allow a WSRP producer to scope artifacts (such as portlet preferences) that are created by a WSRP consumer on the WSRP producer.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
2. Navigate to the desired organization or suborganization and select Services from the View menu.

   All created organizations are displayed in the navigation pane.
3. Click the properties arrow next to the WSRP Producer under Remote Portlets (WSRP).

   The WSRP Producer attributes appear in the data pane.
4. Click the Edit Properties link beside the WSRP producer to be modified.
5. Select the Consumer Registrations tab.
6. Click New under Consumer Registrations.

   The Create New Consumer Registrations page appears.

7. Specify basic properties as needed:
   - Name of the WSRP consumer.
   - Status of the consumer registration (Enabled for an active consumer registration, Disabled for an inactive consumer registration). If a consumer registration is inactive, the producer will not accept requests that include the registration handle for the disabled consumer registration.
   - Consumer Agent (an identifier for the application name and version).

     The format of the identifier is *productName.majorVersion.minorVersion* such as **`Sun Java(tm) System Portal Server.6.3`**
   - Method= “get” (Supported if the WSRP consumer has implemented portlet URLs in a manner that supports HTML markup containing forms with method=get, otherwise Not Supported).
8. Click Next.

   The Specify Registration Parameters page appears if the WSRP producer is configured to require registration.
9. Specify values for the registration properties.
10. Click Next.

    The Review Page page appears.
11. Verify the information that you provided and click Finish.

To Edit a WSRP Consumer Registration

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the desired organization or suborganization and select Services from the View menu.

   All created organizations are displayed in the navigation pane.

3. Click the properties arrow next to the WSRP Producer under Remote Portlets (WSRP).

   The WSRP Producer attributes appear in the data pane.

4. Click the Edit Properties link beside the WSRP producer to be modified.

5. Select the Consumer Registrations tab.

6. Click the Edit Properties link beside consumer registration to be modified.

   The Edit Consumer Registrations page appears.

7. Modify the properties as needed:

   - Name of the WSRP consumer.
   - Status of the consumer registration (Enabled for an active consumer registration, Disabled for an inactive consumer registration). If a consumer registration is inactive, the producer will not accept requests that include the registration handle for the disabled consumer registration.
   - Consumer Agent (an identifier for the application name and version).

     The format of the identifier is *productName.majorVersion.minorVersion* such as **`Sun Java(tm) System Portal Server.6.3`**
   - Method= “get” (Supported if the WSRP consumer has implemented portlet URLs in a manner that supports HTML markup containing forms with method=get, otherwise Not Supported).
   - Consumer Modes (An array of modes that the WSRP consumer is willing manage. See the WSRP 1.0 specification for more information).
   - Consumer Window States (An array of window states that the WSRP consumer is willing manage. See the WSRP 1.0 specification for more information).
   - Consumer User Scopes (The values that the WSRP consumer is willing to process for user context. See the WSRP 1.0 specification for more information).
   - Custom User Profile Data (An array of strings that name user profile extensions. See the WSRP 1.0 specification for more information).

- Registration Properties (List of registration properties. See the WSRP 1.0 specification for more information).

8. Click Save.

To Disable all WSRP Producers

1. Select Service Configuration in the location pane.
2. Click the properties arrow next to the WSRP Producers under Remote Portlets (WSRP).

   The WSRP Producers attributes appear in the data pane.
3. Select Disable All WSRP Producers under Global.
4. Click Save

Administering the WSRP Consumer

This section describes the tasks to administer the WSRP Consumer:

To Create a Remote Portlet Channel

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the desired organization or suborganization and select Services from the View menu.

   All created organizations are displayed in the navigation pane.

3. Click the properties arrow next to the Portal Desktop service.

   The Portal Desktop attributes appear in the data pane.

4. Click the Channel and Container Management link.

   The Channels page appears. At the top is the container path. The defined channels appear in a list.

5. Click New Remote (WSRP) Channel to add a remote channel.

   The New Channel page appears.

6. Select a WSRP producer from the Producer list.

7. Select a remote portlet from the Portlet list.

8. Type the name of the remote channel in Channel Name.

9. Click OK.

To Edit General Properties of the WSRP Consumer

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the desired organization or suborganization and select Services from the View menu.

   All created organizations are displayed in the navigation pane.

3. Click the properties arrow next to the WSRP Consumer under Remote Portlets (WSRP).

   The WSRP Consumer attributes appear in the data pane.

4. Click the General tab.

5. Modify the properties as needed:

   - Name—The WSRP consumer sends the consumer name to producers during registration.

6. Click Save

To Add a Configured WSRP Producer

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the desired organization or suborganization and select Services from the View menu.

   All created organizations are displayed in the navigation pane.

3. Click the properties arrow next to the WSRP Consumer under Remote Portlets (WSRP).

   The WSRP Consumer attributes appear in the data pane.

4. Click the Configured Producers tab.

5. Click New under Producer to add a configured WSRP Producer.

   The Specify Basic Parameters page appears.

6. Specify the name of the new configured WSRP producer and the URL for the WSDL of the desired WSRP producer.
7. Click Next.
8. If the desired WSRP producer requires registration and supports inband registration, specify whether the WSRP consumer will:
   - register programmatically with the desired WSRP producer online
   - provide a registration handle that is obtained by manual means from the desired WSRP producer

   Inband registration allows the WSRP producer to be contacted and passed registration data programmatically, and the WSRP consumer receiving a registration handle from the WSRP producer. Out-of-band registration requires manual contact (such as e-mail or telephone) between the WSRP consumer administrator and the WSRP producer administrator to set up and exchange access to a registration handle. If registration is required, but inband registration is not available, the registration handle that was obtained out of band would need to be entered.
9. Click Next.
10. Provide the information requested in Registration Properties.

    A WSRP producer may require that WSRP consumer who is registering return values for keys (or registration properties that the producer describes in its service description such as company name or postal code). If a producer defines registration properties, they will appear on this page and the values should be supplied.
11. Click Next.

    The Review Page page appears.
12. Verify the information that you provided and click Finish.

NOTE You may need to edit the configured WSRP producer you added to set up the user categories to roles mapping. Otherwise, some portlets may not work properly.

To Edit a Configured WSRP Producer

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the desired organization or suborganization and select Services from the View menu.

   All created organizations are displayed in the navigation pane.

3. Click the properties arrow next to the WSRP Consumer under Remote Portlets (WSRP).

   The WSRP Producer attributes appear in the data pane.

4. Click the Configured Producers tab.

5. Click the Edit Properties link beside the configured WSRP producer to be modified.

   The Edit Basic Properties page appears.

6. Modify the properties as needed:

   - Name of the WSRP producer
   - Status of the WSRP producer—Enabled for an active WSRP producer, Disabled for an inactive WSRP producer.

     Remote portlet channels that target a disabled producer will not be visible on the Portal Desktop. Exercise caution when disabling WSRP producers.

   - Service Description—Click Update Service Description to get the latest service description of the WSRP producer.

7. Click Save

8. If displayed, click the User Categories to Roles Mapping link.

   The User Categories to Roles Mapping page appears.

9. If displayed, assign a WSRP user category to a local Access Manager role.

NOTE Not every WSRP producer supports WSRP user categories.

10. Click Save.
11. If displayed, click the Registration Properties link.

    The Registration Properties page appears.
12. Modify the values of the registration properties as needed.
13. Click Save.

To Disable all WSRP Consumers

1. Select Service Configuration in the location pane.
2. Click the properties arrow next to the WSRP Consumer under Remote Portlets (WSRP).

   The WSRP Consumer attributes appear in the data pane.
3. Select Disable All WSRP Consumers under Global.
4. Click Save.

To Edit the Standard User Profile Mapping

Portal Server maintains a standard set of end user attributes as a way to personalize behavior for the current user.

1. Select Service Configuration in the location pane.
2. Click the properties arrow next to the WSRP Consumer under Remote Portlets (WSRP).

   The WSRP Consumer attributes appear in the data pane.
3. Modify the Standard User Profile Mapping value under Global using the following format:

   WSRP-defined profile|*LDAP-attribute*

   See the WSRP 1.0 specification for more information.

4. Click Save.

To Specify the Consumer Name

The WSRP consumer sends the consumer name to producers during registration. The value specified for the consumer name is used as the default unless a value is specified for consumer name at the organization or suborganization level.

1. Select Service Configuration in the location pane.
2. Click the properties arrow next to the WSRP Consumer under Remote Portlets (WSRP).

   The WSRP Consumer attributes appear in the data pane.
3. Enter a value in Consumer Name.
4. Click Save

Chapter 10

Administering the Display Profile

This chapter describes the Sun Java™ System Portal Server display profile component and how to administer it.

This chapter contains these sections:

Overview of Display Profile

This section describes the display profile component of Portal Server.

The display profile creates the display configuration for the Desktop by defining the following three items:

- **Provider definition—**Specifies the name and the Java™ class for the provider. A provider is a template used to generate content, which is displayed in the channel. See Provider Object for more information.
- **Channel definition—**Specifies the run-time configuration of an instance of the provider class. A channel is a unit of content, usually (but not necessarily) arranged in rows and columns. You can also have channels of channels, that is, container channels.

- **Provider and channel property definitions—**Specify the values for provider and channel properties. Properties defined in a provider usually specify default values for the channels that are derived from the provider. The display configurations for the channels include properties such as the title, description, channel width, and so on. The properties defined in the channel usually specify the specific value for that channel that is different from the default value.

  Container properties define the display definition about how to display the contained channels in the container, including: the layout of the container (thin-wide, wide-thin, or thin-wide-thin); a list of the contained channels; the position of the channel (the row and column number); and the window state of the contained channels (minimized or detached).

NOTE The display profile does not actually define the overall layout or organization of what users see on their Desktops. The display profile exists only to provide property values for channels. However, the display profile does indirectly control some aspects of channel presentation, such as column layout for a table container or how the table container draws channels in a table.

The display profile determines the layout in that channel properties determine layout. For example, the display profile for the sample portal's table provider definition contains the following statement:

```
<Integer name="layout" value="1"/>
```

This refers to thin-thick columns. However, there is nothing here in the structure of the display profile regarding actual layout.

The display profile does not control such things as how `XMLProvider` parses XML, it only has a definition of the kind of rules (XSL file) that are in it.

The Portal Desktop implements a display profile data storage mechanism on top of the Sun Java™ System Access Manager service for storing content provider and channel data. In addition, properties are set for the channels and providers.

The user's display profile is a series of XML documents describing container management and properties for channels. (One display profile document is equivalent to one XML document.) The display profile documents are stored in their entirety as a single attribute in the Sun Java System Access Manager services layer. That is, the display profile documents are an LDAP attribute residing in an instance of Sun Java™ System Directory Server.

To change display profile property values, the providers use the provider APIs (PAPI) to get and set the values. When the channel values are set to the display profile, the PAPI internal implementation uses the Access Manager SDK to set the display profile document in the Sun Java System Access Manager Desktop service attribute.

CAUTION Though possible, you should not edit the display profile using the Sun Java System Access Manager SDK.

Display Profile and the Administration Console

You can edit the display profile and other Portal Desktop service data through the Sun Java System Access Manager administration console and the `dpadmin` command. When you edit the display profile, you add, modify, and remove providers, containers, and channels, and edit properties. The Upload XML and Download XML links allow you to upload and download the display profile document. In addition, the Sun Java System Access Manager administration console provides an Channel and Container Management link in the Portal Desktop attributes page to add channels and containers and edit existing properties. The Channel and Container Management link enables you to define properties when a new channel or container is created. You can also use the Channel and Container Management link to add, modify, and remove channels and containers. See Administering the Display Profile for more information.

NOTE As the Channel and Container Management link enables access to only a portion of the display profile, it is envisioned that delegated administrators will use it. See Chapter 7, “Configuring Delegated Administration” for more information on how to configure delegated administrators.

Display Profile Document Structure

This section describes the overall structure of the display profile documents. The underlying data format for a display profile document is XML. See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for information on the display profile DTD syntax.

The display profile format establishes the Desktop's display configuration by defining provider and channel objects and their properties. The display profile is stored in the Sun Java System Directory Server at the `isp` level (or the top most directory node), the organization level, the role level, or the user level. At run time, a user's display profile is a result of "merging" all the display profile documents from the user's specific profile in the directory tree, and the value of a specific display profile object for that user is decided by the "merge" semantics of the display profile.

The display profile objects map directly to the XML tag that defines them. For example, the `<Channel name> </Channel>` XML tags define a channel object.

In general, the document structure of the display profile resembles the following:

```
<DisplayProfile>
 <Properties>...global properties...</Properties>
 <Channels>...channel definitions...</Channels>
 <Providers>...provider definitions...</Providers>
</DisplayProfile>
```

`<Properties>`, `<Channels>`, and `<Providers>` are mechanisms to do grouping. These mechanisms make the XML display profile document more structured, so that like objects are in each "bag." See Putting Together Display Profile Objects for more information on "bags."

The following sections describe the display profile objects in more detail.

DisplayProfile root Object

The DisplayProfile `root` container object enables the Desktop servlet to act as a container provider to get handles to providers, and so forth. There is no actual provider class associated with the channel. This channel should not be referenced by any other display profile object.

DisplayProfile root Object XML Syntax

```
<Container name="_desktopRoot" provider="none">
    <Properties />
    <Available />
    <Selected />
    <Channels/>
</Container>
```

Provider Object

A provider object is the software entity executed at run time when a channel is rendered. (Thus, a channel is the instance of a provider at run time.) The `<Provider>` display profile definition is a template from which display profile channels are defined. It sets up the class name for the Provider java object and default values for all required properties.

The `<Provider>` display profile definition contains the information necessary for a client of the display profile to construct the `provider` object, namely, the Java™ class name.

The `<Provider>` display profile definition sets default property values for all channels that point to this provider. Channel-specific properties are only necessary when the provider defaults need to be overwritten. The provider display profile object should contain default values for all properties that are used in the provider Java object. For example, if the provider Java code contains:

```
getStringProperty("color")
```

Channel Object

A `channel` object represents a single display element. The objects contained by a `channel` object can be thought of as properties for the channel. The `<Channel name>` definition includes a symbolic reference to the provider. In addition, you can define channel-specific properties to overwrite default values defined in the provider definition. A channel name needs to be unique for a given channel within a display profile document, but you can define the same name at different channel levels.

Example Channel Object XML Syntax

```
<Channel name="SampleXML" provider="XMLProvider">

    <Properties >
        <String name="refreshTime" value="600" advanced="true"/>
        <String name="title" value="XML Test Channel"/>
        <String name="description" value="This is a test of the XML Provider system"/>
        <String name="url"
value="file:///etc/opt/SUNWps/desktop/default/SampleXML/getQuotes.xml"/>
        <String name="xslFileName"
value="/etc/opt/SUNWps/desktop/default/SampleXML/html_stockquote.xsl"/>
    </Properties>

</Channel>
```

Container Object

A `container` object is identical to a `channel` object, except that it a `container` object does not generate content. That is, a container is a channel that gets its content from other channels. A `container` object allows for available and selected channel lists and can contain child channel definitions. A child channel is typically aggregated on a page with other channels and generates its own content. A container channel primarily generates content by aggregating the content of one or more child channels.

Example Container Object XML Syntax

```
<Container name="TemplateTableContainer"provider="TemplateTableContainerProvider">
  <Properties>
    <String name="title" value="Template Based Table Container"/>
    <String name="description"
        value="This is the channel for the front provider"/>
    <Collection name="channelsColumn" advanced="true">
        <String name="SampleJSP" value="2"/>
        <String name="SampleXML" value="2"/>
        <String name="Notes" value="2"/>
    </Collection>
    <Collection name="channelsRow" advanced="true">
        <String name="MailCheck" value="3"/>
        <String name="SampleRSS" value="2"/>
        <String name="SampleXML" value="2"/>
        <String name="App" value="5"/>
        <String name="SampleSimpleWebService" value="6"/>
        <String name="Bookmark" value="4"/>
        <String name="Notes" value="3"/>
    </Collection>
    <Collection name="channelsIsRemovable">
        <Boolean name="UserInfo" value="false"/>
    </Collection>
  </Properties>
  <Available>
    <Reference value="UserInfo"/>
    <Reference value="MailCheck"/>
    <Reference value="SampleRSS"/>
    <Reference value="SampleJSP"/>
    <Reference value="SampleXML"/>
    <Reference value="App"/>
    <Reference value="SampleSimpleWebService"/>
    <Reference value="Bookmark"/>
    <Reference value="Notes"/>
  </Available>
```

```
  <Selected>
    <Reference value="UserInfo"/>
    <Reference value="MailCheck"/>
    <Reference value="SampleRSS"/>
    <Reference value="SampleJSP"/>
    <Reference value="SampleXML"/>
    <Reference value="App"/>
    <Reference value="SampleSimpleWebService"/>
    <Reference value="Bookmark"/>
    <Reference value="Notes"/>
  </Selected>

  <Channels>
  </Channels>

</Container>
```

Putting Together Display Profile Objects

The `root`, `provider`, and `channel` objects can have properties associated with them. The display profile groups properties inside of a properties "bag." The term bag is used to indicate that its only purpose is a holding place for properties. A property does not have a properties bag associated with it. See *Sun Java System Portal Server 6 2004Q2 Desktop Customization Guide* for property definitions.

Property bags in channels, providers, and the root level have different semantics. Global properties are shared for all channels. A property defined as a global property here can be accessed by any channel. Themes are an example of a global property. Theme data is defined globally so they can be shared among all channels.

Properties defined in providers are defaults for channels based on that provider. If the property is not defined in the channel, then the default is used. The implication is that a provider must define every property used by a provider Java object. Thus, if the Java code contains:

```
String f = getStringProperty("color");
```

the corresponding `<Provider name>` definition in the display profile must define:

```
<String name="color" ... />
```

NOTE Do not use global properties as defaults for all channels. A display profile provider definition defines the property interface used by the `provider` object that will use the provider definition.

Channel properties override the defaults in the provider definition to customize the channel. For example, `URLScaperProvider` defines a `url` property. A default does not make sense here, as a channel would naturally override this value.

Display Profile Object Lookup

At runtime, the system never asks for properties directly from a provider. The request always goes to a channel. If a Java `provider` object requests a property, it searches the display profile in the following order until it finds the property, or until it reaches the top of the containment hierarchy:

1. Channel's properties
2. Channel's provider's properties
3. Channel's parent's properties
4. Channel's parent's provider's properties
5. Channel's parent's properties (and so on)
6. The global properties bag defined in the display profile `root` definition

Therefore, when a channel asks for the names of its properties, it gets the set of the union of all the above.

Properties that exist in a `provider` object are intended to have the semantics of default values for the channel. For example, for a provider `xml` that defines property `title`, all channels that are derived from provider `xml` inherit the `title` property. If the channel wants to override this property, it can set the value within its own properties.

Display Profile Properties

This section describes display profile properties and how to specify them.

Display Profile Property Types

The display profile property types are:

- Boolean—An atomic object representing a Boolean value. Example:

  ```
  <Boolean name="isEditable" value="false"/>
  ```

- Collection—An object representing either a list or hash table. A collection is a type of property, or named bag, in which to put other properties. Example:

  ```
  <Collection name="channelsRow">
      <String name="MailCheck" value="4"/>
      <String name="App" value="5"/>
  </Collection>
  ```

- Integer—An atomic object representing an integer value. Example:

  ```
  <Integer name="numberOfHeadlines" value="7"/>
  ```

- String—An atomic object representing a string value. Example:

  ```
  <String name="title" value="Table Container Channel 1"/>
  ```

- Reference—An object representing a pointer to a channel definition (that is, to a channel name in a container's selected and available channel lists.) Reference is an unnamed string useful for design tools to be able to distinguish such things from strings. Example:

  ```
  <Reference value="UserInfo"/>
  ```

Atomic property values can also be specified as body content. Example:

```
<String name=″foo″>bar</String>
<Integer name=″aNumber″>1</Integer>
<Boolean name=″flag″>false</Boolean>
```

Document Type Definition Element Attributes

The Portal Desktop DTD defines element attributes that allow you to control usage of display profile and its properties. Table 10-1 on page 205 lists document type definition element attributes. This three column table lists the attributes in the first column, a brief description in the second column, and an example in the third.

.

Table 10-1 Display Profile Attributes

Attribute	Definition	Example
`advanced`	"Hides" the display profile property from users in the Sun Java System Access Management administration console Channel and Container Management link when set to `true`. However, the property is not hidden when using the Edit XML or Download XML links. The `advanced` attribute is a Boolean attribute that can take a value of `true` or `false`. The default value is `false`.	`<String name="refreshTime" value="0" advanced="true"/>`

`lock`	Enables low-priority documents to prevent a higher-priority document from using merge semantics to change particular aspects of the display profile. When a display profile object is locked, it cannot be affected by merge semantics in lower priority documents.
	
	The `lock` attribute is a Boolean attribute that can take a value of `true` or `false`. The default value is `false`.
		`<Selected merge="fuse">`
		`...`
		`<Reference value="EmployeeNews" lock="true"/>`
		`...`
		`</Selected>`

`merge`	Controls how properties are combined as display profile documents from different LDAP nodes (base DN, DN, and role DNs) are merged to form a single representation (that is, Portal Desktop). Allowable values are `replace`, `remove`, and `fuse`. The default value is `fuse`. Note that fuse is not valid for atomic properties (boolean int, stringv ref).	See Display Profile Merge Types for replace, remove, and fuse examples.

Table 10-1 Display Profile Attributes

Attribute	Definition	Example
`priority`	Sets the priority of the display profile document. Display profile documents are merged from low priority to high priority. A lower number represents a lower priority. For example, a 1 is a lower priority than a 2. High priority documents override values set in lower priority documents using merge semantics (unless a lower priority document has locked the object for merging). Allowable values are integers and the keyword `user`. The priority `user` is the highest priority, and it should only be set for user-level display profile documents.	`<DisplayProfile version="1.0" priority="10">`
`propagate`	Controls how properties are treated when they are read non-locally but set locally. You can mark all display profile properties, including Boolean, Collection, Integer, Strings, and Reference, with the `propagate` attribute. The `propagate` attribute is a Boolean attribute that can take a value of `true` or `false`. The default value is `true`.	`<String name="color" value="blue" propagate="false"/>`

In the display profile XML, the following attributes are not listed in the XML file and displayed in the administration console unless the attribute's default value has been changed:

```
<advanced="false" lock="false" merge="fuse" propagate="true">
```

If a default value is reset, only the attribute whose default value has been changed is included in the XML fragment and displayed in the administration console. The default properties are inherited from the provider. If the default property is edited, it is displayed as customized.

Specifying Display Profile Properties

When you specify display profile properties, you need to consider how to “nest” them, how to use unnamed properties in collections, how to use conditional properties and how properties can be propagated.

Property Nesting

The display profile can contain nested properties (properties within properties) to any depth. This enables you to have collections of collections of collections of strings, or a collection of strings and collections, and so on. For example, here is a collection of collections:

```
<Collection name="people">
    <Collection name="john">
        <Integer name="age" value="31"/>
        <String name="eyes" value="hazel"/>
    </Collection>
    <Collection name="bob">
        <Integer name="age" value="35"/>
        <String name="eyes" value="blue"/>
    </Collection>
    ... etc ...
</Collection>
```

Unnamed Properties

Atomic property types (Boolean, Integer, and String) can be unnamed, for example:

```
<String value="apple"/>
```

is equivalent to

```
<String name=”apple” value=”apple”/>
```

That is, if an atomic property does not have a name then it is equivalent to the string value of that property.

For all practical purposes, this is useful only inside a collection, because it enables you to use collections to represent an ordered set (almost a list), instead of a table. For example, here is a collection representing a list of zip codes:

```
<Collection name="zipcodes">
    <Integer value="95112"/>
    <Integer value="95054"/>
    <Integer value="98036"/>
</Collection>
```

The key to using unnamed properties is that collections can represent tables (*name*=*value*) or lists.

NOTE Do not create an unnamed property with the same value as another unnamed property in the same collection. The property will be created, but the provider will not be able to access the value because of the duplicate name.

In addition, because the Portal Server treats a property that has the same name and value as equivalent to an unnamed Boolean property you may unintentionally create properties with duplicated names in the same collection. This again can result in all but one being inaccessible.

Conditional Properties

This provides a generic operation for retrieving conditional properties. The most common conditions are `locale` and `client`, but you can define properties on any sort of condition. See the *Sun Java System Portal Server 6 2004Q2 Desktop Customization Guide* for more information.

For instance, the implementation of the locale filter is:

```
public class LocalePropertiesFilter extends PropertiesFilter     {
     public LocaleProperties() {
     super();
     }
     String getCondition()
     return "locale";
     }
     public boolean match(ProviderContext pc, String  condition,    String
value) {
     return condition.toLowerCase().equals("locale")     &&
          getValue().equals(value);
     }
}
```

A conditional property lookup involves one or more property filters. If a filter in the filter list is required, then it must match for the overall conditional lookup to succeed. If a filter is not required, then it can fail to match without causing the overall lookup to fail.

A chain of non-required filters can be used to implement a progressively less-specific filter lookup, similar to the semantics of Java resource bundle lookup. For instance, an optional filter would be useful in a case where a locale lookup is followed by a date lookup. Given the filter {`locale=en, locale=US, date=03/03/2003`}, you can get it to successfully match a property with the qualifier {`locale=en; date=03/03/2003`} even though it does not exactly match the filter specification. This is done by setting the locale filter to be optional.

In the administration console, the conditional properties are displayed as `condition-value` and can be edited like collections. The conditional properties can be nested and can be added to a channel or inside another conditional property. Use the Add Property page to add a new conditional property.

<ConditionalProperties> Tag

The `<ConditionalProperties>` tag must be used to define the filtering criteria. The tag contains the following required attributes:

- `condition`: Specifies condition on which the filter should operate
- `value`: Specifies the value to be used in the filter

In the display profile, the `<ConditionalProperties>` tag can be defined as outlined in Code Example 10-1 on page 210.

Code Example 10-1 <ConditionalProperties> Tag Usage Sample

```
<Properties>
    <String name="foo" value="bar">
    <ConditionalProperties condition="locale" value="de">
        <String name="foo" value="german bar">
        <String name="baz" value="a german baz value">
    </ConditionalProperties>
    <ConditionalProperties condition="client" value="nokia">
        <ConditionalProperties condition="locale" value="de">
            <String name="foo" value="nokia german bar">
        </ConditionalProperties>
    </ConditionalProperties>
</Properties>
```

Display Profile Property Propagation

You can mark all display profile properties, including Boolean, Collection, Integer, Strings, and Reference, with the `propagate` attribute. The `propagate` attribute is a Boolean attribute that can take a value of `true` or `false` (the default is `true`). The `propagate` attribute controls how properties are treated when they are read non-locally but set locally.

For example, the set of properties for a channel consists of the set that is the union of:

- The set of properties existing locally in the channel's properties (`<Properties>`) bag
- The set of properties existing locally in the channel's provider (specified by the `provider` attribute on the channel)
- The set of properties existing locally in each ancestor channel of the channel (channel's parent, channel's parent's parent, and so on)
- The set of properties existing locally in each ancestor channel provider of the channel (channel's parent provider, channel's parent's parent provider, and so on)
- The set of global properties existing under the display profile `root` object

When a channel requests a property value, it can be read from any of these "remote" locations. When a property value is set, there are two options where to store the property value:

1. The channel's property bag

2. The remote location

The `propagate` attribute controls the location. When you set the `propagate` attribute to `true`, a property is stored locally to the object that set the property (in most cases, a channel). When you set the `propagate` attribute to `false`, the property is set in place (wherever it was read from). That is, when set to `false`, the existing value is changed, but when `true`, a new property is created and stored locally (unless it was already local).

Consider the following display profile XML fragment:

```
<DisplayProfile>
  <Properties>
    <String name="color" value="blue"/>
  </Properties>
...
<Channel name="testchannel" provider="..."/>
  <Properties/>
</Channel>
...
</DisplayProfile>
```

The property `color` lives in the global properties bag. Because `propagate` is not set (and is `true` by default), the following results if channel `testchannel` sets property `color`:

```
<DisplayProfile>
  <Properties>
    <String name="color" value="blue"/>
  </Properties>
...
<Channel name="testchannel" provider="..."/>
  <Properties/>
    <String name="color" value="new value"/>
</Channel>
...
</DisplayProfile>
```

The property is propagated to the local object that set it (the channel). On the other hand, if `propagate` were set to `false` in the global properties bag, for example:

```
<String name="color" value="blue" propagate="false"/>
```

The result of channel `testchannel` setting property `color` would be:

```
<DisplayProfile>
  <Properties propagate="false">
    <String name="color" value="new value"/>
  </Properties>
...
<Channel name="testchannel" provider="..."/>
  <Properties/>
</Channel>
...
</DisplayProfile>
```

In addition to individual properties, a properties bag can also be marked with the `propagate` attribute, for example:

```
<Properties propagate="false">
...
</Properties>
```

For a property to be considered as `propagate=false`, the following must be true:

- The property's `propagate` attribute must be `false`, or the property's properties bag's `propagate` attribute must be set to `false`.
- The above statement must be true for all mergers of the property.

For anything else, `propagate` is considered to be `true`.

You can only mark top-level properties with the `propagate` attribute. The display profile DTD does not disallow this but the display profile code ignores it. A top-level property is defined directly inside the properties bag.

Display Profile Document Priorities

At runtime, when a user logs in, the system determines the set of documents that makes up the user's display profile document set. The Desktop internal implementation of the display profile (the part that interprets the display profile) determines this set by looking at all of the LDAP nodes that the user belongs to. This can be the organization DN (`o=sesta.com`), suborganizations, role DNs (`cn=Role1,o=sesta.com`), and uid (`uid=jtb,ou=People,cn=Role1,0=sesta.com`), as well as the global display profile. The display profile documents from each of these LDAP nodes and global display profile are then read (if it exists there), and all of the documents are put into a set. The system sorts the set according to the document priorities. A lower number represents a lower priority. For example, a 1 is a lower priority than a 2. The documents are then sorted from lower number to higher number. See How the Merge Process Works for more information on this process.

The user level document (`uid=jtb,ou=People, . . .`) is a special case referred to as the *base document*. Think of the base document as a priority equal to infinity. Thus, it is always the highest number (and hence highest priority). All of the mergers are associated with the base document in sorted order, and the priority setting on a user document is always the highest. The `priority` attribute used in the `<DisplayProfile>` tag takes the special keyword `user` to indicate that the current display profile is the user level display profile.

When a merge occurs, it starts at the lowest priority document (lowest number) and proceeds in increasing priority number, until it arrives at the user (base) document.

Thus, the implication of display profile document priorities is that what really matters is the priority number. For example, an organization level document can have a higher priority than a role level document, but it does not have to. It depends on how you need to prioritize these documents for your site.

You specify the display profile document priority in the XML file with the `<DisplayProfile priority=` *syntax*`>` tag. You can change the priority by directly editing the display profile XML by using the Sun Java System Access Manager administration console or by using the `dpadmin` command to load the display profile. See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for more information on administering command line utilities for more information on the `dpadmin` command.

NOTE Do not assign the same priority to two display profile documents. Doing so causes the Desktop to not appear properly. However, the product does not check for duplicate document priorities.

Document Priority Example 1

This example uses two display profiles, one for the organization acme and one for the uid bill. When Bill logs in (uid=bill) to the Desktop, the bookmark channel titled “Bill’s Bookmarks” is displayed with the following three bookmarks (in that order):

- ACME
- Amazon
- EBay

```
display profile @ o=acme.com
<DisplayProfile version="1.0" priority="10">
...
  <Channel name="Bookmark" provider="BookmarkProvider" merge="fuse">
    <Properties>
      <String name="title" value="My Bookmarks" merge="replace" lock="false"
propagate="true"/>
      <String name="refreshTime" value="600" merge="replace" lock="false"
propagate="true"/>
         <Collection name="targets" merge="fuse" lock="false" propagate="true">
          <String value="ACME home page|http://www.acme.com" merge="replace" lock="false"
propagate="true"/>
        </Collection>
    </Properties>
  </Channel>
...
</DisplayProfile>
```

```
dp @ uid=bill,ou=people,o=acme.com
<DisplayProfile version="1.0" priority="1">
...
  <Channel name="Bookmark" provider="BookmarkProvider" merge="fuse">
    <Properties>
      <String name="title" value="Bill’s Bookmarks" merge="replace" lock="false"
propagate="true"/>
        <Collection name="targets" merge="fuse" lock="false" propagate="true">
          <String value="Amazon|http://www.amazon.com" merge="replace" lock="false"
propagate="true"/>
          <String value="EBay|http://www.ebay.com" merge="replace" lock="false"
propagate="true"/>
        </Collection>
    </Properties>
  </Channel>
...
</DisplayProfile>
```

Document Priority Example 2

This example uses three display profiles, the global display profile, the display profile for the organization `acme`, and the display profile for the role `hradmin`. When the user who is assigned to the `hradmin` role logs in to the Desktop, the `TemplateTableContainer` appears with the following channels:

- `UserInfo`
- `MailCheck`
- `SampleSimpleWebService`

```
dp @ global:
<DisplayProfile version="1.0" priority="0">
...
  <Container name="TemplateTableContainer" provider="TemplateTableContainerProvider"
merge="fuse">
    <Properties>
      ...
    </Properties>
    <Available>
      ...
    </Available>
    <Selected merge="fuse" lock="false">
      <Reference value="UserInfo"/>
    </Selected>
    <Channels/>
  </Container>
...
</DisplayProfile>
```

```
dp @ o=acme.com:
<DisplayProfile version="1.0" priority="10">
...
  <Container name="TemplateTableContainer" provider="TemplateTableContainerProvider"
merge="fuse">
    <Properties>
      ...
    </Properties>
    <Available>
      ...
    </Available>
    <Selected merge="replace" lock="false">
      <Reference value="Bookmark"/>
      <Reference value="Notes"/>
    </Selected>
    <Channels/>
  </Container>
...
</DisplayProfile>
```

```
dp @ cn=hradmin,o=acme.com:
<DisplayProfile version="1.0" priority="5">
...
  <Container name="TemplateTableContainer" provider="TemplateTableContainerProvider"
merge="fuse">
    <Properties>
      ...
    </Properties>
    <Available>
    <Selected merge="fuse" lock="true">
      <Reference value="MailCheck"/>
      <Reference value="SampleSimpleWebService"/>
    </Selected>
    <Channels/>
  </Container>
...
</DisplayProfile>
```

Display Profile Document Priority Summary

A display profile document has a low or high priority depending on whether you consider the merge order or the ability to lock as the defining factor.

Without considering locking, the lower numbered display profile document has a lower priority. The lower numbered display profile document gets merged first so the value of a higher priority document overrides the value of a lower priority document. In this sense, the lower numbered document has a lower priority.

However, the lower numbered display profile document can also lock an object so it cannot be affected by a higher numbered document. In this sense, the lower numbered document has a higher priority.

Display Profile Merge Semantics

The display profile is composed of a hierarchy of XML documents. Portal Server could store a display profile document for the user, each role the user belongs to, and the user's organization or suborganization. At runtime, the system merges these multiple display profile documents to deliver a particular portal desktop to the user. This process of merging display profile documents affects the final display profile by potentially changing channel, provider, and property definitions.

The display profile data format contains syntax that defines how these documents are combined. This definition is commonly known as *merge semantics*.

Merge semantics control how attributes are combined as display profile documents from different LDAP nodes (base DN, DN, and role DNs) are merged to form a single representation (that is, Desktop). Merge semantics assume an ordering to display profile documents. Sun Java System Access Manager does not provide hierarchical structure of roles. Instead, the users' role structure is flat. All roles are peers. Because of this, Portal Server imposes an additional ordering on Sun Java System Access Manager roles to simulate a hierarchical structure.

The set of display profile documents for a user consists of: the documents that exist at the user's LDAP organization an suborganization nodes; the documents that exist at each of the user's role nodes; and the document that exists at the user's entry node. Documents do not need to be defined at each of these nodes, but there must be at least one document defined at a node. The set of documents is sorted according to a priority value that the display profile document defines. See Display Profile Document Priorities for more information.

You can visualize the process of document merging as laying one display profile document on top of another. A merge happens where like named channels, providers, and properties fall on top of one another. Merging is based on the name of the display profile object, not the XML structure defined in the display profile document. Like named channels can exist in different containers within the containment hierarchy in the display profile to be merged.

How the Merge Process Works

When a user logs on to Portal Server, and after authentication takes place, the system determines the user's display profile by:

1. Locating all the display profile documents for that user by searching through the global display profile, and LDAP organization, suborganization, role, and user nodes that the user belongs to.
2. Placing the retrieved display profile documents in a temporary area, which you can visualize as a bag.
3. Sorting the display profile documents in the bag based on priority, starting at the lowest priority. (The node at which the document was retrieved does not influence the priority sorting. Also, the user display profile document always has the highest priority.)
4. Taking the documents out of the bag, lowest priority first, then placing the next higher level priority document over this document, and applying merge and lock semantics.
5. Continuing Step 4 until all the documents have been taken out of the bag so that the system returns a value to the user that is a merge of the objects found in the documents.

Display Profile Merge Types

The display profile uses the following three types of merges to determine how to combine display profile documents:

- `replace`—All the display profile objects defined in the higher priority document completely override the ones defined at the lower one. If the object does not exist in the lower priority document, it is added to the merge result (the object replaces the value in the merge results).
- `remove`—The named object is removed from the merge up to this point (the object is removed from the merge results). It no longer exists in the display profile (but it can be re-introduced by another document to be merged). It can be redefined by a higher priority document.
- `fuse`—The object from the lower priority document is combined with one from the higher priority document (the object is merged with the value in the merge results).

NOTE The exact meaning of each merge type depends on the display profile object they are applied to.

For channels and providers, `fuse` has special meaning. The channels themselves are not actually fused together. Rather, `fuse` indicates that the channel's or provider's properties should be combined. The `replace` semantic replaces the entire channel or provider, including all properties. The `remove` semantic removes the entire channel or provider from the merge up to that point.

The display profile `<DisplayProfile>` root node can also have merge semantics. The `replace` semantic means that all the DP objects defined in the higher priority document completely override the ones defined at the lower one. All merges up to that point are negated and the higher priority document is used as the new base for merging. The `remove` semantic indicates that all merge results up to the point of this document are to be discarded. The merge begins with the next display profile document found in the sorted set. As with channels and providers, the `fuse` semantic means that the contained objects (channels and providers) should be combined.

Atomic display profile properties (those that cannot contain other properties) cannot use the `fuse` semantic. This includes the String, Integer, Boolean, and Reference properties.

The set of properties for a channel consists of the channel's properties plus the channel's provider's properties plus the channel's parent's properties, and so on. You can think of this total set of properties as the channel's single document properties. An implication of document merging is that the total set of properties for a document consists of the set union of the channel's single document properties for all documents in the user's merge set.

Remove Example: Using remove Merge to Modify Container's Selected Channel List

The following example shows how all users' merge set can consist of an organizational level document that has the following display profile fragment.

```
<Container name="TemplateTableContainer" provider="TemplateTableContainerProvider"
merge="fuse">
    <Properties> ... </Properties>

    <Available> ... </Available>
    <Selected merge="fuse">
        <Reference value="UnixTipoftheDay"/>
    </Selected>
</Container>
```

The “unix tip of the day” describes ways to use UNIX. It is likely that users that belong to the admin role would not find this channel helpful. To remove this channel from everyone with the admin role, define the `TemplateTableContainer` channel in the admin role document as follows:

```
admin role
<Container name="TemplateTableContainer"
provider="TemplateTableContainerProvider" merge="fuse">
    <Properties> ... </Properties>

    <Available> ... </Available>
    <Selected merge="fuse">
        <Reference value="Outages"/>
        <Reference value="SolarisAdmin"/>
        <Reference value="AdminTipoftheDay"/>
        <Reference value="UnixTipoftheDay" merge="remove"/>
    </Selected>
</Container>
```

The preceding sample snippet causes the `Reference value="UnixTipoftheDay"` to be removed from the admin role display profile.

Replace Example: Using replace Merge to Remove Channel from All Users' Display

The following example shows how for a particular container, a role admin can ignore all of the channels defined in the organization level. The organization definition resembles the following:

```
organization display profile
<Container name=...>
    ...
    ...
    <Selected>
        <Reference name="X"/>
        <Reference name="Y"/>
        <Reference name="Z"/>
    </Selected>
</Container>
```

Because the role admin does not want any of the users under that role to have the X, Y, or Z channels, the container is defined as follows:

```
admin role
<Container name=...>
    ...
    ...
    <Selected merge="replace">
        <Reference name="A"/>
        <Reference name="B"/>
        <Reference name="C"/>
    </Selected>
</Container>
```

The selected list in the role document's container replaces the selected list in the organization document's container.

Fuse Example: Using fuse Merge to Create Role-based Channel List

You commonly use the `fuse` merge semantic to combine non-atomic display profile objects. These objects include `Collection` and the available or selected channel lists. Here, `fuse` indicates that all the properties contained in the non-atomic property should also be merged. Using `fuse` in this way enables the final non-atomic property presented to the user to be build up from various documents.

The following example display profile documents are for a user who belongs to the admin, employee, and movieFreak roles. The selected channels for the user appear at the end.

```
admin role
<Container name="TemplateTableContainer"
provider="TemplateTableContainerProvider" merge="fuse">
    <Properties> ... </Properties>

    <Available> ... </Available>
    <Selected merge="fuse">
        <Reference value="Outages"/>
        <Reference value="SolarisAdmin"/>
        <Reference value="AdminTipoftheDay"/>
    </Selected>
</Container>
```

```
employee role
<Container name="TemplateTableContainer"
provider="TemplateTableContainerProvider" merge="fuse">
    <Properties> ... </Properties>

    <Available> ... </Available>
    <Selected merge="fuse">
        <Reference value="Benefits"/>
        <Reference value="EmployeeNews"/>
    </Selected>
</Container>
```

```
movieFreak role
<Container name="TemplateTableContainer"
provider="TemplateTableContainerProvider" merge="fuse">
    <Properties> ... </Properties>

    <Available> ... </Available>
    <Selected merge="fuse">
        <Reference value="NewMoviesReleases"/>
        <Reference value="MovieShowTimes"/>
    </Selected>
</Container>
```

The resultant list of selected channels for the user is as follows, with the available channel list ordered in the same way that the merging was applied, from lower to higher priority:

```
<Container name="TemplateTableContainer"
provider="TemplateTableContainerProvider" merge="fuse">
    <Properties> ... </Properties>

    <Available> ... </Available>
    <Selected merge="fuse">
        <Reference value="Outages"/>
        <Reference value="SolarisAdmin"/>
        <Reference value="AdminTipoftheDay"/>
        <Reference value="Benefits"/>
        <Reference value="EmployeeNews"/>
        <Reference value="NewMoviesReleases"/>
        <Reference value="MovieShowTimes"/>
    </Selected>
</Container>
```

Merge Locking

Any display profile object that is able to be merged can also be locked. When an object is locked, it cannot be affected by merge semantics in higher priority documents. This enables low-priority documents to prevent a high-priority document from using the merge semantics to change particular aspects of the display profile.

Merge Locking Example: Using lock Merge to Force Property Value for All Users

The following example shows how to ensure that for a particular organization, all users see the “employee news” channel. The users cannot remove this channel from their display. At the organization level document, the container channel’s selected list is defined as follows:

```
<Selected merge="fuse">
    ...
    <Reference value="EmployeeNews" lock="true"/>
    ...
</Selected>
```

Merge Locking Example: Using lock Merge to Force-remove Channel from All Users’ Display

The following example shows how to force the “online games” channel to be removed. In this scenario, users have added this channel to the selected channels list in their user document, so simply removing it from the organization level document’s selected channel’s list will not work. Instead, the employee and organization lists will be merged together resulting in the “online games” channel being present. To forcibly remove the channel from all users under the organization, the selected channels list is defined as follows:

```
<Selected merge="fuse">
    ...
    <Reference value="OnlineGames" merge="remove" lock="true"/>
    ...
</Selected>
```

The `remove` semantic removes the channel from merged result, and `lock` prevents lower priority documents from merging the value back in.

Display Profile and Sun Java System Access Manager

The set of display profile documents for a user can consist of:

- The document that exists at the user's LDAP organization (or suborganization) node
- The documents that exist at each of the user's role nodes
- The document that exists at the user's entry node
- The document that exists at the global display profile

Documents do not need to be defined at each of these nodes, but there must be at least one document defined at a node. The set of documents is sorted according to a priority value that the display profile document defines. See Display Profile Document Priorities for more information. Merge semantics control how attributes are combined as display profile documents from different nodes are merged to form a single representation or Desktop. See Display Profile Merge Semantics for more information.

Administrators can edit the display profile using the Sun Java System Access Manager administration console. You can set up delegated administrators so that they do not see the display profile in the Sun Java System Access Manager administration console. You do this when you create the Desktop service template. When you create the template for the Desktop service, if you unselect the "Show Desktop Service Attributes" box, you can hide the display profile text from a delegated administrator.

TIP The organization administrator can define a container (or container hierarchy) associated with certain roles through the Portal Desktop service. Then, the delegated administrator (role administrator) can define the necessary channels and containers under this container through the Channel and Container Management link in the Portal Desktop attributes page. See Using the Channel and Container Management Link to Administer Channels for more information.

Administering the Display Profile

You edit the display profile (and other Portal Desktop service data) through the Sun Java System Access Manager administration console and `dpadmin` command. When you edit the display profile, you add, modify, and remove providers, containers, and channels from the display profile, and edit properties.

In addition, the Sun Java System Access Manager administration console provides the Channel and Container Management link in the Portal Desktop attributes page to add channels and edit properties. This link also enables you to modify properties when a new channel is created.

NOTE The Channel and Container Management link is suited for delegated administration and allows the administrator to add and modify attributes of containers and channels. The overall system administrator should be responsible for adding the container and providers available to the delegated administrator.

Table 10-2 on page 226 explains the different types of display profiles and how to use the Sun Java System Access Manager administration console to administer them. This three column table lists the types of display profiles in the first column, how to access that display profile using the Sun Java System Access Manager administration console, and a brief description in the third column.

Table 10-2 Types of Display Profile Documents

Type of Display Profile Document	How to View Using the Sun Java System Access Manager Administration Console	Description
Global Display Profile Document	Choose View Service Management. Click the properties arrow next to Portal Desktop. In the Desktop Global attributes section, click Edit XML.	Defines display profile elements that are inherited by all users on the system, regardless of the organization or role to which they belong. (Although currently not enforced, you might also want to use the display profile XML document to define the common providers that will be used by everyone.)

Table 10-2 Types of Display Profile Documents

Type of Display Profile Document	How to View Using the Sun Java System Access Manager Administration Console	Description
Dynamic Display Profile Document	Choose View Service Management. Click the properties arrow next to Portal Desktop. In the Desktop Dynamic attributes section, click Edit XML.	Describes container management and properties for channels. This display profile is not 'used' to generate a user's Desktop at runtime, but becomes the default for each newly created organization and role. By default, the dynamic display profile document is blank. To use the dynamic display profile, you need to first populate it.
Organization, Suborganization, or Role Display Profile	Choose View User Management. Select the appropriate organization, suborganization, and if necessary, select Roles from the Show menu. Select Services from the Show menu. Click the properties arrow next to Portal Desktop. In the Desktop page, click Edit XML.	Shows the display profile for the selected organization, suborganization, or role. When you create a new organization, suborganization, or role, you create a template for this entity. When you create the template for the Desktop service, the initial display profile is set to the dynamic display profile document as mentioned above. Thus, if the dynamic display profile is blank, nothing is filled in. Most likely, you use this display profile document to customize container management and channel properties to fit the needs of different organizations and roles.

When you install Portal Server, you create an initial organization. The installer then imports the display profile global level document, and the default display profile, based on what you specify.

After that, each time you create a new organization, suborganization, or role, the display profile is not automatically loaded. You must manually load the display profile for a newly created organization, suborganization, or role. See To Load the Display Profile (Administration Console) for more information.

The high-level steps to administer the display profile are:

1. Loading the display profile for any newly created organization, suborganization, or role. (You do not need to perform this step for the organization that is created during the installation process.)
2. Modifying the display profile using the `dpadmin` command, the Edit XML link, or as a file that has been saved and then loaded using the Download XML and Upload XML links.

3. Adding channels and containers, and adding, deleting, and modifying their properties using the Channel and Container Management link.

Default Display Profile Documents

Table 10-3 explains the display profile documents that the Portal Server Desktop supplies in the `/opt/SUNWps/sample/desktop` directory at the time the sample portal is installed. This two column table lists the display profile documents in the first column and a brief description in the second column.

Table 10-3 Display Profile Documents Supplied with Sample Portal

Display Profile Document	Description
`dp-anon.xml`	Used by the authless anonymous user.
dp-org.xml	Sample display profile loaded at the default organization level. It defines all the global properties that are used for the organization and the channel definitions that are used by the organization.
`dp-org-final.xml`	A copy of `dp-org.xml`, with NetMail links defined in the Bookmark and Applications channels. This display profile document is used when the NetMail service is created.
`dp-providers.xml`	Sample display profile loaded at the global display profile level. This document defines all the provider definitions. Because these providers are going to be used by all organizations, the system loads this display profile at the top level, and every organization is able to use them. If a provider definition is used only by one organization, define it in the organization level display profile.

See the *Sun Java System Portal Server 6 2004Q2 Desktop Customization Guide* for information on customizing these sample display profiles.

Loading the Display Profile

When you first install Portal Server, the installer create an initial organization. The installer also imports the display profile global level document, and the default display profile, based on what you specify. If you decide not to install the sample portals, the sample display profile documents are not installed.

After that, when you create a new organization, suborganization or role, the display profile is not automatically loaded. You must manually load the display profile for a newly created organization, suborganization, or role.

There are three basic methods for loading the display profile:

- Using the Edit XML link of the Sun Java System Access Manager administration console. With this method you use the Edit XML link and an existing display profile in an organization, which you copy then paste into the blank display profile of the newly created organization, suborganization, or role. See To Load the Display Profile (Administration Console).
- Using the command line. With this method you use the `dpadmin` command to load the display profile. See To Load the Display Profile (Command Line). Before using the `dpadmin` command, see Guidelines for Using the dpadmin Command.
- Using the Download and Upload links of the Sun Java System Access Manager administration console. With this method you download a display profile to a file and then upload a display profile from a file. See To Download and Upload a Display Profile.

NOTE You cannot edit the display profile XML directly through the administration console if your browser is Netscape 4.x.

To Load the Display Profile (Administration Console)

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Select the organization or suborganization from which you want to copy the display profile.
3. Choose Services from the View menu.
4. Click the properties arrow next to Desktop in the navigation pane.

   The Portal Desktop attributes appear in the data pane.

TIP You might have to scroll down to see the Desktop service.

5. Copy the Display Profile.

   Click Edit XML then select and copy the entire text of the display profile.

6. Select the organization, suborganization, or role for which you want to load the display profile.
7. Choose Services from the View menu in the navigation pane.
8. Click the properties arrow next to Desktop in the navigation pane.

   A list of Portal Desktop service attributes, including the display profile XML, is displayed in the data pane.

9. Click Edit XML.

   The display profile XML document appears in a text window.

10. Paste the copied display profile into the display profile window.
11. When done, click Save.

    The changes affect only users in this particular organization.

To Load the Display Profile (Command Line)

Use the `modify` subcommand of the `dpadmin` command to load a display profile.

For example, the following command loads the display profile (`dp-org.xml`):

```
dpadmin add -u "uid=amAdmin,ou=People,o=sesta.com,o=isp" -w password -d
"o=sesta.com,o=isp" dp-org.xml
```

NOTE You can add the `-r` or `--dry-run` option to the end of the command before the file name to verify that the command will be successful before actually writing any changes to LDAP.

To Download and Upload a Display Profile

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Select the organization or suborganization from which you want to copy the display profile.
3. Choose Services from the View menu in the navigation pane.
4. Click the properties arrow next to Desktop in the navigation pane.

   The Portal Desktop attributes appear in the data pane.

TIP You might have to scroll down to see the Desktop service.

5. Click Download XML in the Global attributes section and save the display profile to a file.
6. Select the organization, suborganization, or role for which you want to upload the display profile in the navigation pane.
7. Choose Services from the View menu in the navigation pane.
8. Click the properties arrow next to Desktop in the navigation pane.
9. Click Upload XML and specify the file to load.
10. Click Upload.

    A message indicating that the display profile upload was a success appears.

11. Click Close.

    The changes affect only users in this particular organization.

To View the Entire Display Profile

- Run the `dpadmin` command with the `list` subcommand to view the entire display profile, for example:

```
dpadmin list -u "uid=amAdmin,ou=People,o=sesta.com,o=isp" -w password -d "o=sesta.com,o=isp"
```

To Remove a Display Profile

If you need to remove a display profile for some reason, for example if it is corrupted, you can use the `dpadmin` command with the `remove` subcommand.

For example, to remove the entire display profile (`dp-org.xml`) from the root:

```
dpadmin remove -u "uid=amAdmin,ou=People,o=sesta.com,o=isp" -w password -d
"o=sesta.com,o=isp" -t root
```

If you remove a display profile from the root or from a node at which you require a display profile, you must load a new one. For example, if you removed the `dp-org.xml` display profile as shown above, you will have to load another similar display profile such as the `dp-org-final.xml` display profile. See To Load the Display Profile (Command Line) for information on loading a display profile.

Using the Channel and Container Management Link to Administer Channels

You use the Channel and Container Management link to administer:

- Properties—You can define and add global display profile properties.
- Containers—You can add or remove a content container to or from a container. You can also modify a content container's properties.
- Channels—You can add or remove a channel to or from a container
- You can also modify a channel's properties.

NOTE Currently, you can work with channels and containers and their properties using the Channel and Container Management link. This link does not work with providers.

When using the Desktop attributes page, delegated administrators see only the Channel and Container Management link. All other display profile attributes are hidden, and thus made secure.

Channel and Container Management Default Providers

The Portal Desktop Channel and Container Management link displays a management screen that allows you add or remove container channels or content channels.

Add Channels

The Add link for the Channels list allows you to select a content provider to add from a list of defined content providers. Table 10-4 on page 233 shows the provider channels that are available to use as a basis to create new channels. This two-column table lists the providers in the first column and a brief description of the provider in the second column. For more information on defined content providers, see the *Sun Java System Portal Server 6 2005Q1 Desktop Customization Guide*.

Table 10-4 Defined Provider Channels

Provider	Description
`AppProvider`	Lists links to web applications (users can customize list).
`BookmarkProvider`	Allows users to manage a list of bookmarks displayed on a portal page.
`JSPProvider`	Obtains content from one or more JSP™ files.
`LoginProvider`	Allows users to authenticate to a Sun Java System Access Manager from an anonymous portal page.
`MailCheckProvider`	Gives information about a user's mail status.
`NotesProvider`	Lists system-wide messages and allows users to post such messages
`SearchProvider`	Supplies a search function using the Sun Java System Portal Server Search Engine.

Table 10-4 Defined Provider Channels *(Continued)*

Provider	Description
`URLScraperProvider`	Obtains content from a given URL and uses the Sun Java System Portal Server to format the content.
`UserInfoProvider`	Collects information from the display profile and Access Manager. It displays a greeting, the user's name, time zone, locale and has access to the user's IMAP and SMTP data.
`XMLProvider`	Obtains XML content from a given URL and uses XSLT to translate the content to markup language.

Simple Web Services Provider

The Simple Web Services (SWS) Provider provides the ability to access data-oriented Web Services. Based on this provider, a sample channel demonstrates Web Services' implementation by accessing a currency conversion rate service.

There are two types of simple web service channels:

- Pre-Configured Web Service Channel
- New Container Channels

The sample pre-configured web service channel is available on the sample portal desktop by default. The sample configurable web service channel can be added by the administrator using the Access Manager admin console.

Either web service channels are best suited for use with relatively simple web services; for example, web services that have non-complex input parameters and user interface presentation requirements. If the Simple Web Service Provider detects that it is not equipped to handle a particular web service, it will display a suitable message to the user.

At any given time, a channel based on this provider can be bound to a single web service and associated method. The Simple Web Service Provider will support simple data types, such as integer, string, double. In this release, the simple web service provider:

- Will support arrays of simple and complex types in the input and output parameters.
- Will support NestedComplex Types and one-dimensional homogeneous arrays of SimpleType and ComplexTypes.

- Will support web services that would involve complex xmll messages,whose Java equivalents would be Arrays, Structures(Java Beans) and One-dimensional arrays of SimpleTypes(int,char,string) and Structures / Objects(Complex Types)
- Will support one-way operations in webservices. It does support only notification style one-way operations of webservice. It does not support solicit response style services (This is a limitation of current jax-rpc).
- Will not support the use of fault data in the binding operations in the WSDL Definition.

The Simple Web Service Provider will support the following WSDL configuration property types:

- SOAP Binding Style: rpc & document
- SOAP Encoding Type: encoded & literal

NOTE The rpc/literal combination is not supported. Support for `.Net` based web services might be limited.

Pre-Configured Web Service Channel

The sample pre-configured web service channel provides the means to interface with sample currency converter service.

To set up a pre-configured web service channel, you will be required to specify the WSDL URL and method name via the administration console.

Configurable Web Service Channel

The configurable web service channel allows the user to switch the channel to point to a user specified web service. This is achieved by giving the user the ability to modify values for the WSDL URL and the method name belonging to the web service. However, unlike the pre-configured channel type, the configurable web service channel will not allow the user any facility to store default values for the web service input parameters.

New Container Channels

The New link for the Container Channels list allows you to select a container provider to create from a list of defined container providers. Table 10-5 shows the shows them defined provider channels that are available to use as a basis to create new channels. This two-column table lists the providers in the first column and a brief description of the provider in the second column. For more information on defined content providers, see the *Sun Java System Portal Server 6 2004Q2 Desktop Customization Guide*

Table 10-5 Defined Provider Container Channels

Provider	Description
`JSPFrameCustomTableContainerProvider`	Create a new frame on a user's JSP frameset-based Portal Desktop.
`JSPSingleContainerProvider`	Displays a single channel.
`JSPTabContainerProvider`	Displays a channel that is made up of a number of tabs with titles on them.
`JSPTabCustomTableContainerProvider`	Creates a new tab on a user's JSP tab-based Portal Desktop.
`JSPTableContainerProvider`	Displays the content channels in a table.
`TemplateEditContainerProvider`	Draws the frame for the Edit page.
`TemplateTabContainerProvider`	Supports multiple tabs.
`TemplateTabCustomTableContainerProvider`	Creates a new tab.
`TemplateTableContainerProvider`	Displays content channels in a table.

To Create a Channel or Container Channel

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Select the organization, suborganization, or role to which you want to add a channel.

   When you log in as a delegated administrator, you are automatically taken to the organization, suborganization, or role to which you have administrative access.

3. Choose Services from the View menu in the navigation pane.

4. Click the properties arrow next to Portal Desktop in the navigation pane.

   The Desktop attributes page appears in the data pane.

5. In the Desktop page, click the Channel and Container Management link.

   The Channels page appears, with the container path set at the root.

6. Click the Container that you want to add the channel or container to.

   The top of the page displays the container path where the channel will be added. Defined channels and container, if any, appear in lists.

7. Click New to add a container channel or channel.

   To add a container channel, click New under Container Channel. To add a channel, click New under Channel.

   The New Channel page appears.

8. Type a channel name and select the type of provider from the menu.

   See Table 10-4 on page 233 for the available providers.

9. Click Create.

To Modify a Channel or Container Channel Property

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Select the organization, suborganization, or role in which you want to modify a channel.

   When you log in as a delegated administrator, you are automatically taken to the organization, suborganization, or role to which you have administrative access.

3. Choose Services from the View menu.

4. Click the properties arrow next to Portal Desktop in the navigation pane.

   The Desktop attributes page appears in the data pane.

5. In the Desktop page, click the Channel and Container Management link.

   The Channels page appears. At the top is the container path. The defined channels appear in a list.

6. Click the Edit Properties link beside the channel or container channel to be modified.

   The Properties page appears.

7. Modify the properties as needed.

   See the *Sun Java System Portal Server 6 2004Q2 Desktop Customization Guide* for more information on channel properties.

8. When done, click Save.

To Remove a Channel or Container Channel

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Select the organization, suborganization, or role in which you want to modify a channel.

   When you log in as a delegated administrator, you are automatically taken to the organization, suborganization, or role to which you have administrative access.

3. Choose Services from the View menu in the navigation pane.

4. Click the properties arrow next to Portal Desktop in the navigation pane.

   The Desktop attributes page appears in the data pane.

5. In the Desktop page, click the Channel and Container Management link.

   The Channels page appears. At the top is the container path. The defined channels appear in a list.

6. Click the checkbox beside the channel or container channel to be removed. Then click Delete.

7. The channel is deleted and the Channels list is updated to show its removal.

Administering Containers

When administering containers, you can use the Sun Java System Access Manager administration console to directly edit the display profile XML. You can also use the `dpadmin` command, which for the most part this section describes by using various examples.

These examples include:

- To View a Display Profile Object
- To Replace a Channel in a Container
- To Replace a Property in a Channel
- To Add a Channel to a Container
- To Add a Property to a Collection
- To Add a Collection Property

- To Remove a Property from a Channel or Container
- To Remove a Provider
- To Remove a Channel from a Container
- To Change a Display Profile Document Priority
- To Make a Channel Available for a Container
- To Make a Channel Unavailable for a Container
- To Select a Channel from a Container's Available Channel List
- To Unselect a Channel from a Containers Available Channel List

See Using the Display Profile Text Window for information on editing the display profile through the Sun Java System Access Manager administration console.

Using the dpadmin Command

The syntax of the `dpadmin` command is:

```
$ dpadmin list|merge|modify|add|remove [command-specific options] -u uid -w password {-g|-d dn} [-l locale] [-r] [-b] [-h] {-v|--version} [file]
```

See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for the complete syntax of the `dpadmin` command. When running the `dpadmin` command, note the following:

- `file` argument—If present, the `file` argument must also be the last argument on the command line. It specifies a path to an XML file that contains an XML fragment that conforms to the display profile DTD. Subcommands that require XML input include `modify` and `add`.

  When adding or modifying an entire display profile, always include a proper XML header, for example

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
```

- `list` subcommand—Retrieves and displays display profile node objects. Objects are displayed in their native XML format. The object to be displayed is sent to standard out. If you do not use the `-n` or `--name` option, the entire display profile document is displayed. If you do not use the `-n` or `--name` option does not specify a DP node object, then the entire DP document is displayed.
- `merge` subcommand—retrieves and displays the merged result of the specified DP node objects. Objects are displayed in their native XML format. The object to be displayed is sent to standard out. If you do not use the `-n` or `--name` option, then an error is reported.
- `modify` subcommand— Changes the value of an existing display profile object. This command assumes that the object already exists in the display profile. The `modify` subcommand reads data for the new object from either standard input or the file specified as an argument. Data for the new object must be XML and conform to the display profile DTD. Specifically, the object data must be a well-formed XML fragment.
- `add` subcommand—Adds a new object to the display profile. This subcommand assumes that the object to be added does not exist in the display profile. The `add` subcommand reads data for the new object from either standard input or the file specified as an argument. Data for the new object must be XML and conform to the display profile DTD. Specifically, the object data must be a well-formed XML fragment.
- `remove` subcommand—Removes an existing object from the display profile.
- `-g` option—Specifies the global level display profile document.
- `-d` *dn* option—Designates the DN where `dpadmin` will execute. The `-d` and `-g` options are mutually exclusive.
- `-r` or `--dry-run` option—Reports whether the current command will succeed or fail, and does not write any changes into LDAP. This is useful to ensure that a particular `dpadmin` command is correctly formatted.
- `-n` or `--name` option—Specifies a fully qualified name of the display profile container, channel or provider object, or of the parent of the display profile object. If the name agrument does not specify a DP node object, then an error is reported.
- `-p` or `--parent` options—Specify a fully qualified name of the parent display profile container, channel or provider object, or of the parent of the display profile object.
- `-v` or `--version` options—Print the version number of the `dpadmin` command to standard output.

Guidelines for Using the dpadmin Command

Use the following guidelines when running the dpadmin command to update the display profile:

- Make sure no other administrator is currently using the Sun Java System Access Manager administration console or dpadmin command to make display profile modifications. Such a situation could cause changes to be lost, as there is no locking mechanism to prevent dpadmin and the administration console from accessing the display profile at the same time.
- The preferred sequence when using dpadmin is to put your modifications into a file as an XML "fragment" then run the dpadmin command with the add subcommand. For example,

  /opt/SUNWps/bin/dpadmin add -u "uid=amAdmin,ou=People,o=sesta.com,o=isp" -w *password* **-d "uid=anonymous,ou=people,o=sesta.com,o=isp" newtheme.xml**

  In this example, newtheme.xml is a file containing the XML "fragment" to be added to the display profile.
- If you edit a display profile document directly, first use the dpadmin command with the list subcommand to obtain the latest contents of the display profile, make your edits, then run the dpadmin command with the modify subcommand. For example,

  /opt/SUNWps/bin/dpadmin list -u"uid=amAdmin,ou=People,o=sesta.com,o=isp" -w *password* **-d "o=sesta.com,o=isp" > dp-org.xml**

  (Edit the dp-org.xml file.)

  /opt/SUNWps/bin/dpadmin modify -u "uid=amAdmin,ou=People,o=sesta.com,o=isp" -w *password* **-d "o=sesta.com,o=isp" dp-org.xml**

CAUTION Between the time you run the dpadmin list and dpadmin modify commands, do not change the display profile document in the LDAP server in any way (by using the administration console, dpadmin, or ldapmodify commands). Otherwise, those changes will be overwritten by the latest dpadmin modify.

Modifying the Display Profile

You can modify display profile objects by performing one of the following:

- Manually editing an existing display profile document then loading it at the appropriate LDAP node or global level by using the `dpadmin modify` command.
- Running the `dpadmin` command with the specified changes, in XML text, on standard input. When adding a new object you use the `add` subcommand. When modifying an existing object, you use the `modify` subcommand.
- Creating a new display profile document from scratch then loading it at the appropriate LDAP node or global level by using the `dpadmin modify` command.

Understanding Display Profile Error Messages

The system reports errors when you try and save a display profile document containing invalid XML. The error messages appear as a title, a message, and a sub-message. The title of the message box is “Invalid XML document.” The message appears as one of the following:

- Failed to parse XML...
- Missing doctype in the XML
- Failed to sore DP...
- Invalid XML input...

If you receive an “Invalid XML document” error, you need to correct the error to be able to save the XML document you are working on.

To View a Display Profile Object

- Use the `list` subcommand to view a display profile object.

  For example, the following command gets the channel, container, or provider named `TemplateTableContainer` and prints it to standard output.

```
dpadmin list -n "TemplateTableContainer" -u "uid=amAdmin,ou=people,o=sesta.com,o=isp" -w
password -d "o=sesta.com,o=isp"
```

NOTE You can view the entire display profile document by omitting the `-n` option.

To Replace a Channel in a Container

1. Use the `modify` subcommand to replace a channel in a container with a value specified on standard input.

   For example, this command replaces the channel `Test` in the container `TemplateTableContainer` with value specified on standard input.

```
dpadmin modify -p TemplateTableContainer -u "uid=amAdmin,ou=People, o=sesta.com,o=isp" -w
password -d "o=sesta.com,o=isp" <<EOF
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
<Channel name="Test" provider="testprovider">
    <Properties>
        <String name="title" value="Test Channel"/>
        <String name="description" value="This channel is a test."/>
    </Properties>
</Channel>
EOF
```

2. Use the `list` subcommand to verify that the channel was replaced.

   See To View a Display Profile Object for information.

To Replace a Property in a Channel

1. Use the `modify` subcommand to replace a property in a channel with a value specified on standard input.

   For example, the following command acts upon the channel `NewChannel` to replace the property named in the `new.xml` with the new object in said file, where `new.xml` is:

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
<String name="welcome" value="Hi, welcome to your desktop!"/>
```

```
dpadmin modify -p TemplateTableContainer/NewChannel -u
"uid=amAdmin,ou=People,o=sesta.com,o=isp" -w password -d "o=sesta.com,o=isp" new.xml
```

2. Use the `list` subcommand to verify that the property was replaced.

   See To View a Display Profile Object for more information.

To Add a Channel to a Container

1. Modify your display profile input XML file to include only the new `<Channel>` definition, for example, create the following file `testadd.xml`:

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
<Channel name="TestChannel" provider="testprovider">
    <Properties>
      <String name="teststring" value="sfds"/>
    </Properties>
</Channel>
```

2. Use the `add` subcommand to add the channel to a container.

   For example, the following command adds a new channel defined in `testadd.xml` to the display profile. In this example, the new channel must be added in the `TemplateTableContainer` level. If you do not specify a parent object with the `-p` option, the channel is added at the root level.:

```
dpadmin add -p "TemplateTableContainer" -u "uid=amAdmin,ou=People, o=sesta.com,o=isp" -w
password -d "o=sesta.com,o=isp" testadd.xml
```

NOTE When you add a new channel to JSPTabContainer, you actually add a new tab. JSPTabContainer requires TabProperties defined for all its available and selected tabs. Thus, for any new container or channel added to the JSTTabContainer, add the following XML snippet inside the TabProperties Collection in the JSPTabContainer for which the new channel or container is added.

```
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
<Collection name="<New Channel Name>">
            <String name="title" value="<New Channel Title>"/>
            <String name="desc" value="<New Channel Description>"/>
            <Boolean name="removable" value="false"/>
            <Boolean name="renamable" value="true"/>
            <Boolean name="predefined" value="true"/>
</Collection>
```

3. Use the list subcommand to verify that the channel was added.

   See To View a Display Profile Object for information.

To Add a Property to a Collection

1. Use the combine (-m) option to add a new property to a collection.

   For example, the following command adds a new property msg2 to the collection bar. The existing property, msg, still remains in the result. The list subcommand is used before and after to show the property values.

```
dpadmin list -n TemplateTableContainer -u "uid=amAdmin,ou=People, o=sesta.com,o=isp" -w
password -d "o=sesta.com,o=isp
...
<Collection name="foo">
    <Collection name="bar">
        <String name="msg" value="hi"/>
    </Collection>
</Collection>
...
```

```
dpadmin modify -p TemplateTableContainer -u "uid=amAdmin,ou=People, o=sesta.com,o=isp" -w
password -d "o=sesta.com,o=isp" -m <<EOF

<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
<Collection name="foo">
    <Collection name="bar">
        <String name="msg2" value="woo hoo"/>
    </Collection>
</Collection>
EOF
```

```
dpadmin list -n TemplateTableContainer -u "uid=amAdmin,ou=People, o=sesta.com,o=isp" -w
password -d "o=sesta.com,o=isp”
...
<Collection name="foo">
    <Collection name="bar">
        <String name="msg" value="hi"/>
        <String name="msg2" value="woo hoo"/>
    </Collection>
</Collection>
...
```

To Add a Collection Property

1. Use the `add` subcommand to add a collection with a value specified on standard input.

   For example, the following command adds the collection property `zipCodes` specified on standard input to the channel, container, or provider named `Postal`.

```
dpadmin add -p SamplesTabPanelContainer/Postal -u "uid=amAdmin,ou=People,
o=sesta.com,o=isp" -w password -d "o=sesta.com,o=isp" <<EOF
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
<Collection name="zipCodes">
    <Integer value="98012"/>
    <Integer value="98036"/>
    <Integer value="94025"/>
    <Integer value="95112"/>
</Collection>
EOF
```

2. Use the `list` subcommand to verify that the collection property was added.

   See To View a Display Profile Object for information.

To Remove a Property from a Channel or Container

1. Use the `remove` subcommand to remove a property from a channel or container.

   For example, the following command removes the property `locations` from the `Bookmarks` channel (or container) at the global level.

```
dpadmin remove -t property -p Bookmarks -n locations -u "uid=amAdmin,ou=People,
o=sesta.com,o=isp" -w password -g
```

2. Use the `list` subcommand to verify that the property was removed.

   See To View a Display Profile Object for information.

To Remove a Provider

1. Use the `remove` subcommand to remove a provider.

   For example, the following command removes the provider `NotesProvider`.

```
dpadmin remove -t provider -n "NotesProvider" -u "uid=amAdmin,ou=People, o=sesta.com,o=isp"
-w password -d "o=sesta.com,o=isp"
```

2. Use the `list` subcommand to verify that the provider was removed.

   See To View a Display Profile Object for information.

To Remove a Channel from a Container

1. Use the `remove` subcommand to remove a channel from a container.

   For example, the following command removes the channel `Test` that exists in the parent container `TemplateTableContainer`.

```
dpadmin remove --type channel --parent TemplateTableContainer --name "Test" --runasdn
"uid=amAdmin,ou=People,o=sesta.com,o=isp" --password password --dn "o=sesta.com,o=isp"
```

2. Use the `list` subcommand to verify that the channel was removed.

   See Chapter 10, “Administering the Display Profile” for information.

To Change a Display Profile Document Priority

1. Use the `modify` subcommand to change the priority of a display profile document.

   For example, the following command changes the document priority from the original priority to 10 for the organization.

```
dpadmin modify -m -u "uid=amAdmin,ou=People,o=sesta.com,o=isp" -w password -d
"o=sesta.com,o=isp" <<EOF
<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
<DisplayProfile priority=”30” version=”1.0”
<Properties/>
<Channels/>
<Providers/>
EOF
```

2. Use the list subcommand to verify that the priority change was made.

   See To View a Display Profile Object for information.

To Make a Channel Available for a Container

1. Use the modify subcommand with the combine (-m) option to add a channel specified on standard input to a container's existing Available list.

   For example, the following command adds the BookMark channel to the Available list of the TemplateTableContainer.

```
dpadmin modify -p TemplateTableContainer -u "uid=amAdmin,ou=People, o=sesta.com,o=isp" -w
password -d "o=sesta.com,o=isp" -m <<EOF

<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
<Available>
    <Reference value="BookMark">
</Available>
EOF
```

2. Use the list subcommand to verify that the priority change was made.

   See To View a Display Profile Object for information.

To Make a Channel Unavailable for a Container

1. Use the `remove` subcommand to remove a channel from a container's `Available` list.

   For example, the following command removes the channel `Test` from the `Available` list in the parent container `TemplateTableContainer`.

```
dpadmin remove --type available --parent TemplateTableContainer --name "Test" --runasdn
"uid=amAdmin,ou=People,o=sesta.com,o=isp" --password password --dn "o=sesta.com,o=isp"
```

2. Use the `list` subcommand to verify that the channel was removed.

   See To View a Display Profile Object for information.

To Select a Channel from a Container's Available Channel List

1. Use the `modify` subcommand with the `combine` (`-m`) option to add a channel specified on standard input to a container's existing `Selected` list.

   For example, the following command adds the `BookMark` channel to the `Selected` list of the `TemplateTableContainer`.

```
dpadmin modify -p TemplateTableContainer -u "uid=amAdmin,ou=People, o=sesta.com,o=isp" -w
password -d "o=sesta.com,o=isp" -m <<EOF

<?xml version="1.0" encoding="utf-8" standalone="no"?>
<!DOCTYPE DisplayProfile SYSTEM "jar://resources/psdp.dtd">
<Selected>
    <Reference value="BookMark">
</Selected>
EOF
```

2. Use the `list` subcommand to verify that the priority change was made.

   See To View a Display Profile Object for information.

To Unselect a Channel from a Containers Available Channel List

1. Use the `remove` subcommand to remove a channel from a container's `Selected` list.

   For example, the following command removes the channel `Test` from the `Selected` list of the parent container `TemplateTableContainer`.

```
dpadmin remove --type selected --parent TemplateTableContainer --name "Test" --runasdn
"uid=amAdmin,ou=People,o=sesta.com,o=isp" --password password --dn "o=sesta.com,o=isp"
```

2. Use the `list` subcommand to verify that the channel was removed.

   See To View a Display Profile Object for information.

Using the Display Profile Text Window

The Sun Java System Access Manager provides a text window for viewing and directly editing the display profile text. As long as you have administrative access to an organization, suborganization, or role, you can use the Sun Java System Access Manager administration console to navigate to this text window and view or edit the display profile.

NOTE You cannot edit the display profile XML directly through the administration console if your browser is Netscape 4.x.

To Access the Display Profile Text Window

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Select the organization, suborganization, or role for which you want to modify the display profile document.

   When you log in as a delegated administrator, you are automatically taken to the organization, suborganization, or role to which you have administrative access.

3. Choose Services from the View menu in the navigation pane.

4. Click on the properties arrow next to Portal Desktop in the navigation pane.

   The Desktop attributes page appears in the data pane.

5. In the Desktop page, click the Display Profile Edit XML link.

   The display profile appears in a text window.

NOTE By default, the display profile priority level is set to the keyword `user`, indicating that the current display profile is the user level display profile. Other allowable values are integers with lower numbers representing lower priorities. For example, a 1 is a lower priority than a 2.

6. Make your changes and click Save.

NOTE Changes to global, organization, suborganization, or role level documents are effectively immediately. Changes to user level documents are effectively after users log out and log in.

Chapter 11

Administering the NetMail Service

This chapter describes how to administer the NetMail service. This chapter contains these sections:

Overview of the NetMail Service

NetMail service implements the NetMail (Java™) and NetMail Lite email clients. These clients work with standard IMAP and SMTP servers. NetMail allows users to access one or more mail servers to read, compose and delete emails, and create, access and delete folders.

In Sun Java™ System Portal Server 6, you define and manage the NetMail service through the Sun Java™ System Access Manager administration console. The NetMail service defines the service attributes and default values for the NetMail client for managing email messages and its configuration. You define and customize service attribute values for an organization and its users to control how the NetMail client behaves.

Administering the NetMail Service

The Sun Java System Access Manager Policy Service enables you to define rules or access to resources. Policies can be role-based or organization-based and can offer privileges or define constraints.

By default, the Policy Configuration service is automatically added to the top-level organization. Suborganizations must add their policy services independently of their parent organization. Any policy service you create must be added to all organization. The high-level steps to use policies are:

1. Adding the Policy service for an organization.
2. Creating a referral policy for a suborganization. You can delegate an organization's policy definitions and decisions to another organization. (Alternately, policy decisions for a resource can be delegated to other policy products.) A referral policy controls this policy delegation for both policy creation and evaluation. It consists of a rule and the referral itself. If the policy service contains actions that do not require resources, referral policies cannot be created for suborganizations. See "To Create a Referral Policy for a Suborganization" on page 257" for information.
3. Creating a normal policy for a peer or suborganization. You create a normal policy to define access permissions. A normal policy can consist of multiple rules, subjects, and conditions. See "To Create a Normal Policy for a Suborganization" on page 258" for information.

To Add a Policy Service for a Peer or Suborganization

Suborganizations do not inherit their parent's services, so you need to add a suborganization's Policy service.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
2. Navigate to the organization or suborganization that you want to create a referral policy.

   All created organizations are displayed in the navigation pane.

3. Select Organizations from the View menu in the navigation pane and select desired organization from the Name menu.
4. Select Services from the View menu.
5. Click Add.

   The Add Services page appears in the data pane. Click the check box for the NetMail service, then click OK.

   The newly added service appear in the navigation pane.
6. Configure the NetMail service by clicking the properties arrow.
7. The following message appears in the data pane:

   No template available for this service. Do you want to create it?
8. Click Create in the message box to create the template.

   The NetMail attributes appear.
9. Make any changes to the NetMail attributes.

   See *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for information on the NetMail attributes.
10. Click Save to store the final values in the service template.

NOTE When you create a new organization, you need to create and assign a NetMail policy for that organization. You do not need to do so for the sample portal as NetMail is already enabled by default.

To Create a Referral Policy for a Suborganization

You can delegate an organization's policy definitions and decisions to another organization. A referral policy controls this policy delegation for both policy creation and evaluation. It consists of a rule and the referral itself. The referral must define the parent organization as the resource in the rule, and it must contain a SubOrgReferral with the name of the organization as the value in the referral.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization that contains the suborganization where you want to create a referral policy.

   All created organizations are displayed in the navigation pane.

3. Select Policies from the View menu.

4. Click New to create new policy.

   The Create Policy page appears in the data pane.

5. For Name, type SubOrgReferral_NetMail. Make sure you select Referral in Type of Policy. Then click Create.

6. Click Rules from the View menu in the data pane and click Add. Make sure NetMail is selected and click Next.

   The Add Rule template appears in the data pane.

7. Select NetMail in Service and click Next

8. Enter NetMailRule in Rule Name and click Create.

9. Click Referrals from the View menu in the data pane and click Add.

   The Add Referral template appears in the data pane.

10. Enter SubOrgReferral_*suborg_name* in Name.

    Make sure that the name of the suborganization (is selected for Value in the data pane and click Create to complete the policy's configuration.

11. Click Save in the data pane.

    The message "The policy properties have been saved" is displayed when the data is saved.

To Create a Normal Policy for a Suborganization

You create a normal policy to define access permissions. A normal policy can consist of multiple rules, subjects, and conditions.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Navigate to the organization or suborganization that you want to assign a policy.

   All created organizations are displayed in the navigation pane.

3. Choose Policies from the View menu.

   The policies for that organization are displayed.

4. Select New in the navigation pane. The New Policy page opens in the data pane.

5. Enter SubOrgNormal_NetMail in Name. Make sure you select Normal in Type of Policy. Click Create

6. Choose Rules from the View menu in the data pane and click Add. The Add Rule page opens in the data pane

7. Select NetMail from the Service menu and click Next. Enter NetMailRule in Rule Name. Make sure Has Privilege to Execute NetMail is checked

8. Select NetMail from the Service menu and click Next. Make sure Has Privilege to Execute NetMail is checked.

9. Select the type of subject from the Type menu and click Next to complete subject configuration.

10. Choose Subjects from the View menu in the data pane and click Add. The Add Subject page opens in the data pane.

11. Click Create to complete the policy configuration.

    The message “The policy properties have been saved.” is displayed when the data is saved.

To Modify NetMail Service Attributes (Specific Organization)

You can customize the NetMail service by modifying the attributes for the service.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
2. Choose the organization.
3. Choose Services from the View menu.
4. Click the properties arrow next to NetMail in the navigation pane.

   A list of NetMail service attributes appears in the data pane.
5. Modify the service attribute values and then click Save to save the changes.

   The changes affect only users in the selected organization.

   See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for more information on the NetMail attributes.

To Modify NetMail Service Attributes (All Organizations)

Occasionally, you need to modify the global NetMail service attribute values that affect all organizations that want to add for the NetMail service in the future.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Choose Service Management in the location pane.

3. Click the properties arrow next to NetMail in the navigation pane.

   A list of NetMail service attributes appears in the data pane.

4. Modify the service attribute values then click Save to save the changes.

   The changes affect all organizations that add the NetMail service in the future.

To Configure NetMail Lite to Open a New Window

In the default configuration, if users click on the NetMail Lite link on the Desktop when they have NetMail Lite running and are composing a message, their current NetMail Lite window is replaced with a new instance of NetMail Lite and they lose the text in the message. To avoid this issue, you can configure NetMail Lite to open in a new window each time a user clicks on NetMail Lite link on the Desktop.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.

2. Choose the organization.

3. Click the properties arrow next to Desktop in the navigation pane.

   A list of Desktop service attributes appears in the data pane.

4. Click Channel and Container Management link in the data pane

5. Click the Edit link of App channel under Channels.

6. Choose the organization, and choose Services from the View menu.
7. Click the Edit link of targets property.
8. Replace the NetMail Lite property with the following:

```
NetMail Lite| ^javascript:var nmServerURL = document.URL; nmDestURL
=nmServerURL.split('dt')[0];nmAdjustedURL = nmDestURL
+'NetMailServlet?nsid=newHTMLSession';
openAppURL(nmAdjustedURL,'_blank');return false;
```

9. Click Save.
10. Verify the change.

    Log in as a test user within the organization. Access NetMail Lite and start composing a message. Click the NetMail Lite link. A new window containing NetMail Lite should open.

Using the Remote Address Book (LDAP)

To enable the remote address book feature for NetMail, you configure the LDAP server list attribute in the NetMail service.

NOTE The address book search capability enables users to search for names based on user specified text compared using the following criteria if supported by the search engine: containing, equal to, beginning with, ending with, and sounding like.

The personal address book only supports searching by contain. If you add an LDAP address book, you will see these other options enabled.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and Organizations is selected in the Navigation pane.
2. Choose the organization.
3. Choose Services from the View menu.
4. Click the properties arrow next to NetMail in the navigation pane.

   A list of NetMail service attributes appears in the data pane.

5. Modify the LDAP Server Details to Use in Address Book Search value. Each entry is a comma separated list of `name="value"` pairs where the valid names are:

   - `name`—The name that is shown in the Address page of NetMail (default: none)
   - `server`—The fully qualified domain name of the LDAP server (default: none)
   - `base`—The distinguished name (DN) that is used to start the search (default: "")
   - `searchin`—A comma separated list of attributes to look in (default: "cn,gn,sn")
   - `result`—The attribute that contains the email address (default: "`mail`")
   - `filter`—An additional LDAP filter to use for the search (default: ""). The syntax of the filter uses LDAP filter syntax.
   - `referral`—Value defining whether to follow LDAP referrals. The default is "`follow`"; use "" to define not to follow referrals.

   For example, to search the `Sesta` LDAP directory, use the following entry:

   `name="Sesta LDAP",server="ldap-server.sesta.com",base="dc=sesta,dc=com"`

6. Click Save.

Chapter 12

Administering the Rewriter Service

This chapter describes how to administer the Rewriter service of the Sun Java System Portal Server.

This chapter includes the following sections:

Overview of the Rewriter Service

The Sun Java System Portal Server Rewriter provides an engine for performing URL translation in markup languages and JavaScript™ code. The `URLScraperProvider` and the `XMLProvider` in the Desktop and the Sun Java™ System Portal Server: Secure Remote Access gateway service all use the Rewriter service.

Rewriter scans the content of web pages and identifies the URLs it finds on those web pages. It uses a collection of rules defined in a ruleset to determine the elements of a web page to rewrite. Once Rewriter identifies a URL it can rewrite the URL by:

Expanding Relative URLs to Absolute URLs

The URLScraperProvider is part of the core Portal Server product. In a non-gateway scenario, the URLScraperProvider can be used to expand relative URLs to absolute URLs. For example, if a user is trying to access the site:

```
<a href="../mypage.html">
```

The Rewriter translates this to:

```
<a href="http://www.yahoo.com/mail/mypage.html">
```

where http://www.yahoo.com/mail/ is the base URL of the page scraped.

URLScraperProvider Limitations

The URLScraperProvider simply tries to display a designated URL in a channel. There's no way to specify parts of a document URL (document) to display. The URLScraperProvider acts much like an HTTP client, in that it makes a request for the content of the specified URL. Just like in a browser, the target URL to scrape must be network visible, or you must have a proxy configured.

The resultant URL scraper channel, however, is not a mini-browser nor is it a frame. Therefore, if you have a link in the content, it effects the whole page, not just the channel. You should not browse inside the URL scraper channel. If you select a link within the channel the browser can interpret the link and replace the currently displayed page (your portal server Desktop) with the contents of the link location.

The appearance of the scraped channel is controlled by whatever is producing the original content. The URLScraperProvider does not modify the content at all and only displays whatever is available through the URL. Since the channel is essentially a cell in an HTML table, it can only display HTML content that is legal to appear in table cells. That is, a frameset cannot be scraped using the URLScraperProvider because a <FRAMESET> tag cannot appear within a <BODY> tag. The URLScraperProvider will also not execute JavaScript code in <HEAD> tags. Because of this, the following scraping scenarios are inappropriate for the URLScraperProvider:

- When an Edit function of some sort is required so that the user can customize the channel.
- When the data comes from a non-HTML, non-web server source, that is, a database or mail server.
- When the data needs to be reformatted in some way for the channel.
- When a more efficient solution is required as the URLScraperProvider will do a request and look up for every Desktop display and user.

When cookies are sent by the origin server, they are forwarded back every time web content is re-scraped. So the origin should get the cookies it sent as the web content scraped the first time, when portal desktop is updated or reloaded. But those cookies are not expected to be sent back when user clicks on any links in the url scraper channel.

Prefixing the Gateway URL to an Existing URL

In an implementation with a gateway such as the Sun Java System Portal Server: Secure Remote Access, the gateway acts as a proxy for the client and accesses intranet sites and returns responses to the client. The Rewriter translates URLs in downloaded pages so that they point back to the gateway rather than to the original site by prefixing the gateway URL to the existing URL.

For example, if a user tries to access an HTML page on `mymachine` using the following URL:

```
<a href="http://mymachine.intranet.com/mypage.html>
```

The Rewriter prefixes this URL with a reference to the gateway as follows:

```
<a
href="https://gateway.company.com/http://mymachine.intranet.com/mypage.
html>
```

When a user selects a link associated with this anchor, the browser contacts the gateway. The gateway fetches the content of `mypage.html` from `mymachine.intranet.com`.

See the *Sun Java System Portal Server: Secure Remote Access 6 2005Q1 Administration Guide* for more information on using the Rewriter to prefix a gateway URL to an existing URL.

Supported URLs

Rewriter supports rewriting of all standard URLs as specified by RFC-1738. These URLs are supported whether the protocol is HTTP or HTTPS and regardless of the capitalization of the protocol. For example, hTtP, HTtp, and httP are all valid. Some sample standard URLs are listed below:

```
http://www.my.sesta.com
http://www.example.org:8000/imaginary/test
http://www.example.edu/org/admin/people#andy
http://info.example.org/AboutUs/Index/Phonebook?dobbins
http://www.example.org/RDB/EMP?*%20where%20name%%3Dobbins
http://info.example.org/AboutUs/Phonebook
http://user:password@example.com
```

Rewriter supports rewriting of some basic non-standard URLs. The information to convert non-standard URLs to a standard format is taken from the base URL of the page where the URL appears and can include the protocol, host name, and path. The back slash (\) is supported only when it is part of a relative URL and not part of an absolute URL. For example, `http://sesta.com\index.html` is rewritten, but `http:\\sesta.com` is not.

In addition, URLs with a single slash (/) after the protocol or scheme such as `http:/sesta.com` are not rewritten.

Defining Rewriter Rules and Rulesets

The Rewriter modifies the URL portions of various elements that appear on a web page. The Rewriter comes with a default set of rules to determine the elements of a web page to rewrite. A collection of rules for various categories and subcategories is stored in a `.dtd` file and is called a ruleset. The Rewriter rulesets are defined in XML.

The DTD is located in `/opt/SUNWps/web-src/WEB-INF/lib/rewriter.jar` (`resources/RuleSet.dtd`). Rulesets are used to identify URLs. By default, all strings in web content starting with characters such as "/", `../`, "http" and "https" are considered to be URLs and are candidates for rewriting.

To configure the Rewriter for your implementation, you create a ruleset and define rules in the Rewriter section of the Portal Server Configuration in the administration console. See “Administering the Rewriter Service” for details on creating and modifying rulesets. You define multiple rules based on the content type in the web pages. For example, the rule required to rewrite HTML content would be different from the rule required to rewrite JavaScript content. Rewriter rules fall into the following broad categories:

- Rules for HTML Content
- Rules for JavaScript Content

- Rules for XML Content

NOTE As Wireless Markup Language (WML) is similar to HTML, HTML rules are applied for WML content.

No rules are required for CSS content.

The ruleset is an XML document and the XML within it must be properly formed. When defining rules in a ruleset, keep the following guidelines in mind:

- All rules need to be enclosed within the `<ruleset> </ruleset>` tags.
- Include all rules to rewrite HTML content in the `<HTML> </HTML>` section of the ruleset.
- Include all rules to rewrite JavaScript content in the `<JSRules> </JSRules>` section of the ruleset.
- Include all rules to rewrite XML content in the `<XML> </XML>` section of the ruleset.

Rules for HTML Content

HTML content in web pages can be classified into attributes, JavaScript tokens, forms, and applets. Accordingly, the rules for HTML content are classified as:

- Attribute Rules for HTML Content
- JavaScript Token Rules for HTML Content
- Form Rules for HTML Content
- Applet Rules for HTML Content

Attribute Rules for HTML Content

Attribute rules identify the basic attribute tags in HTML pages to rewrite. Rewriter modifies the various occurrences of the defined tags by expanding or prefixing the existing URL. The default ruleset rewrites the following attribute tags:

- action
- background
- codebase
- code

- href
- src
- value
- imagePath
- lowsrc
- archive

The syntax for attribute rules is:

```
<Attribute name="name" [tag="tag" valuePatterns="patterns"]
```

where *name* specifies the attribute, *tag* specifies the tag to which the attribute belongs (set to * to match all tags), and *patterns* specifies the possible patterns to match with the attribute. The `tag` and `valuePatterns` parameters are optional.

JavaScript Token Rules for HTML Content

Web pages can contain pure JavaScript code within the JavaScript tags, or they can contain JavaScript tokens or functions. For example, a web page can contain an `onClick()` function that causes a jump to a different URL. In order for the page to function properly, the value of the `onClick()` function needs to be translated and rewritten. In most cases, the rules provided in the default ruleset are sufficient to rewrite the URLs in JavaScript tokens. The default ruleset rewrites the following JavaScript tokens:

- `onAbort`
- `onBlur`
- `onChange`
- `onClick`
- `onDblClick`
- `onError`
- `onFocus`
- `onKeyDown`
- `onKeyPress`
- `onKeyUp`
- `onLoad`

- onMouseDown
- onMouseMove
- onMouseOut
- onMouseOver
- onMouseUp
- onReset
- onSelect
- onSubmit
- onUnload

The syntax for JavaScript Token rules is:

<JSToken>*javascript_function_name*</JSToken>

where *javascript_function_name* is the name of the function such as onLoad or onClick.

Form Rules for HTML Content

Users can browse HTML pages that contain forms. Form elements, such as input, can take a URL as a value. The default ruleset does not rewrite any form elements. The syntax for form rules is:

<Form source="*/source.html*" name="*form1*" field="*field1*"> [valuePatterns="*pattern*"] />

where */source.html* is the URL of the HTML page containing the form, *form1* is the name of the form, *field1* is the field of the form to be rewritten, and *pattern* indicates the part of the field to be rewritten. All content that follows the pattern specified is rewritten.

The valuePatterns parameter is optional.

Applet Rules for HTML Content

A single web page can contain many applets, and each applet can contain many parameters. The Rewriter rule for URLs in applets should contain pattern matching information for the following:

- source, such as filename.htm
- code, such as classname.class
- parameter name, such as servername

- parameter value, such as `some_url`

Rewriter matches the values specified in the rule with the content of the applet and modifies the URLs as required. This replacement is carried out at the server and not when the user is browsing the particular web page. A wildcard character (*) can also be used as part of the rule. For example, the parameter name could be *, in which case, the Rewriter does not compare the parameter name in the applet.

The default ruleset does not rewrite any applet parameters.

The syntax for applet rules is:

`<Applet source="`*sourcehtml.jsp*`" code="`*class*`" param="`*parameter_name*`" [valuePatterns="`*pattern*`"]`

where */sourcehtml.jsp* is the URL containing the applet, *class* is the name of the applet class, *parameter_name* is the parameter whose value needs to be rewritten, and *pattern* indicates the part of the field to be rewritten. All content that follows the pattern specified is rewritten. The valuePatterns parameter is optional.

Rules for JavaScript Content

URLs can occur in various portions of JavaScript code. The Rewriter cannot directly parse the JavaScript code and determine the URL portion. A special set of rules needs to be written to help the JavaScript processor translate the URL.

JavaScript elements that contain URLs are classified as follows:

- JavaScript Variables
- JavaScript Function Parameters

JavaScript Variables

JavaScript variables are again classified into five categories:

- JavaScript URL Variables
- JavaScript EXPRESSION Variables
- JavaScript DHTML Variables
- JavaScript DJS (Dynamic JavaScript) Variables
- JavaScript System Variables

JavaScript URL Variables

URL variables have a URL string on the right hand side. The default ruleset rewrites the following JavaScript URL variables:

- `imgsrc`
- `location.href`
- `_fr.location`
- `mf.location`
- `parent.location`
- `self.location`

The syntax of URL variables in JavaScript content rules is:

`<Variable type="URL">`*variable_name*`</Variable>`

where *variable_name* is the name of the variable to be rewritten.

JavaScript EXPRESSION Variables

EXPRESSION variables have an expression on the right hand side. The result of this expression is a URL. The Rewriter appends a JavaScript function for converting the expression to the HTML page as it cannot evaluate such expressions. This function takes the expression as a parameter and evaluates it at the client browser.

The default ruleset rewrites the `location` JavaScript EXPRESSION variable.

The syntax of EXPRESSION variables in JavaScript content rules is:

`<Variable type="EXPRESSION">`*variable_exp*`</Variable>`

where *variable_exp* is the expression variable.

JavaScript DHTML Variables

DHTML variables are JavaScript variables that hold HTML content. The default ruleset rewrites the following JavaScript DHTML variables:

- `document.write`
- `document.writeln`

The syntax of DHTML variables in JavaScript content is:

`<Variable type="DHTML">`*variable*`</Variable>`

where *variable* is the DHTML variable.

JavaScript DJS (Dynamic JavaScript) Variables

DJS (Dynamic JavaScript) variables are JavaScript variables that hold JavaScript content.

The syntax of DJS variables in JavaScript content is:

`<Variable type="DJS">`*variable*`</Variable>`

where *variable* is the DJS variable.

The JavaScript code contained in the variable needs another rule to translate it.

JavaScript System Variables

System variables are variables that are not declared by the user, but that are available as a part of the JavaScript standard.

The default ruleset rewrites the `window.location.pathname` JavaScript system variable.

The syntax of system variables in JavaScript content is:

`<Variable type="SYSTEM">`*variable*`</Variable>`

where *variable* is the system variable.

JavaScript Function Parameters

Function parameters are classified into four categories:

- JavaScript URL Parameters
- JavaScript EXPRESSION Parameters
- JavaScript DHTML Parameters
- JavaScript DJS Parameters

JavaScript URL Parameters

URL parameters are string parameters that directly contain the URL.

The default ruleset rewrites the following JavaScript URL parameters:

- `openURL`
- `openAppURL`
- `openNewWindow`
- `parent.openNewWindo`
- `window.open`

The syntax for URL parameters is:

```
<Function type = "URL" name = "function" [paramPatterns="y,y,"] />
```

where *function* is the name of the function to be evaluated and *y* indicates the position of the parameter(s) that need to be rewritten. Parameter positions are delimited by commas. For example, in the syntax line the first and second parameters need to be rewritten, but the third parameter should not be rewritten.

JavaScript EXPRESSION Parameters

EXPRESSION parameters are variables within a function that result in a URL when they are evaluated. The syntax for EXPRESSION parameters is

```
<Function type = "EXPRESSION" name = "function" [paramPatterns="y,y,"] />
```

where *function* is the name of the function to be evaluated and *y* indicates the position of the parameter(s) that need to be rewritten. Parameter positions are delimited by commas. For example, in the syntax line the first and second parameters need to be rewritten, but the third parameter should not be rewritten.

JavaScript DHTML Parameters

DHTML parameters are native JavaScript methods that generate an HTML page dynamically. For example, the `document.write()` method falls under this category.

The default ruleset rewrites the following JavaScript DHTML parameters:

- `document.write`
- `document.writeln`

The syntax for DHTML parameters is:

```
<Function type = "DHTML" name = "function" [paramPatterns="y,y,"] />
```

where *function* is the name of the function to be evaluated and *y* indicates the position of the parameter(s) that need to be rewritten. Parameter positions are delimited by commas. For example, in the syntax line the first and second parameters need to be rewritten, but not the third parameter should not be rewritten.

JavaScript DJS Parameters

Dynamic JavaScript (DJS) parameters such as Cascading Style Sheets (CSS) in HTML are also translated. There are no rules defined for this translation as the URL appears only in the `url()` function of the CSS. The syntax for DJS parameters is:

```
<Function type = "DJS" name = "function" [paramPatterns="y,y,"] />
```

where *function* is the name of the function to be evaluated and *y* indicates the position of the parameter(s) that need to be rewritten. Parameter positions are delimited by commas. For example, in the syntax line the first and second parameters need to be rewritten, but not the third parameter should not be rewritten.

Rules for XML Content

Web pages can contain XML content which in turn can contain URLs and Rewriter can rewrite URLS in XML content.

XML content that contains URLs is classified as follows:

- Tag Text in XML
- Attributes in XML

Tag Text in XML

Rewriter translates XML content based on the tag name.

The default ruleset rewrites the following tags in XML:

- `baseroot`
- `img`

The syntax for tag text is:

`<TagText tag ="`*attribute*`" attributePatterns="name=`*src*`"/>`

where *attribute* is the name of the tag and *src* is the name of the attribute.

Attributes in XML

The rules for attributes in XML are similar to the rules for attributes in HTML. See "Attribute Rules for HTML Content" on page 269" for additional information Rewriter translates attribute values based on the attribute and tag names.

The default ruleset rewrites the following attributes in XML:

- `xmlns`
- `href`

The syntax for attributes in HTML is:

```
<Attributes>
    <Attribute name="attribute" [valuePatterns="name=src"/>
</Attributes>
```

where *attribute* is the name of the tag and *src* is the name of the attribute.

Administering the Rewriter Service

In Portal Server 6, the Rewriter service uses Sun Java System Access Manager attributes to provide persistent storage for the Rewriter rulesets. A Rewriter ruleset defines how contents in a web page should be rewritten by the Rewriter. Multiple Rewriter rulesets can be defined and stored as Sun Java System Access Manager service attribute values through the Sun Java System Access Manager administration console.

You can also administer the Rewriter using the command line. See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for more information on the `rwadmin` command.

Because the Sun Java System Access Manager administration console does not have any concept of a rewriter ruleset, Portal Server uses a customized service management plug-in module to manage them. All Rewriter rulesets are global to the organizations in Sun Java System Access Manager. There is no provision to enable the creation of ruleset at any particular organization level.

NOTE The `URLScraperProvider` can only scrape content that is valid inside of an HTML table cell. If the HTML markup to scrape contains markup that cannot be rendered in a table cell, such as `<body>`, `<base>`, and certain JavaScript procedures, that cannot be rendered within a table cell, the display of the Desktop page can be corrupted. When defining content to scrape, try to confirm the content is valid HTML. See "URLScraperProvider Limitations" for further information.

To Configure the Rewriter URLScraperProvider for SSL

You can use the Rewriter's `URLScraperProvider` to scrape SSL pages and rewrite the URLs for access over a secure session.

1. Initialize the trust database in the web server administration console for the server on which you installed Portal Server as follows:
   a. From a browser, enter the following URL to access the Web Server admin page:

      http://*servername***:8088**

   b. Log in as Admin and click the Security tab.
   c. Enter the Database password twice and select OK.
2. Create a password file as follows:
   a. Change directories to */AccessManager-base*/SUNWam/config.
   b. Create a hidden text file .wtpass.
   c. Type the password that you gave when you initialized the trust database.
3. Add the following line to the */AccessManager-base*/SUNWam/lib/AMConfig-*instance_nickname*.properties file if the root CA is not installed for the certificates used by the Web servers accessed using the URLScraperProvider.

   ```
   com.sun.am.jssproxy.trustAllServerCerts=true
   ```

   This option tells JSS to trust the certificate.
4. Restart Portal Server.

To Create a New Ruleset from the Default Template

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.
2. Select Service Configuration from the location pane.
3. Click the properties arrow next to Rewriter in the navigation pane.

   A list of currently defined rulesets appears in the data pane.

4. Click New.

   This displays a ruleset template for possible modifications.

5. Edit the `<RuleSet id="ruleset_template">` line, replacing `default_ruleset` with the name for the new ruleset.

6. Add or modify the rules within the ruleset template to rewrite URLs as necessary.

7. Click Save to create the new ruleset.

   Upon success, you see the initial page and the list of all currently defined rulesets, which should include the one you just created.

To Edit an Existing Ruleset

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

2. Select Service Configuration from the location pane.

3. Click the properties arrow next to Rewriter in the navigation pane.

   A list of currently defined rulesets appears in the data pane.

4. Click the Edit link for the ruleset to edit.

   This displays the XML for the ruleset to edit.

5. Add or modify the rules within the ruleset template to rewrite URLs as necessary.

6. If you would like to change the name of the ruleset, edit the `<RuleSet id="ruleset_template">` line, replacing name with a name for the ruleset.

7. Click Save.

To Download a Ruleset

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

Rulesets can be downloaded and saved to a file.

1. Log in to the Sun Java System Access Manager administration console as administrator.
2. Select Service Configuration from the location pane.
3. Click the properties arrow next to Rewriter in the navigation pane.

   A list of currently defined rulesets appears in the data pane.
4. Click the Download link for the ruleset to save to a file.
5. Specify a name for the file and save it.

To Upload a Ruleset

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

A ruleset file can be uploaded into the system.

1. Log in to the Sun Java System Access Manager administration console as administrator.
2. Select Service Configuration from the location pane.
3. Click the properties arrow next to Rewriter in the navigation pane.

   A list of currently defined rulesets appears in the data pane.
4. Click the Upload link next to any ruleset in the list.
5. Browse to or type the file name for the ruleset to upload.

6. Click Upload.

   If the name defined in the `<RuleSet id="ruleset_template">` line within the file matches a ruleset name on the system that ruleset file will be replaced with the contents of the file. If the name defined in the `<RuleSet id="ruleset_template">` line is unique, a new ruleset will be created with that name and added to the list.

To Delete an Existing Ruleset

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.
2. Select Service Configuration from the location pane.
3. Click the properties arrow next to Rewriter in the navigation pane.

   A list of currently defined rulesets appears in the data pane.
4. Click the checkbox next to the ruleset to be deleted.

   You can select more than one ruleset.
5. Click Delete.

   A confirmation message appears.
6. Click Yes to delete the selected rulesets.

To Restore the Default Ruleset

In case you accidentally delete the default ruleset, you can restore it as follows:

```
rwadmin store --runasdn "uid=amadmin, ou=people, o=sesta.com, o=isp"
--password "testing123" /resources/DefaultRuleSet.xml
```

where "/resources/DefaultRuleSet.xml" is the location of the ruleset stored in the rewriter.jar file.

NOTE The default ruleset packaged from the installation is restored. If you have customized the default ruleset, the changes are not restored.

Chapter 13

Administering the Search Engine Service

This chapter describes how to configure and administer the Sun Java™ System Portal Server Search Engine service.

This chapter contains these sections:

Overview of the Search Engine Service

The Portal Server Search Engine is a taxonomy and database service designed to support search and browse interfaces similar to popular internet search engines such as Google, Alta Vista, and so on. Search Engine includes a robot to discover, convert, and summarize document resources. In Portal Server 6, the interface is provided by the Desktop exclusively, using JSP™ providers. Search Engine

includes administration tools for configuration editing and command-line tools for system management. Configuration settings can be defined and stored as Sun Java™ System Access Manager service attribute values through the Sun Java System Access Manager administration console.

NOTE Although the administration console permits an administrator to configure a majority of the Search Engine options, the administration console does not perform all the administrative functions available through the command line.

Search Database

Search users search through a database to locate particular resources or kinds of resources. The individual entries in the database are called resource descriptions (RDs). A Resource Description is a specific set of information about a single resource. The fields of each Resource Description are determined by the database schema.

To get RDs into the database, you can use two approaches:

- Creating RDs—This is by far the most common method, using a robot process to locate resources and generate their descriptions.
- Exchanging RDs—This method is appropriate for large, distributed network indexes. A remote system generates RDs, and the Search Engine imports those into its database.

The RDs in the Portal Server Search Engine are based on open Internet standards, such as the Summary Object Interchange Format (SOIF) and resource description messages (RDM). This ensures that the Search Engine can operate in a cross-platform enterprise environment.

Search Robots

One method of filling the database is via robots. The Search Engine uses robots to find and report on the resources in their domains. A *robot* is a small program that does two things:

- Extracts and follows links to resources (also called enumeration or crawling)
- Describes those resources and puts the descriptions in the database (also called generation or indexing)

As the system administrator, you control every aspect of these processes in a number of ways, including the following:

- When the robot runs by starting, stopping, and scheduling the robot.
- Where the robot looks for resources by defining the sites the robot visits.
- How aggressively it searches by defining the crawling attributes.
- What types of resources the robot indexes by defining filters.
- What kind of entries it creates for the database by defining the indexing attributes.

The Search Engine also provides utilities to ensure that the robot has done what you wanted.

Database Taxonomy Categories

Users interact with the Search system in two distinct ways: they can type direct queries to search the database, or they can browse through the database contents using a set of categories you design. A hierarchy of categories is sometimes called a taxonomy. Categorizing resources is like creating a table of contents for the database.

Browsing is an optional feature in a Search system. That is, you can have a perfectly useful Search system that does not include browsing by categories. You need to decide whether adding browsable categories will be useful to the users of your index, and then what kind of categories you want to create.

The resources in a Search database are assigned to categories to clarify complexity. If there is a large number of items in the database, it is helpful to group related items together. This allows users to quickly locate specific kinds of items, compare similar items, and choose which ones they want.

Such categorizing is common in the product and service indexes. Clothing catalogs divide men's, women's, and children's clothing, with each of those further subdivided for coats, shirts, shoes, and so on. An office products catalog could separate furniture from stationery, computers, and software. And advertising directories are arranged by categories of products and services.

The principles of categorical groupings in a printed index also apply to online indexes. The idea is to make it easy for users to locate resources of a certain type, so that they can choose the ones they want. No matter what the scope of the index you design, the primary concern in setting up your categories should be usability. That is, you need to know how users will use the categories. For example, if you were

designing an index for a company that has three offices in different locations, you might make your top-level categories correspond to each of the three offices. But if users are more interested in, say, functional divisions that cut across the geographical boundaries, it might make more sense to categorize resources by corporate divisions.

Once the categories are defined, you must set up rules to assign resources to categories. These rules are called *classification rules*. If you do not define your classification rules properly, users will not be able to locate resources by browsing in categories. You need to avoid categorizing resources incorrectly, but you also should avoid failing to categorize documents at all.

Documents can be assigned to multiple categories, up to a maximum number defined in the settings. Classification rules are simpler than filter rules because they don't involve any flow-control decisions. In classification rules you determine what criteria to use to assign specific categories to a resource as part of its Resource Description. A classification rule is a simple conditional statement, taking the form "if <some condition> is true, assign the resource to <a category>."

Configuring the Search Channel

This section describes how to initially configure the Search Engine service. Configuration settings can be defined and stored as Sun Java System Access Manager service attribute values through the Sun Java System Access Manager administration console.

The Search service is added globally and its configuration applies to the entire Portal Server. By default, the organization you specify during the Portal Server installation will have the Search service added. If you install the sample portal, the Search tab on the sample portal Desktop contains the search channel. This is configured for you during the Portal Server installation. However, for new organizations and for new instances you must define the Search URL.

The default behavior for a search provider user is that "No document matches" found will be displayed when the user enters a query.

You need to configure the Search server and create the document database to get search results.

To Initially Configure the Search Server

Use these steps to configure the Search provider. This is a sample way to fill in the database. You can also use the import function.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Create a new site.
   a. Click Robot.
   b. Click Sites.
   c. Click New under Manage Sites to define sites for the Robot to index.
   d. Specify the type of site (URL or domain), the site to index, and the depth for the robot to crawl.
   e. Click Create Site to use the default Search attributes or select Create and Edit Site to define the search site more completely.

      See the *Portal Server Technical Reference Guide* for more information on the search attributes that define the site.
5. Create a taxonomy.

   You can create a taxonomy using the Category Editor under Categories or by copying a sample taxonomy SOIF file to `config/taxonomy.rdm`).
6. Disable any of the default filters that you do not want to use.

   Click Robot and then Filters. Turn off any filters in the Filter Rule list you do not want to use.

7. (Optional) Create robot classification rules if you need to get document results under categories.

   You can create a create robot classification rules using the Classification Rules Editor under Categories

8. Start the robot.

   Click Robot, Overview, and then Start to start the robot.

9. Reindex the categories

   Click Categories then Reindex to reindex.

To Define the Search URL

The `searchServer` property defines the Search URL. It is automatically configured for the default organization; however, this value is not defined when new organizations are created, when new `SearchProvider` instances are created, or when the sample `dp-org.xml` is loaded manually. If users search when value is not defined, the following error message is displayed on the user's Desktop:

```
You got a com.sun.portal.search.providers.taglib.SearchTaglibException:
SearchRequest Error: search server is not defined.
```

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose the organization for which you want to define the Search URL.

3. Choose Services from the View menu

   Use the Show menu in the navigation pane and the Location path in the location pane.

4. Click the properties arrow next to Desktop in the navigation pane.

   The Desktop attributes page appears in the data pane.

5. In the Desktop page, click the Channel and Container Management link.

   The Channels page appears. At the top is the container path. The defined channels appear in a list.

6. Click the Edit Properties link beside the Search channel to be modified.

   The Properties page appears.

7. Specify the SearchURL in the Search Server property in the format:

   `http://`*`portal_server_name`***`:`***`port`***`/portal/search`**

8. Click Save.

9. To verify the Search URL, do the following:

   a. Log in to the organization for which you configured the Search URL. For example, log in to an organization named B as follows:

      `http://`*`portal_server_name`***`:`***`port`***`/amserver/ui/login?org=B`**

   b. Perform a search from the Search channel.

Administering the Search Engine

Once you have initially configured the Search Engine and generated a database, you can view and manage the Search Engine from the Sun Java System Access Manager administration console.

Viewing, Managing, and Monitoring Search Engine Operations

Search Engine operational attributes have two levels: basic and advanced.The basic settings page appears by default when the Search service is selected from the administration console. The basic settings displayed include the server root, the location of the temporary files, and the document level security. The advanced settings include the log locations for various Search Engine components and the configured log level.

In addition, the administration console allows administrators to view the log files or specific information extracted from the log files.

To View or Manage the Basic Settings

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Click Server then Settings from the menu bar.
5. View or specify the Server Root directory for the Search Engine.
6. View or specify the Temporary Files directory for the Search Engine.
7. View or specify the Document Level Security attribute.

   Off means all users have access to the RDs in the database. On indicates that the ReadACL field in the RD must be evaluated to determine if the user has permission to access the RD.
8. Click Save to record any altered attributes.

To View or Manage the Advanced Settings

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Click Server then Advanced from the menu bar.
5. View or specify the Advanced attributes

   The attributes available are: Search (rdm), Disable Search Log, Index Maintenance, RD Manager, RDM Server, and Log Level.
6. Click Save to record any altered settings.

To Monitor Search Engine Activity

The Search Engine provides a number of reports to allow you to monitor the search activity.

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

To view the various reports:

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Click Reports.
5. Click on a link in the menu bar to view a specific report.

   The following report options are available: Starting Points, Excluded URLs, Robot Advanced Reports, Log Files, and Popular Searches.

Administering the Robot

The following are some configuration and maintenance tasks you might need to do to administer the robot:

- Defining Sites
- Controlling Robot Crawling
- Filtering Robot Data
- Defining the Indexing Attributes
- Using the Robot Utilities
- Scheduling the Robot

Defining Sites

The robot finds resources and determines if (and how) to add descriptions of those resources to the database. The determination of which servers to visit and what parts of those servers to index is called site definition.

Defining the sites for the Search Engine is one of the most important jobs of the server administrator. You need to be sure you send the robot to all the servers it needs to index, but you also need to exclude extraneous sites that can fill the database and make it hard to find the correct information.

To Define Sites for the Robot to Index

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.

4. Click Robot then Sites from the menu bar.
5. To create a site:
   a. Click New.
   b. Select the type of site (url or domain).
   c. Specify the site and depth.
   d. Click Save.
6. To edit the site attributes, click the Edit link.

   This displays a form containing site attributes. See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for information on the Search Site attributes.

   e. Edit the attributes.
   f. Click Save.

Controlling Robot Crawling

The robot crawls to the various sites selected for indexing. Administrators can control how the robot searches sites by defining crawling operational parameters. Crawling parameters allow you to define the speed, completion actions, logging level, standards compliance, authentication parameters, proxy settings, maximum number of links to follow, and other settings. See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for descriptions of the robot crawling attributes.

To Control Robot Crawling

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Click Robot and then Crawling from the menu bar.

   This displays a form containing attributes that define the Robot Crawling operational parameters and their settings. See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for information on the Robot Crawling attributes.
5. Modify the Robot Crawling attributes as necessary.

NOTE If the `server.xml` file has a proxy set up (using the `http.proxyHost=` and `http.proxyPort=` options) you must check Accepts Commands from Any Host for the Robot to run.

6. Click Save.

Filtering Robot Data

Filters allow an attribute of a resource to be compared against a filter definition to identify a resource so that it can be excluded or included by the Site definitions. The Robot comes with a number of predefined filters some of which are enabled by default. The following filters are predefined; files marked with an asterisk are enabled by default:

- Archive Files*
- Audio Files*
- Backup Files*
- Binary Files*
- CGI Files*
- Image Files*
- Java, JavaScript, Style Sheet Files*
- Log Files*
- Power Point Files
- Revision Control Files*

- Source Code Files*
- Temporary Files*
- Video Files*
- Spreadsheet Files
- Plug-in Files
- Lotus Domino Documents
- Lotus Domino OpenViews
- System Directories (UNIX)
- System Directories (NT)

To manage the filtering process, you can create new filter definitions, modify a filter definition, or enable or disable filters.

To Create a New Filter Definition

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Robot then Filters from the menu bar.
5. Click New and specify a Nick Name for the new filter.
6. In the Filter Definition, check the checkbox and specify the Filter Source, Filter by and Filter String values. You may specify as many Filter Definitions as necessary.
7. Type a description of the filter.

8. Check New Site if you would like this filter to be used when creating new sites.
9. Click the button to indicate whether to include or exclude resources that match this filter.
10. Click Save.

To Modify an Existing Filter Definition

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Robot then Filters from the menu bar.
5. Locate the Filter to modify from the Filter Rules list and click the Edit link.
6. Modify the Filter as necessary.
7. Type a description of the filter.
8. Click Save.

To Enable or Disable a Filter

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Robot then Filter from the menu bar.
5. Locate the Filter to modify from the Filter Rules list.
6. Select the button to indicate whether to turn the filter on or off.
7. Click Save.

Defining the Indexing Attributes

For each resource that passes through the robot's filters, the robot generates an RD that it places in the database. The choices you make in setting up the generation of RDs determine what users will see when they search the database. For example, you can choose to index the full text of each document or only some fixed portion of the beginning of the document.

To Define the Indexing Attributes

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.

4. Select Robot then Indexing from the menu bar.

   This displays a page containing attributes that define the Robot Indexing operational parameters and their settings. See the *Portal Server Technical Reference Guide* for information on the Robot Indexing attributes.

5. Modify the Robot Indexing attributes as necessary.

6. Click Save.

Using the Robot Utilities

The Robot includes two debugging tools or utilities:

- Site Probe—Checks for DNS aliases, server redirects, virtual servers, and the like.
- Simulator—Performs a partial simulation of robot filtering on a URL. Type one or more URLs to check and select OK. The simulator will indicate whether the listed sites would be accepted by the robot.

To Run the Site Probe Utility

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

3. Click the properties arrow next to Search in the navigation pane.

4. Select Robot then Site Probe from the menu bar.

5. Type the URL of the site to probe.

6. Click Show Advanced DNS information if you want the probe to return DNS information.

7. Click OK to start the Site Probe.

To Run the Simulator

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Robot then Simulator from the menu bar.
5. Type in one or more URLs on which to perform the simulation.
6. Select Check for DNS aliases if you would like the Simulator to check for aliases.
7. Select Check for Server Redirects (302) if you would like the Simulator to check for redirects.
8. Click OK to start the Simulator.

Scheduling the Robot

In order to keep the search data timely, the robot should search and index sites regularly. Because robot crawling and indexing can consume processing resources and network bandwidth. To avoid these resource constraints, you should schedule the robot to run during non-peak days and times. The administration console allows administrators to set up a `cron` job with the time and days to run the robot.

To Schedule the Robot

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide.*

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Robot then Schedule from the menu bar.
5. Select the time (hour and minutes) and days to start the robot.
6. Select the time and days to stop the robot.
7. Click Save.

Administering the Database

The Search Engine stores its descriptions of resources in a database. The following are some configuration and maintenance tasks you may need to perform to administer the database:

- Importing to the Database
- Editing Resource Descriptions
- Editing the Database Schema
- Defining Schema Aliases
- Viewing Database Analysis
- Reindexing the Database
- Expiring the Database
- Purging the Database

- Partitioning the Database

Importing to the Database

Normally, the items in your Search database come from the robot. You tell the robot which sites to visit, and it locates and describes all the resources it finds there. But you can also import databases of existing items, either from other Portal Server Search Engines, from iPlanet Web Servers or Netscape™ Enterprise Servers or from databases generated from other sources. Import existing databases of RDs instead of sending the robot to create them anew helps reduce the amount of network traffic and also enables large indexing efforts to be completed more quickly by breaking the effort down into smaller parts. If the central database is physically distant from the servers being indexed, it can be helpful to generate the RDs locally, then have the central database import the various remote databases periodically.

The Search Engine uses an import agent to import RDs from another server or from a database. An *import agent* is a process that retrieves a number of RDs from an external source and merges that information into the local database. It contains parameters that tell it where to go to import RDs, what to ask for when it gets there, and some other information that fine-tunes the way it goes about the job.

Before you can import a database, you must create an import agent. Once an agent is created, you can start the import process immediately or schedule a time to run the import process.

To Create an Import Agent

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

3. Click the properties arrow next to Search in the navigation pane.

4. Select Database then click the Import Agents link.
5. Click New.

   The attributes page for the import agent appears.
6. Specify the appropriate attributes for the import agent.

   See the *Portal Server Technical Reference Guide* for information on the Database Import attributes.

   a. Indicate whether the source is a local file or search server.
   b. If the source is a file, specify the local file path.
   c. If the source is another search server, specify the URL for the remote server, the instance name, and the search URI.
   d. Specify the name of the database to import.
   e. Specify the character set for the import agent.
7. Click Save.

To Edit an Existing Import Agent

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration om the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Database then click the Import Agents link.
5. Click the Edit link to the right of the agent to edit.
6. Specify the appropriate attributes for the import agent.

   See the *Portal Server Technical Reference Guide* for information on the Database Import attributes
7. Click Save.

Editing Resource Descriptions

At times you will find it necessary to change the contents of one or more Resource Descriptions. For example, you might need to correct a typographical error copied into an Resource Description from an original document.

To Edit the Resource Descriptions

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Database then Resource Descriptions from the menu bar.
5. Select the type of Resource Description to search for to edit.

   The following types are available: All RDs, Uncategorized RDs, Categorized RDs, RDs by category, Specific RD by URL, RDs that contain.
6. For Resource Descriptions that contain, specify a text string to search for in the Resource Description.
7. Click Search.
8. From the list of Resource Descriptions found, select the Resource Description to edit.
9. Edit the appropriate Resource Description attribute.
10. Click Save.

Editing the Database Schema

A schema determines what information your Search Engine maintains on each resource, and in what form. The design of your schema determines two factors that affect the usability of your index:

- The way users can search for resources
- The ways users view resource information

The schema is a master data structure for Resource Descriptions in the database. Depending on how you define and index the fields in that data structure, users will have varying degrees of access to the resources.

The schema is closely tied to the structure of the files used by the Search Engine and its robot. You should only make changes to the data structure by using the schema tools in administration console. You should never edit the schema file (`schema.rdm`) directly, even though it is a text file.

You can edit the database schema of the Search Engine to add a new schema attribute, edit a schema attribute, or delete attributes.

The schema includes the following attributes:

- Editable—If checked, this attribute indicates that the attribute appears in the Resource Description Editor, so you can change its values. The Resource Description Editor is explained in "Editing Resource Descriptions" on page 303.
- Indexable—This attribute indicates that the field appears in the pop-up menu in the Advanced Search screen. This allows users to search for values in that particular field.
- Description—This is a text string to use to describe the schema. You can use it for comments or annotations.
- Aliases—This attribute allows you to define aliases to convert imported database schema names into your own schema.

To Edit the Database Schema

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

3. Click the properties arrow next to Search in the navigation pane.

4. Select Database then Schema from the menu bar.

   The Schema attributes page appears.

5. To add a new attribute to the schema:

   a. Select New under Schema List.

   b. Type a name and description for the new attribute in the Name and Description fields.

   c. Check Editable to allow the attribute to be edited.

   d. Check Indexable to make the attribute indexed.

6. To make an existing schema attribute editable or indexable:

   a. Click the Edit link next to an attribute from the schema list.

      The Schema attributes page appears.

   b. Check Editable to allow the attribute to be edited.

   c. Check Indexable to make the attribute indexed.

   d. Click Update

7. To delete an attribute:

   a. Check an attribute from the schema list.

   b. Click Delete.

NOTE Changes to the search engine schema may require that the entire database be reindexed and the server restarted. This is because the search engine highlighting functions are sensitive to the order and types of the schema fields. Adding or removing (or even removing and then adding back again) a text field has a high likelihood of causing search result highlighting to be incorrect.

Defining Schema Aliases

There are several instances where you might encounter discrepancies between the names used for fields in database schemas. One is when you import Resource Descriptions from one server into another. You cannot always guarantee that the two servers use identical names for items in their schemas. Similarly, when the robot converts HTML META tags from a document into schema fields, the document controls the names.

The Search Engine allows you to define schema aliases for your schema attributes, to map these external schema names into valid names for fields in your database.

To Define Schema Aliases

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Database then click the Schema link.

   The Schema attributes page appears.
5. Click the attribute for which to define an alias.
6. Specify the field name of the alias as it is used in the imported database.
7. Click Update.
8. Click Reindex.

   The reindexing process may take several hours for a large database.

Viewing Database Analysis

The Search Engine provides a report with information about the number of sites indexed and the number of resources from each in the database.

To View Database Analysis Information

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Database then Analysis from the menu bar.

   A sorted list of all sites and the number of resources from that site currently in the search database.
5. To generate a up-to-date list, click Save.

Reindexing the Database

In certain instances, you might need to reindex the Resource Description database for the Search Engine. One obvious instance is if you have edited the schema to add or remove an indexed field.

You might also need to reindex the database if a disk error corrupts the index file. It's also a good idea to reindex after adding a large number of new Resource Descriptions.

Reindexing the database can take several hours.

The time required to reindex the database is proportional to the number of records in the database, so if you have a large database, you should perform reindexing at a time when the server is not in high demand.

To Reindex the Database

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Database then Management.
5. Click Reindex under Database List.
6. Check the Reindex the database? checkbox and click OK.

   The Search Engine rebuilds the search collection and its index files.

Expiring the Database

Expiring the database will expire Resource Descriptions deemed out of date. Resource Descriptions will expire ONLY when you run the expiration. The expired Resource Descriptions will be deleted, however the database size will not decreased.

To Expire the Database

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Database then Management.
5. Select Expire under Database List.
6. Check the Expire RDs? checkbox and click OK.

Purging the Database

One attribute of a Resource Description is its expiration date. Your robots can set the expiration date from HTML META tags or from information provided by the resource's server. By default, Resource Descriptions expire in three months from creation unless the resource specifies a different expiration date. Periodically your Search Engine should purge expired Resource Descriptions from its database.

Purging allows you to remove the contents of the database. Disk space used for indexes will be recovered, but disk space used by the main database will not be recovered, instead, it is reused as new data are added to the database.

To Purge Expired Resource Descriptions from a Server

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

3. Click the properties arrow next to Search in the navigation pane.

4. Select Database then Management.

5. Select Purge under Database List.

6. Check the Purge the database? checkbox and click OK.

   When the purge is complete, the system displays the message “The database contents were successfully purged.”

Partitioning the Database

The Search Engine allows you to split the physical files that contain the search database across multiple disks, file systems, directories, or partitions. By spreading the database across different physical or logical devices, you can create a larger database than would fit on a single device.

By default, the Search Engine sets up the database to use only one directory. The command-line interface allows you to perform two kinds of manipulations on the database partitions:

- Adding New Partitions
- Moving Partitions

The Search Engine does not perform any checking to ensure that individual partitions have space remaining. It is your responsibility to maintain adequate free space for the database.

You can add new database partitions up to a maximum of 15 total partitions.

NOTE Once you increase the number of partitions, you will need to delete the entire database if you later want to reduce the number again.

You can change the physical location of any of your database partitions by specifying the name of the new location. Similarly, you can rename an existing partition. Use the `rdmgr` command to manipulate the partitions. See the *Sun Java System Portal Server 2005Q1 Technical Reference Guide* for information on the `rdmgr` command.

Administering the Database Taxonomy

The following are some configuration and maintenance tasks you may need to perform to administer the database taxonomy:

- Configuring Categories
- Defining Classification Rules

Configuring Categories

Using the Sun Java System Access Manager administration console you can perform the following procedures to configure the database taxonomy:

- To Create a Subcategory (a subcategory of a parent category)
- To Update a Category
- To Delete a Category

To Create a Subcategory

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

3. Click the properties arrow next to Search in the navigation pane.
4. Select Categories then Category Editor from the menu bar.
5. Select a category in which to create a subcategory.

   If you have not previously defined any categories, only the root category titled "Search" is listed. Click the lower Search link to expand the root category.
6. In the Name field, specify a name for the category.
7. In the Description field, specify a description for the category (optional).
8. Click Add as a Subcategory to create the category.
9. Click Save.

NOTE The Category Editor has a go-to list that appears whenever the list of visible categories spans multiple pages. Use the page-up and page-down buttons to scroll up or down one page from the current page. Use the go-to button to access more than one page.

To Update a Category

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Categories then Category Editor from the menu bar.
5. Select a category to update.
6. To change the name of the category, specify a new name for the category in the Name field.

7. To change the description of the category, specify a description for the category In the Description field.
8. Click Update.
9. Click Save.

NOTE The Category Editor has a go-to list that appears whenever the list of visible categories spans multiple pages. Use the page-up and page-down buttons to scroll up or down one page from the current page. Use the go-to button to access more than one page.

To Delete a Category

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Categories then Category Editor from the menu bar.
5. Select the category to delete.

   When a category is deleted, all its subcategories will also be deleted.
6. Click Delete.
7. Click Save.

NOTE The Category Editor has a go-to list that appears whenever the list of visible categories spans multiple pages. Use the page-up and page-down buttons to scroll up or down one page from the current page. Use the go-to button to access more than one page.

Defining Classification Rules

A classification rule is a simple conditional statement. Its form is "if <some condition> is true, assign the resource to <a category>".

To Define a Classification Rule

NOTE For current and complete information on the Access Manager admin console, refer to the *Sun Java System Access Manager 2005Q1 Administration Guide*.

1. Log in to the Sun Java System Access Manager administration console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.
2. Choose Service Configuration in the location pane.
3. Click the properties arrow next to Search in the navigation pane.
4. Select Categories then Classification Rules Editor from the menu bar.
5. f you are creating a new rule, click New.
6. If you are editing an existing rule, select the rule.
7. Click the element type or attribute to use to classify the resource from the drop-down menu.
8. Click the comparison test in the drop-down menu.

   Comparison tests available are is, contains, begins with, ends with, or regular expression.
9. Define a text string to compare.
10. Click the category in which to classify the resource if the comparison is true.
11. Click save.

Chapter 14

Administering the Search Engine Robot

This chapter describes the Sun Java™ System Portal Server Search Engine robot and its corresponding configuration files. The following topics are discussed:

Search Engine Robot Overview

A Search Engine robot is an agent that identifies and reports on resources in its domains. It does so by using two kinds of filters: an enumerator filter and a generator filter.

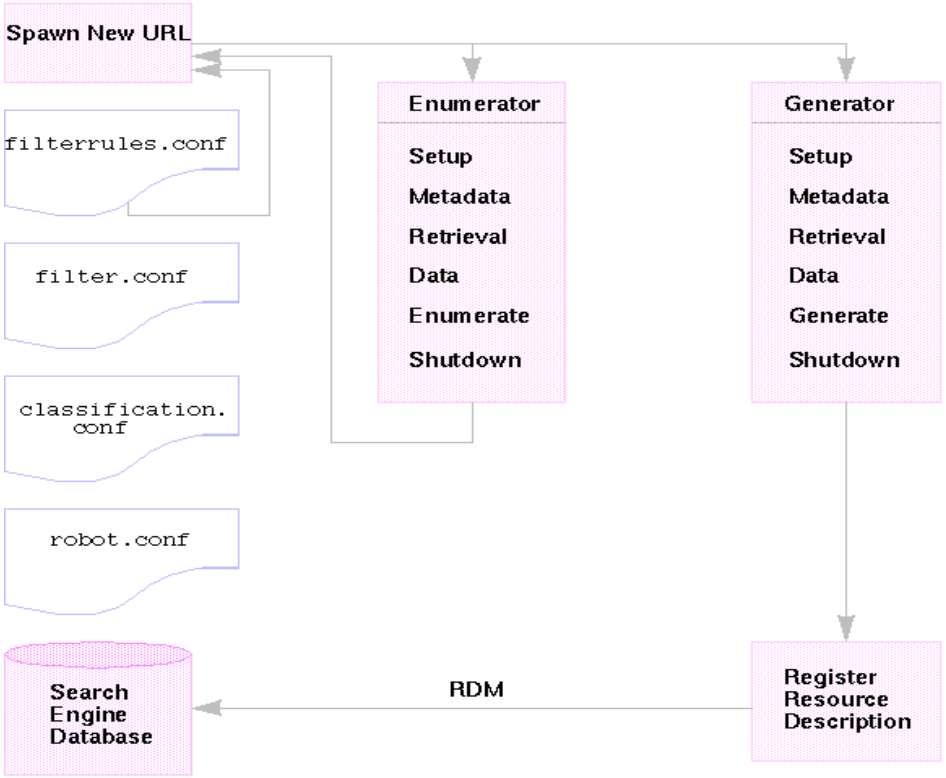
The enumerator filter locates resources by using network protocols. It tests each resource, and, if it meets the proper criteria, it is enumerated. For example, the enumerator filter can extract hypertext links from an HTML file and use the links to find additional resources.

The generator filter tests each resource to determine if a resource description (RD) should be created. If the resource passes the test, the generator creates an RD which is stored in the Search Engine database.

How the Robot Works

Figure 14-1 illustrates how the Search Engine robot works. In Figure 14-1, the robot examines URLs and their associated network resources. Each resource is tested by both the enumerator and the generator. If the resource passes the enumeration test, the robot checks it for additional URLs. If the resource passes the generator test, the robot generates a resource description that is stored in the Search Engine database.

Figure 14-1 How the Robot Works



Robot Configuration Files

Robot configuration files define the behavior of the Search Engine robots. These files reside in the directory `/var/opt/SUNWps/http-`*hostname-domain*`/portal/config`. Table 14-1 provides a description for each of the robot configuration files. The table contains two columns. The first column lists configuration file and the second column describes contents of the file.

Table 14-1 Robot Configuration Files

Robot Configuration File	Description
`classification.conf`	Contains rules used to classify RDs generated by the robot.
`filter.conf`	Contains all the filters available to the Search Engine robot for enumeration and generation. Including the same filtering rules for both the enumeration and generation filters ensures that a single rule change can be made to both types of filters. By reference, this file also includes the filtering rules stored in `filterrules.conf`.
`filterrules.conf`	Contains the starting points (also referred to as seed URLS) and filtering rules.
`robot.conf`	Defines most of the operating parameters for the robot. In addition, this file points the robot to applicable filters in the file `filter.conf`.

NOTE The Search service uses two other configuration files: `convert.conf` and `import.conf`. These files are generated by the Search server and in general should not be manually edited

Because you can set most parameters by using the Search Engine Administration Interface, you typically do not need to edit the `robot.conf` file.

However, advanced users might manually edit this file in order to set parameters that cannot be set through the interface.

Setting Robot Process Parameters

The file `robot.conf` defines many options for the robot, including pointing the robot to the appropriate filters in `filter.conf`. (For backwards-compatibility with older versions, `robot.conf` can also contain the seed URLs.)

The Sun Java System Access Manager administration console is used to edit the file `robot.conf`. Note that the few parameters you might manually edit by hand are described in detail in the "User-Modifiable Parameters" on page 322"section.

The most important parameters are `enumeration-filter` and `generation-filter`, which determine the filters the robot uses for enumeration and generation. The default values for these are `enumeration-default` and `generation-default`, which are the names of the filters provided by default in the `filter.conf` file.

All filters must be defined in the file `filter.conf`. If you define your own filters in `filter.conf`, you must add any necessary parameters to `robot.conf`.

For example, if you define a new enumeration filter named `my-enumerator`, you would add the following parameter to `robot.conf`:

```
enumeration-filter=my-enumerator
```

The Filtering Process

The robot uses filters to determine which resources to process and how to process them. When the robot discovers references to resources as well as the resources themselves, it applies filters to each resource in order to enumerate it and to determine whether or not to generate a resource description to store in the Search Engine database.

The robot examines one or more seed URLs, applies the filters, and then applies the filters to the URLs spawned by enumerating the seed URLs, and so on. The seed URLs are defined in the `filterrules.conf` file.

A filter performs any required initialization operations and applies comparison tests to the current resource. The goal of each test is to either allow or deny the resource. A filter also has a shutdown phase during which it performs any required cleanup operations.

If a resource is allowed, that means that it is allowed to continue passage through the filter. If a resource is denied, then the resource is rejected. No further action is taken by the filter for resources that are denied. If a resource is not denied, the robot will eventually enumerate it, attempting to discover further resources. The generator might also create a resource description for it.

These operations are not necessarily linked. Some resources result in enumeration; others result in RD generation. Many resources result in both enumeration and RD generation. For example, if the resource is an FTP directory, the resource typically will not have an RD generated for it. However, the robot might enumerate the individual files in the FTP directory. An HTML document that contains links to other documents can receive an RD and can lead to enumeration of the linked documents as well.

The following sections detail the filter process:

- Stages in the Filter Process
- Filter Syntax
- Filter Directives
- Writing or Modifying a Filter

Stages in the Filter Process

Both enumerator and generator filters have five phases in the filtering process. They both have four common phases:

The phases are as follows:

- Setup—Performs initialization operations. Occurs only once in the life of the robot.
- Metadata—Filters the resource based on metadata that is available about the resource. Metadata filtering occurs once per resource before the resource is retrieved over the network. Table 14-2 lists examples of common metadata types. The table contains three columns. The first column lists the metadata type, the second column provides a description, and the third column provides an example.

Table 14-2 Common Metadata Types

Metadata	Description	Example
Complete URL	The location of a resource	`http://home.siroe.com/`
Protocol	The access portion of the URL	http, ftp, file
Host	The address portion of the URL	`www.siroe.com`
IP address	Numeric version of the host	198.95.249.6
PATH	The path portion of the URL	`/index.html`
Depth	Number of links from the seed URL	5

- Data—Filters the resource based on its data. Data filtering is done once per resource after it is retrieved over the network. Data that can be used for filtering include:
  - content-type
  - content-length
  - content-encoding
  - content-charset
  - last-modified
  - expires
- Enumerate—Enumerates the current resource in order to determine if it points to other resources to be examined.
- Generate—Generates a resource description (RD) for the resource and saves it in the Search Engine database.
- Shutdown—Performs any needed termination operations. Occurs once in the life of the robot.

Filter Syntax

The `filter.conf` file contains definitions for enumeration and generation filters. This file can contain multiple filters for both enumeration and generation. Note that the robot can determine which filters to use because they are specified by the `enumeration-filter` and `generation-filter` parameters in the file `robot.conf`.

Filter definitions have a well-defined structure: a header, a body, and an end. The header identifies the beginning of the filter and declares its name, for example:

```
<Filter name="myFilter">
```

The body consists of a series of *filter directives* that define the filter's behavior during setup, testing, enumeration or generation, and shutdown. Each directive specifies a function, and if applicable, parameters for the function.

The end is marked by `</Filter>`.

Code Example 14-1 on page 321 shows a filter named `enumeration1`

Code Example 14-1 Enumeration File Syntax

```
<Filter name="enumeration1>
   Setup fn=filterrules-setup config=./config/filterrules.conf
#  Process the rules
   MetaData fn=filterrules-process
#  Filter by type and process rules again
   Data fn=assign-source dst=type src=content-type
   Data fn=filterrules-process
#  Perform the enumeration on HTML only
   Enumerate enable=true fn=enumerate-urls max=1024 type=text/html
#  Cleanup
   Shutdown fn=filterrules-shutdown
</Filter>
```

Filter Directives

Filter directives use Robot Application Functions (RAFs) to perform operations. Their use and flow of execution is similar to that of NSAPI directives and Server Application Functions (SAFs) in the file `obj.conf`. Like NSAPI and SAF, data are stored and transferred using parameter blocks, also called *pblocks*.

There are six robot directives, or RAF classes, corresponding to the filtering phases and operations listed in "The Filtering Process" on page 318:

- Setup
- Metadata
- Data
- Enumerate
- Generate
- Shutdown

Each directive has its own robot application functions. For example, use filtering functions with the Metadata and Data directives, enumeration functions with the Enumerate directive, generation functions with the Generate directive, and so on.

The built-in robot application functions, as well as instructions for writing your own robot application functions, are explained in the *Sun Java System Portal Server 6 2004Q2 Developer's Guide*.

Writing or Modifying a Filter

In most cases, you should not need to write filters from scratch. You can create most of your filters using the administration console. You can then modify the `filter.conf` and `filterrules.conf` files to make any desired changes. These files reside in the directory `/var/opt/SUNWps/http-`*hostname-domain*`/portal`.

However, if you want to create a more complex set of parameters, you will need to edit the configuration files used by the robot.

Note the following points when writing or modifying a filter:

- The order of execution of directives (especially the available information at each phase)
- The order of rules

For a discussion of the parameters you can modify in the file `robot.conf`, the robot application functions that you can use in the file `filter.conf`, and how to create your own robot application functions, see the *Sun Java System Portal Server 6 2004Q2 Developer's Guide*.

User-Modifiable Parameters

The `robot.conf` file defines many options for the robot, including pointing the robot to the appropriate filters in `filter.conf`. For backwards-compatibility with older versions, `robot.conf` can also contain the seed URLs.

Because you can set most parameters by using the administration console, you typically do not need to edit the `robot.conf` file. However, advanced users might manually edit this file in order to set parameters that cannot be set through the administration console. See "Sample robot.conf File" on page 328 for an example of this file.

Table 14-3 on page 323 lists the user-modifiable parameters in the `robot.conf` file. The first column of the table lists the parameter, the second column provides a description of the parameter, and the third column provides an example.

Table 14-3 User-Modifiable Parameters

Parameter	Description	Example
auto-proxy	Specifies the proxy setting for the robot. It can be a proxy server or a JavaScript file for automatically configuring the proxy. For more information see, the *Sun Java System Portal Server 6 2005Q1 Administration Guide*.	`auto-proxy="http://proxy_server/proxy.pac"`
bindir	Specifies whether the robot will add a bind directory to the `PATH` environment. This is an extra `PATH` for users to run an external program in a robot, such as those specified by `cmd-hook` parameter.	`bindir=path`
cmd-hook	Specifies an external completion script to run after the robot completes one run. This must be a full path to the command name. The robot will execute this script from the `/var/opt/SUNWps/` directory. There is no default. There must be at least one RD registered for the command to run. For information about writing completion scripts, see the *Sun Java System Portal Server 6 2004Q2 Developer's Guide*.	`cmd-hook="command-string"` There is no default.
command-port	Specifies the socket that the robot listens to in order to accept commands from other programs, such as the Administration Interface or robot control panels. For security reasons, the robot can accept commands only from the local host unless `remote-access` is set to `yes`.	command-port=port_number
connect-timeout	Specifies the maximum time allowed for a network to respond to a connection request. The default is 120 seconds.	command-timeout=seconds

Table 14-3 User-Modifiable Parameters

Parameter	Description	Example
convert-timeout	Specifies the maximum time allowed for document conversion. The default is 600 seconds.	convert-timeout=seconds
depth	Specifies the number of links from the seed URLs (also referred to as starting point) that the robot will examine. This parameter sets the default value for any seed URLs that do not specify a depth. The default is 10. A value of negative one (depth=-1) indicates that the link depth is infinite.	`depth=integer`
email	Specifies the email address of the person who runs the robot. The email address is sent with the user-agent in the HTTP request header, so that Web managers can contact the people who run robots at their sites. The default is *user@domain*.	`email=user@hostname`
enable-ip	Generates an IP address for the URL for each RD that is created. The default is *true*.	`enable-ip=[true | yes | false | no]`
enable-rdm-probe	Determines if the server supports RDM, the robot decides whether to query each server it encounters by using this parameter. If the server supports RDM, the robot will not attempt to enumerate the server's resources, since that server is able to act as its own resource description server. The default is *false*.	`enable-rdm-probe=[true | false | yes | no]`
enable-robots-txt	Determines if the robot should check the `robots.txt` file at each site it visits, if available. The default is yes.	`enable-robots-txt=[true | false | yes | no]`

Table 14-3 User-Modifiable Parameters

Parameter	Description	Example
engine-concurrent	Specifies the number of pre-created threads for the robot to use. The default is 10. This parameter cannot be set interactively through the administration console.	`engine-concurrent=[1..100]`
enumeration-filter	Specifies the enumeration filter that is used by the robot to determine if a resource should be enumerated. The value must be the name of a filter defined in the file `filter.conf`. The default is `enumeration-default`. This parameter cannot be set interactively through the administration console.	`enumeration-filter=enumfiltername`
generation-filter	Specifies the generation filter that is used by the robot to determine if a resource description should be generated for a resource. The value must be the name of a filter defined in the file `filter.conf`. The default is `generation-default`. This parameter cannot be set interactively through the administration console.	`generation-filter=genfiltername`
index-after-ngenerated	Specifies the number of minutes that the robot should collect RDs before batching them for the Search Engine. If you do not specify this parameter, it is set to 256 minutes.	`index-after-ngenerated=30`

Table 14-3 User-Modifiable Parameters

Parameter	Description	Example
loglevel	Specifies the levels of logging. The `loglevel` values are as follows: • Level 0: log nothing but serious errors • Level 1: also log RD generation (default) • Level 2: also log retrieval activity • Level 3: also log filtering activity • Level 4: also log spawning activity • Level 5: also log retrieval progress The default value is 1.	`loglevel=[0...100]`
max-connections	Specifies the maximum number of concurrent retrievals that a robot can make. The default is 8.	`max-connections=[1..100]`
max-filesize-kb	Specifies the maximum file size in kilobytes for files retrieved by the robot.	`max-filesize-kb=1024`
max-memory-per-url / max-memory	Specifies the maximum memory in bytes used by each URL. If the URL needs more memory, the RD is saved to disk. The default is 1. This parameter cannot be set interactively through the administration console.	`max-memory-per-url=n_bytes`
max-working	Specifies the size of the robot working set, which is the maximum number of URLs the robot can work on at one time. This parameter cannot be set interactively through the administration console.	`max-working=1024`

Table 14-3 User-Modifiable Parameters

Parameter	Description	Example
onCompletion	Determines what the robot does after it has completed a run. The robot can either go into idle mode, loop back and start again, or quit. The default is `idle`. This parameter works with the `cmd-hook` parameter. When the robot is done, it will do the action of `onCompletion` and then run the `cmd-hook` program.	`OnCompletion=[idle | loop | quit]`
password	Specifies the `password` is used for httpd authentication and ftp connection.	`password=string`
referer	Specifies the parameter sent in the HTTP request if it is set to identify the robot as the referer when accessing Web pages	`referer=string`
register-user and register-password	Specifies the user name used to register RDs to the Search Engine database. This parameter cannot be set interactively through the Search Engine Administration Interface.	`register-user=string`
register-password	Specifies the password used to register RDs to the Search Engine database. This parameter cannot be set interactively through the administration console.	`register-password=string`
remote-access	This parameter determines if the robot can accept commands from remote hosts. The default is `false`.	`remote-access=[true | false | yes | no]`
robot-state-dir	Specifies the directory where the robot saves its state. In this working directory, the robot can record the number of collected RDs and so on.	`robot-state-dir="/var/opt/SUNWps/instance/portal/robot"`
server-delay	Specifies the time period between two visits to the same web site, thus preventing the robot from accessing the same site too frequently.	`server-delay=delay_in_seconds`

Table 14-3 User-Modifiable Parameters

Parameter	Description	Example
site-max-connections	Indicates the maximum number of concurrent connections that a robot can make to any one site. The default is 2.	`site-max-connections=[1..100]`
smart-host-heuristics	Enables the robot to change sites that are rotating their DNS canonical host names. For example, `www123.siroe.com` is changed to `www.siroe.com`. The default is `false`.	`smart-host-heuristics=[true | false]`
tmpdir	Specifies a place for the robot to create temporary files. Use this value to set the environment variable `TMPDIR`.	`tmpdir=path`
user-agent	Specifies the parameter sent with the email address in the `http-request` to the server.	`user-agent=iPlanetRobot/4.0`
username	Specifies the user name of the user who runs the robot and is used for httpd authentication and ftp connection. The default is `anonymous`.	`username=string`

Sample robot.conf File

This section describes a sample `robot.conf` file. Any commented parameters in the sample use the default values shown. The first parameter, `csid`, indicates the Search Engine instance that uses this file; it is important not to change the value of the this parameter. See "User-Modifiable Parameters" on page 322 for definitions of the parameters in this file.

NOTE This sample file includes some parameters used by the Search Engine that you should not modify such as the `csid` parameter.

```
<Process csid="x-catalog://budgie.siroe.com:80/jack" \
   auto-proxy="http://sesta.varrius.com:80/"
   auto_serv="http://sesta.varrius.com:80/"
   command-port=21445
   convert-timeout=600
   depth="-1"
   # email="user@domain"
   enable-ip=true
   enumeration-filter="enumeration-default"
   generation-filter="generation-default"
   index-after-ngenerated=30
   loglevel=2
   max-concurrent=8
   site-max-concurrent=2
   onCompletion=idle
   password=boots
   proxy-loc=server
   proxy-type=auto
   robot-state-dir="/var/opt/SUNWps/https-budgie.siroe.com/ \
   ps/robot"
   server-delay=1
   smart-host-heuristics=true
   tmpdir="/var/opt/SUNWps/https-budgie.siroe.com/ps/tmp"
   user-agent="iPlanetRobot/4.0"
   username=jack
</Process>
```

Chapter 15

The Pre-defined Robot Application Functions

This chapter provides descriptions, parameter specifications, and examples of pre-defined Robot Application Functions (RAFs) in the Sun Java™ System Portal Server Search Engine. You can use these functions in the `filter.conf` file to create and modify filter definitions. The file `filter.conf` is located in the directory `/var/opt/SUNWps/http-`*hostname-domain*`/portal/config`.

The file `filter.conf` contains definitions for the enumeration and generation filters. Each of these filters invokes a set of rules which are stored in the file `filterrules.conf`. The filter definitions contain instructions that are specific to each filter while the filter rules contain the rules used by both filters.

To understand how filter rules are defined, examine the file `filterrules.conf`. Note that you typically need not manually edit this file since you create filter rules by using the administration console.

To see an example of filter definitions, you should examine the file `filter.conf`. You only need to edit the `filter.conf` file to modify the filters in a way that is not accommodated in the administration console, such as instructing the robot to enumerate some resources without generating resources for them.

This chapter contains the following sections:

- Shutdown Functions

Sources and Destinations

Most of the Robot Application Functions (RAFs) require sources of information and generate data that goes to destinations. The sources are defined within the robot itself and are not necessarily related to the fields in the resource description it ultimately generates. Destinations, on the other hand, are generally the names of fields in the resource description, as defined by the resource description server's schema.

For details on using the administration console to determine the database schema, see Chapter 13, "Administering the Search Engine Service"

The following sections describe the different stages of the filtering process, and the sources available at those stages.

Sources Available at the Setup Stage

At the Setup stage, the filter is set up and cannot yet get information about the resource's URL or content.

Sources Available at the MetaData Filtering Stage

At the MetaData stage, the robot encounters a URL for a resource, but it has not downloaded the resource's content, thus information is available about the URL as well as data that is derived from other sources such as the `filter.conf` file. At this stage, however, information is not available about the content of the resource.

Table 15-1 lists the sources available to the RAFs at the MetaData phase. The table contains three columns. The first column lists the source, the second column provides a description, and the third column provides an example.

Table 15-1 Sources Available to the RAFs at the MetaData Phase

Source	Description	Example
csid	Catalog Server ID	`x-catalog//budgie.siroe.com:8086/alexandria`

Table 15-1 Sources Available to the RAFs at the MetaData Phase *(Continued)*

Source	Description	Example
depth	Number of links traversed from starting point	`10`
enumeration filter	Name of Enumeration filter	`enumeration1`
generation filter	Name of Generation filter	`generation1`
host	Host portion of URL	`home.siroe.com`
IP	Numeric version of host	`198.95.249.6`
protocol	Access portion of the URL	`http, https, ftp, file`
path	Path portion of the URL	`/, /index.html, /documents/listing.html`
URL	Complete URL	`http://developer.siroe.com/docs/manuals/`

Sources Available at the Data Stage

At the Data stage, the robot has downloaded the content of the resource at the URL, and can access data about the content, such as the description, the author, and so on.

If the resource is an HTML file, the Robot parses the `<META>` tags in the HTML headers. Consequently, any data contained in `<META>` tags is available at the Data stage.

During the data phase, the following sources are available to RAFs, in addition to those available during the MetaData phase. The table contains three columns. The first column lists the source, the second column provides a description, and the third column provides an example.

Table 15-2 Sources Available to the RAFs at the Data Phase

Source	Description	Example
`content-charset`	Character set used by the resource	
`content-encoding`	Any form of encoding	
`content-length`	Size of the resource in bytes	
`content-type`	MIME type of the resource	`text/html, image/jpeg`
`expires`	Date the resource itself expires	

Table 15-2 Sources Available to the RAFs at the Data Phase *(Continued)*

Source	Description	Example
`last-modified`	Date the resource was last modified	
`data in <META> tags`	Any data that is provided in `<META>` tags in the header of HTML resources	Author Description Keywords

All these sources (except for the data in `<META>` tags) are derived from the HTTP response header returned when retrieving the resource.

Sources Available at the Enumeration, Generation, and Shutdown Stages

At the Enumeration and Generation stages, the same data sources are available as the Data stage.

At the Shutdown stage, the filter completes its filtering and is shuts down. Although functions written for this stage can use the same data sources as those available at the Data stage, the shutdown functions typically restrict their operations to shutdown and cleanup activities.

Enable Parameter

Each function can have an enable parameter. The values can be `true`, `false`, `on`, or `off`. The administration console uses these parameters to turn certain directives on or off.

The following example enables enumeration for `text/html` and disables enumeration for `text/plain`:

```
#  Perform the enumeration on HTML only
Enumerate enable=true fn=enumerate-urls max=1024 type=text/html
Enumerate enable=false fn=enumerate-urls-from-text max=1024 type=text/plain
```

Adding an `enable=false` parameter or an `enable=off` parameter has the same effect as commenting the line. Because the administration console does not write comments, it writes an `enable` parameter instead.

Setup Functions

This section describes the functions that are used during the setup phase by both enumeration and generation filters. The following functions are described:

filterrules-setup

When you use the `filterrules-setup` function, `logtype` is the type of log file to use. The value can be `verbose`, `normal`, or `terse`.

Parameters

Table 15-3 lists the parameter used with the `filterrules-setup` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-3 `filterrules-setup` Parameters

Parameter	Description
`config`	Path name to the file containing the filter rules to be used by this filter.

Example

```
Setup fn=filterrules-setup config=./config/filterrules.conf
logtype=normal
```

setup-regex-cache

The `setup-regex-cache` function initializes the cache size for the `filter-by-regex` and `generate-by-regex` functions. Use this function to specify a number other than the default of 32.

Parameters

Table 15-4 lists the parameter used with the setup-regex-cache function. The table contains three columns. The first column lists the parameter, the second column provides a description, and the third column provides an example.

Table 15-4 setup-regex-cache Parameter

Parameter	Description
cache-size	Maximum number of compiled regular expressions to be kept in the regex cache.

Example

```
Setup fn=setup-regex-cache cache-size=28
```

setup-type-by-extension

The setup-type-by-extension function configures the filter to recognize file name extensions. It must be called before the assign-type-by-extension function can be used. The file specified as a parameter must contain mappings between standard MIME content types and file extension strings.

Parameters

Table 15-5 lists the parameter used with the setup-type-by-extension function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-5 setup-type-by-extension Parameter

Parameter	Description
file	Name of the MIME types configuration file.

Example

```
Setup fn=setup-type-by-extension file=./config/mime.types
```

Filtering Functions

The following functions operate at the Metadata and Data stages to allow or deny resources based on specific criteria specified by the function and its parameters.

These functions can be used in both Enumeration and Generation filters in the file `filter.conf`.

Each “filter-by” function performs a comparison, then either allows or denies the resource. Allowing the resource means that processing continues to the next filtering step. Denying the resource means that processing should stop, because the resource does not meet the criteria for further enumeration or generation. The following functions are described:

- filter-by-exact
- filter-by-max
- filter-by-md5
- filter-by-prefix
- filter-by-regex
- filterrules-process

filter-by-exact

The `filter-by-exact` function allows or denies the resource if the allow / deny string matches the source of information exactly. The keyword `all` matches any string.

Parameters

Table 15-6 lists the parameters used with the `filter-by-exact` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-6 `filter-by-exact` Parameter

Parameter	Description
`src`	Source of information.
`allow/deny`	Contains a string.

Example

The following example filters out all resources whose content-type is `text/plain`. It allows all other resources to proceed:

```
Data fn=filter-by-exact src=type deny=text/plain
```

filter-by-max

The `filter-by-max` function allows the resource if the specified information source is less than or equal to the given value. It denies the resource if the information source is greater than the specified value.

This function can be called no more than once per filter.

Parameters

Table 15-7 lists the parameters used with the `filter-by-max` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-7 `filter-by-max` Parameters

Parameter	Description
`src`	Source of information. It must be one of the following: hosts, objects, or depth.
`value`	Specifies a value for comparison.

Example

This example allows resources whose content-length is less than 1024 K:

```
MetaData fn-filter-by-max src=content-length value=1024
```

filter-by-md5

The `filter-by-md5` function only allows the first resource with a given MD5 checksum value. If the current resource's MD5 has been seen in an earlier resource by this robot, the current resource is denied. As a result, duplication of identical resources or single resources with multiple URLs is prevented.

You can only call this function at the Data stage or later. It can be called no more than once per filter. The filter must invoke the `generate-md5` function to generate an MD5 checksum before invoking `filter-by-md5`.

Parameters

none

Example

The following example shows the typical method of handling MD5 checksums by first generating the checksum and then filtering based on it:

```
Data fn=generate-md5

Data fn=filter-by-md5
```

filter-by-prefix

The `filter-by-prefix` function allows or denies the resource if the given information source begins with the specified prefix string. The resource doesn't have to match completely. The keyword `all` matches any string.

Parameters

Table 15-8 lists the parameters used with the `filter-by-prefix` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-8 `filter-by-prefix` Parameters

Parameter	Description
`src`	Source of information.
`allow/deny`	Contains a string for prefix comparison.

Example

The following example allows resources whose content-type is any kind of text, including `text/html` and `text/plain`:

```
MetaData fn=filter-by-prefix src=type allow=text
```

filter-by-regex

The `filter-by-regex` function supports regular expression pattern matching. It allows resources that match the given regular expression. The supported regular expression syntax is defined by the `POSIX.1` specification. The regular expression `\\*` matches anything.

Parameters

Table 15-9 lists the parameters used with the `filter-by-regex` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-9 `filter-by-regex` Parameters

Parameter	Description
`src`	Source of information.
`allow/deny`	Contains a regular expression string.

Example

The following example denies all resources from sites in the government domain:

```
MetaData fn=filter-by-regex src=host deny=\\*.gov
```

filterrules-process

The `filterrules-process` function handles in the rules in the `filterrules.conf` file.

Parameters

none

Example

```
MetaData fn=filterrules-process
```

Filtering Support Functions

The following functions are used during filtering to manipulate or generate information on the resource. The robot can then process the resource by calling filtering functions. These functions can be used in Enumeration and Generation filters in the file `filter.conf`. The following functions are described:

- assign-source
- assign-type-by-extension
- clear-source
- convert-to-html
- copy-attribute

- generate-by-exact
- generate-by-prefix
- generate-by-regex
- generate-md5
- generate-rd-expires
- generate-rd-last-modified
- rename-attribute

assign-source

The assign-source function assigns a new value to a given information source. This permits editing during the filtering process. The function can assign an explicit new value, or it can copy a value from another information source.

Parameters

Table 15-10 lists the parameters used with the assign-source function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-10 assign-source Parameters

Parameter	Description
dst	Name of the source whose value is to be changed.
value	Specifies an explicit value.
src	Information source to copy to dst

You must specify either a value parameter or a src parameter, but not both.

Example

```
Data fn=assign-source dst=type src=content-type
```

assign-type-by-extension

The assign-type-by-extension function uses the resource's file name to determine its type and assigns this type to the resource for further processing.

The setup-type-by-extension function must be called during setup before assign-type-by-extension can be used.

Parameters

Table 15-11 lists the parameter used with the assign-type-by-extension function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-11 assign-type-by-extension Parameter

Parameter	Description
src	Source of file name to compare. If you do not specify a source, the default is the resource's path.

Example

```
MetaData fn=assign-type-by-extension
```

clear-source

The clear-source function deletes the specified data source. You typically do not need to perform this function. You can create or replace a source by using the assign-source.

Parameters

Table 15-12 lists the parameter used with the clear-source function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-12 clear-source Parameter

Parameter	Description
src	Name of source to delete.

Example

The following example deletes the path source:

```
MetaData fn=clear-source src=path
```

convert-to-html

The convert-to-html function converts the current resource into an HTML file for further processing, if its type matches a specified MIME type. The conversion filter automatically detects the type of the file it is converting.

Parameters

Table 15-13 lists the parameter used with the `convert-to-html` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-13 `convert-to-html` Parameter

Parameter	Description
type	MIME type from which to convert.

Example

The following sequence of function calls causes the filter to convert all Adobe Acrobat PDF files, Microsoft RTF files, and FrameMaker MIF files to HTML, as well as any files whose type was not specified by the server that delivered it.

```
Data fn=convert-to-html type=application/pdf

Data fn=convert-to-html type=application/rtf

Data fn=convert-to-html type=application/x-mif

Data fn=convert-to-html type=unknown
```

copy-attribute

The `copy-attribute` function copies the value from one field in the resource description into another.

Parameters

Table 15-14 lists the parameters used with the `copy-attribute` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-14 `copy-attribute` Parameters

Parameter	Description
src	Field in the resource description from which to copy.

Table 15-14 `copy-attribute` Parameters

Parameter	Description
`dst`	Item in the resource description into which to copy the source.
`truncate`	Maximum length of the source to copy.
`clean`	Boolean parameter indicating whether to fix truncated text (such as not leaving partial words). This parameter is `false` by default.

Example

```
Generate fn=copy-attribute \
  src=partial-text dst=description truncate=200 clean=true
```

generate-by-exact

The `generate-by-exact` function generates a source with a specified value, but only if an existing source exactly matches another value.

Parameters

Table 15-15 lists the parameters used with the `generate-by-exact` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-15 `generate-by-exact` Parameter

Parameter	Description
`dst`	Name of source to generate.
`value`	Value to assign `dst`.
`src`	Source against which to match.

Example

The following example sets the classification to Siroe if the host is `www.siroe.com`.

```
Generate fn="generate-by-exact" match="www.siroe.com:80" src="host"
value="Siroe" dst="classification"
```

generate-by-prefix

This `generate-by-prefix` function generates a source with a specified value, but only if the prefix of an existing source matches another value.

Parameters

Table 15-16 lists the parameters used with the `generate-by-prefix` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-16 `generate-by-prefix` Parameters

Parameter	Description
`dst`	Name of the source to generate.
`value`	Value to assign to `dst`.
`src`	Source against which to match.
`match`	Value to compare to `src`.

Example

The following example sets the classification to Compass if the protocol prefix is HTTP:

```
Generate fn="generate-by-prefix" match="http" src="protocol"
value="World Wide Web" dst="classification"
```

generate-by-regex

The `generate-by-regex` function generates a source with a specified value, but only if an existing source matches a regular expression.

Parameters

Table 15-17 lists the parameters used with the `generate-by-regex` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-17 `generate-by-regex` Parameters

Parameter	Description
`dst`	Name of the source to generate.
`value`	Value to assign to `dst`.
`src`	Source against which to match.

Table 15-17 `generate-by-regex` Parameters

Parameter	Description
`match`	Regular expression string to compare to `src`.

Example

The following example sets the classification to Siroe if the host name matches the regular expression `*.siroe.com`. For example, resources at both `developer.siroe.com` and `home.siroe.com` will be classified as `Siroe`:

```
Generate fn="generate-by-regex" match="\\*.siroe.com" src="host" value="Siroe"
dst="classification"
```

generate-md5

The `generate-md5` function generates an MD5 checksum and adds it to the resource. You can then use the `filter-by-md5` function to deny resources with duplicate MD5 checksums.

Parameters

none

Example

```
Data fn=generate-md5
```

generate-rd-expires

The `generate-rd-expires` function generates an expiration date and adds it to the specified source. The function uses metadata such as the HTTP header and HTML `<META>` tags to obtain any expiration data from the resource. If none exists, it generates an expiration date three months from the current date.

Parameters

Table 15-18 lists the parameter used with the `generate-rd-expires` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-18 `generate-rd-expires` Parameters

Parameter	Description
`dst`	Name of the source. If you omit it, it defaults to `rd-expires`.

Example

```
Generate fn=generate-rd-expires
```

generate-rd-last-modified

The `generate-rd-last-modified` function adds the current time to the specified source.

Parameters

Table 15-19 lists the parameter used with the `generate-rd-last-modified` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-19 `generate-rd-last-modified` Parameter

Parameter	Description
`dst`	Name of the source. If you omit it, it defaults to `rd-last-modified`.

Example

```
Generate fn=generate-last-modified
```

rename-attribute

The `rename-attribute` function changes the name of a field in the resource description. It is most useful in cases where, for example, `extract-html-meta` copies information from a `<META>` tag into a field, and you want to change the name of the field.

Parameters

Table 15-20 lists the parameter used with the `generate-rd-last-modified` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-20 `generate-rd-last-modified` Parameter

Parameter	Description
`src`	String containing a mapping from one name to another.

Example

The following example renames an attribute from author to author-name:

```
Generate fn=rename-attribute src="author->author-name"
```

Enumeration Functions

The following functions operate at the Enumerate stage. These functions control if and how a robot gathers links from a given resource in order to use as starting points for further resource discovery. The following functions are described in this section:

- enumerate-urls
- enumerate-urls-from-text

enumerate-urls

The `enumerate-urls` function scans the resource and enumerates all URLs found in hypertext links. The results are used to spawn further resource discovery. You can specify a content-type to restrict the kind of URLs enumerated.

Parameters

Table 15-21 lists the parameters used with the `enumerate-urls` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-21 `enumerate-urls` Parameters

Parameter	Description
`max`	The maximum number of URLs to spawn from a given resource. The default, if max is omitted, is 1024.
`type`	Content-type that restricts enumeration to those URLs that have the specified content-type. `type` is an optional parameter. If omitted, it will enumerate all URLs.

Example

The following example enumerates HTML URLs only, up to a maximum of 1024:

```
Enumerate fn=enumerate-urls type=text/html
```

enumerate-urls-from-text

The `enumerate-urls-from-text` function scans text resources, looking for strings matching this regular expression: `URL:.*`. It spawns robots to enumerate the URLs from these strings and generate further resource descriptions.

Parameters

Table 15-22 lists the parameter used with the `enumerate-urls-from-text` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-22 `enumerate-urls-from-text` Parameter

Parameter	Description
`max`	The maximum number of URLs to spawn from a given resource. The default, if max is omitted, is 1024.

Example

```
Enumerate fn=enumerate-urls-from-text
```

Generation Functions

The following functions are used in the Generate stage of filtering. Generation functions can generate information that goes into a resource description. In general, they either extract information from the body of the resource itself or copy information from the resource's metadata. The following functions are described in this section:

- extract-full-text
- extract-html-meta
- extract-html-text
- extract-html-toc
- extract-source
- harvest-summarizer

extract-full-text

The `extract-full-text` function extracts the complete text of the resource and adds it to the resource description.

NOTE The `extract-full-text` function should be used with caution, because it can significantly increase the size of the resource description, thus causing database bloat and overall negative impact on network bandwidth.

Parameters

Table 15-23 lists the parameters used with the `extract-full-text` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-23 `extract-full-text` Parameters

Parameter	Description
`truncate`	The maximum number of characters to extract from the resource.
`dst`	Name of the schema item that will receive the full text.

Example

```
Generate fn=extract-full-text
```

extract-html-meta

The `extract-html-meta` function extracts any `<META>` or `<TITLE>` information from an HTML file and adds it to the resource description. A content-type may be specified to restrict the kind of URLs that are generated.

Parameters

Table 15-24 lists the parameters used with the `extract-html-meta` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-24 `extract-html-meta` Parameters

Parameter	Description
`truncate`	The maximum number of bytes to extract.
`type`	Optional parameter. If omitted, it will generate all URLs.

Example

```
Generate fn=extract-html-meta truncate=255 type=text/html
```

extract-html-text

The `extract-html-text` function extracts the first few characters of text from an HTML file, excluding the HTML tags, and adds the text to the resource description. This permits the first part of a document's text to be included in the RD. A content-type may be specified to restrict the kind of URLs that are generated.

Parameters

Table 15-25 lists the parameters used with the `extract-html-text` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-25 `extract-html-text` Parameters

Parameter	Description
`truncate`	The maximum number of bytes to extract.
`skip-headings`	Set to `true` to ignore any HTML headers that occur in the document.
`type`	Optional parameter. If omitted, it will generate all URLs.

Example

```
Generate fn=extract-html-text truncate=255 type=text/html
skip-headings=true
```

extract-html-toc

The `extract-html-toc` function extracts the table-of-contents from the HTML headers and add it to the resource description.

Parameters

Table 15-26 lists the parameters used with the `extract-html-toc` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-26 `extract-html-toc` Parameters

Parameter	Description
`truncate`	The maximum number of bytes to extract.
`level`	Maximum HTML header level to extract. This parameter controls the depth of the table of contents.

Example

```
Generate fn=extract-html-toc truncate=255 level=3
```

extract-source

The `extract-source` function extracts the specified values from the given sources and adds them to the resource description.

Parameters

Table 15-27 lists the parameter used with the `extract-source` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-27 `extract-source` Parameter

Parameter	Description
`src`	List of source names; you can use the `->` operator to define a new name for the RD attribute, for example, `type->content-type` would take the value of the source named `type` and save it in the RD under the attribute named `content-type`.

Example

```
Generate fn=extract-source src="md5,depth,rd-expires,rd-last-modified"
```

harvest-summarizer

The `harvest-summarizer` function runs a Harvest summarizer on the resource and adds the result to the resource description.

To run Harvest summarizers, you must have `$HARVEST_HOME/lib/gatherer` in your `path` before you run the robot.

Parameters

Table 15-28 lists the parameter used with the `harvest-summarizer` function. The table contains two columns. The first column lists the parameter, and the second column provides a description.

Table 15-28 `harvest-summarizer` Parameter

Parameter	Description
`summarizer`	Name of the summarizer program.

Example

```
Generate fn-harvest-summarizer summarizer=HTML.sum
```

Shutdown Functions

The following function can be used during the shutdown phase by both enumeration and generation functions.

filterrules-shutdown

After the rules are run, the `filterrules-shutdown` function performs clean up and shutdown responsibilities.

Parameters

none

Example

```
Shutdown fn=filterrules-shutdown
```

Chapter 16

Administering the Subscriptions Service

This chapter contains the following sections:

Overview

The Subscriptions service enables users to create a profile of interest covering many sources of information. In this release, the sources of information supported include categories, discussions, and searchable documents. The profile is updated with the latest information every time the user accesses the Subscriptions channel. The Subscriptions channel summarizes the number of hits (relevant information) that matches each profile entry the user defined for categorized document and/or discussions.

The Search service is used to:

- Match and count the number of new documents in a target category from a specified range of days
- Match and count the number of new relevant comments within a discussion from a specified range of days
- Match and count the number of document hits against saved searches

The result is displayed as a link that shows the number of matching information to the profile entry. This link redirects the user to a more detailed view of the match itself.

In case of a category subscription, the link redirects the user to the search channel where the specific documents of interest are summarized in a standard category search result format. The Subscriptions channel acts as the doorway to a more detailed view for the user.

The Profiler function provides e-mail notifications when the content of specified interests has changed. The Profiler obtains subscription details for users from the Access Manager, fetches the results from the Search service, and sends e-mail notifications to the users. The Profiler can be scheduled to run at a specific time at the organization level.

Administering the Subscriptions Service

The administrator can enable or disable subscriptions service. The Subscriptions service can be administered at the:

- Root Level
- Organization level
- Organization User level

Root Level

Administering the Subscriptions service at the Root level sets the system wide default maximum number of subscriptions per type (that is, for categories, discussions, and for saved searches). See "To Define the Subscriptions Service at the Root Level" on page 357 for information on defining Subscriptions service at root level.

Organization level

Administering the Subscriptions service at the Organization level overwrites the system wide default maximum number of subscription per type (that is, for categories, discussions, and for saved searches). See "To Define the Subscriptions Service at the Organization Level" on page 357 for information on defining Subscriptions service at organization level.

Organization User level

Administering the Subscriptions service at the Organization User level edits user's Subscriptions service settings. The administrator can maintain the user's service data, such as:

- Update user subscriptions
- Delete user subscriptions

See "To Manage the Subscriptions Service for the User" on page 358 for information on administering Subscriptions service for a user.

To Define the Subscriptions Service at the Root Level

1. Log into the Sun Java System Access Manager administration console and select the Service Configuration tab.
2. Select the Subscriptions service from the list in the left pane.
3. Modify the default values for:
   - **Maximum number of Categories subscriptions** specifies the maximum number of categories that a user can subscribe to.
   - **Maximum number of Discussion subscriptions** specifies the maximum number of discussions that a user can subscribe to.
   - **Maximum number of Saved searches** specifies the maximum number of searches that can be saved.
4. Select:
   - Save to save your values.
   - Reset to reset the values if you modified them.

To Define the Subscriptions Service at the Organization Level

1. Log in to the Sun Java System Access Manager administration console and select Services from the View pull-down menu for your organization.

2. Select the Subscriptions service from the list in the left pane.
3. Select the time (hour and minutes) and days to start the profiler.
4. Select the time (hour and minutes) and days to stop the profiler.
5. Modify the default values for:
   - **Conflict Resolution Level** can be set to Highest, Higher, High, Medium, Low, Lower, and Lowest.
   - **Maximum number of Categories subscriptions** specifies the maximum number of categories that a user can subscribe.
   - **Maximum number of Discussion subscriptions** specifies the maximum number of discussions that a user can subscribe.
   - **Maximum number of Saved searches** specifies the maximum number of searches that can be saved.
6. Select:
   - Save to save your values.
   - Reset to reset the values if you modified them.
   - Delete.

To Manage the Subscriptions Service for the User

1. Log in to the administration console and select Users from the View pull-down menu for your organization.
2. Select the User.

   The user information is displayed on the right pane.
3. Select Subscriptions from the View pull-down menu.

   A page to edit the user's subscriptions is displayed.
4. Edit the subscriptions definition

   For each type of subscription, add or remove subscriptions. The format of:
   - Category subscription is:

   ```
   label | target category | scope | lapsed time
   ```

where:

`label`	Refers to a logical reference given to the edited subscription and it must be a string. This is a required field.
`target category`	Must be of the string format *ABC:DEF:GHI*
`scope`	Refers to a search query and it must be of a string format that is a valid search string, including search operators.
`lapsed time`	Must be one of the following numbers: •0=forever •7=since last week •30=since last month •180=since last 6 months •365=since last year

- Discussions subscriptions is:

  `label | target discussion RD’s URL | scope | lapsed time | minimum rating`

  where:

`label`	Refers to a logical reference given to the edited subscription and it must be a string. This is a required field.
`target discussion RD’s URL`	Must be of string format matching the Discussion’s URL. This cannot be edited by the user using the subscriptions channel for editing the discussion.
`scope`	Refers to a search query and if must be of a string format that is a valid search string, including search operators.
`lapsed time`	Must be one of the following numbers: •0=forever •7=since last week •30=since last month •180=since last 6 months •365=since last year

`minimum rating`	Refers to a filter base on a minimum rating.

- Saved searches is:

`label | target category | scope | lapsed time`

where:

`label`	Refers to a logical reference given to the edited subscription and it must be a string. This is a required field.
`target category`	Must be of the string format *ABC:DEF:GHI*
`scope`	Refers to a search query and if must be of a string format that is a valid search string, including search operators.
`lapsed time`	Must be one of the following numbers: •0=forever •7=since last week •30=since last month •180=since last 6 months •365=since last year

Using the Subscriptions Channel

The Subscriptions channel shows subscriptions by types which can be category subscriptions, discussion subscriptions, and saved searches. For each type of subscription, the following is displayed:

- The subscription label
- A link to the subscription detail representing the number of hits for that particular subscription

An end user can update all the subscriptions and unsubscribe via the subscriptions channel Edit button. End user alerts for matching subscriptions is grouped in the Subscriptions channel. The alerts are generated upon the subscriptions channel's refresh time. The administrator can set the `refreshTime` property for the channel that make the actual rendering of the content cache for a certain period of time.

When the end user tries to refresh the content of the subscriptions channel more than once within less time than the refreshTime parameter, then the content would be read from the cache instead of being generated from the actual data. The `refreshTime` channel property value can be specified in seconds.

An end user is alerted of a new document when the document:

- Is categorized in a subscribed category and matches the scope and time criteria.
- Is a comment on a subscribed discussion and matches the scope and time criteria.
- Matches saved basic or advanced search criteria and time criteria.

To Subscribe to a Category

1. Log in to the sample Desktop.

   You can subscribe to categories via:

   - Browse categories - this includes a Subscribe link
   - Search results that show categories - this includes a Subscribe link
   - Results of search within a category - this includes a Subscribe to Category link

2. Select the subscribe link next to the category you wish to subscribe.

   The page to specify subscription information is displayed.

3. Specify:

   - Subscription Name - A name for the category
   - Target Category- Name of the category
   - Scope of Search- A query string, similar to the Search text field
   - Since - Amount of time you wish to be subscribed to the specified category. It can forever, since last week, since last month, since last 6 months, since last year

4. Select the Finished button.

   The category is added to your list of subscriptions.

To Subscribe to a Discussion

1. Log in to the sample Desktop.

   You can subscribe to discussions via the view discussions link - this includes a Subscribe link.

2. Select the subscribe link for the discussion you wish to subscribe.

   The page to specify subscription information is displayed.

3. Specify:

   - subscription name - A name for the category
   - target category - Name of the category
   - scope of search - A query string, similar to the Search text field
   - since - Amount of time you wish to be subscribed to the specified category. It can forever, since last week, since last month, since last 6 months, since last year.
   - rating - Threshold rating above which subscription is valid

4. Select the Finished button.

   You are now subscribed to the discussion.

To Save a Search

1. Log in to the sample Desktop.
2. Access the Search tab and search for a document.

   The search result page is displayed.

3. Select the subscribe link at the top of the result list.

   The page to specify subscription information is displayed.

4. Specify:

   - Label - Save search label
   - Scope of Search- A query string, similar to the Search text field
   - Since - Amount of time you wish to save the specified search result. It can be forever, since last week, since last month, since last 6 months, since last year.

5. Select the Finished button.

   Your search result is now saved.

Discussions

This section contains the following:

Discussions Overview

Discussions are tied to topics and specific documents. It is a powerful way for people to add and talk about existing documents or create their own. It provides an easy way to share information about specific documents or new topics.

The Sun Java System Portal Server software discussions feature includes discussion threads, starting discussions based on documents or new topics, searching discussions, and rating discussions. By default, the Discussions channel is available on the sample portal for anonymous users. However, an anonymous user cannot subscribe to a discussion or edit the Discussion channel.

The DiscussionLite channel and the Discussions channel are based on the DiscussionProvider. Similar to the search channel JSPs, they have a query portion, a display portion, and use Desktop themes.

DiscussionProvider

The DiscussionProvider is JSP provider that uses the Desktop themes. It retrieves data from the backend Search service using search tag libraries and API. The discussions and comments are stored as different Resource Descriptors (RDs) in the discussion database. The DiscussionProvider supports:

- A full view (via the Discussions channel) and an abbreviated view (via the DiscussionLite channel) that:
  - Starts a new discussion from the discussion channel.

- Posts reply to an existing discussion.
- Starts a new discussion based on web documents from the search channel.

- A Discussion List that:
  - Retrieves main posts sorted by last-modified date.
  - Has pagination so users can access older discussions.
- a discussion view that displays each discussion subtree. The main item is displayed in detail and the subtree is displayed below the main item. View discussion includes:
  - Several filters on the page. A document display can be based on filters such as document rating (irrelevant, routine, interesting, important, and must read).
  - Display preference can be set to threaded or flat display.
  - Expansion threshold helps to control displayed items in the subtree. The users can choose to expand only highly rated documents, or expand all or collapse all. Default value is collapse all. Expand all will expand all the filtered comments. It will also show a description of the discussion, provide a menu for rating the discussion, and allow the user to post a reply.
  - Support to search within a discussion.

  The user also has the option to set these preferences through the channel edit page.
- Commenting and rating a discussion. For example, users can:
  - Add a comment on an existing discussion.
  - Rate all discussions and comments. However user rating is not immediately visible. The rating calculation is based on an algorithm such that the rating for any comment goes up gradually. For example, a comment has to be rated important three times before it is marked as important.
- Searching all discussions and within a discussion. These functions are routed to the search provider. Users can also search by rating in Advance Search.

- Subscriptions. Authenticated users can choose to subscribe to a particular discussion by selecting the subscribe link. The request is handled by the SubscriptionProvider. The `displaySubscription` property (see Code Example 16-1) can be disabled if the feature is not required. By default, the value is true.

Display Profile XML Fragment for DiscussionProvider

Code Example 16-1 shows the DiscussionProvider provider XML fragment in the display profile.

Code Example 16-1 DiscussionProvider Provider Display Profile XML Fragment

```
<Provider name="DiscussionProvider" class="com.sun.portal.providers.jsp.JSPProvider">
    <Properties>
        <String name="title" value="*** Discussions Provider ***"/>
        <String name="description" value="*** DESCRIPTION ***"/>
        <String name="refreshTime" value="0" advanced="true"/>
        <String name="helpURL" value="en/desktop/discussions.htm" advanced="true"/>
        <String name="fontFace1" value="Sans-serif"/>
        <String name="productName" value="Sun Java System Portal Server"/>
        <String name="contentPage" value="discussionContent.jsp"/>
        <String name="editPage" value="discussionEdit.jsp"/>
        <String name="processPage" value="discussionDoEdit.jsp"/>
        <Boolean name="isEditable" value="true" advanced="true"/>
        <String name="editType" value="edit_subset" advanced="true"/>
        <Boolean name="showExceptions" value="false"/>
        <Boolean name="showErrors" value="true"/>
        <String name="width" value="thick"/>
        <String name="column" value="2"/>
        <String name="searchServer" value=""/>
        <String name="dbname" value=""/>
        <Integer name="viewHits" value="8"/>
        <String name="defaultDiscussionDisplay" value="Threaded"/>
        <String name="defaultFilter" value="Irrelevant"/>
        <String name="defaultExpansionThreshold" value="Collapse all"/>
        <Boolean name="viewDiscussionWindow" value="false"/>
        <String name="anonymousAuthor" value="anonymous"/>
        <Boolean name="displaySearch" value="true"/>
        <Boolean name="showDescription" value="false"/>
        <String name="ratingText" value="Irrelevant,Routine,Interesting,Important,Must
Read"/>
    </Properties>
</Provider>
```

Administering the DiscussionProvider

The DiscussionProvider administration is distributed between:

- Channel edit page (this is user configurable)
- Desktop Channel and Container Management link on the administration console for the DiscussionProvider's channel
- Search Service

DiscussionLite Channel

The DiscussionLite channel displays the top twenty discussion titles (which can be reconfigured) and the date. The discussions are sorted by creation date (last modified) and the newest discussion is displayed first. The DiscussionLite channel view has links to view each discussion, view all discussions which target the Discussions Channel, and start a discussion. By default, the channel is displayed in a single container and all links are brought up in a JSPDynamicSingleContainer.

Properties can be configured from the administration console. By default, there are no user editable properties for this channel.

Discussions are stored in the discussion database specified in the `dbname` property in the display profile. Search server host (`searchServer` property), database name (`dbname` property), and the number of discussions to be displayed (`viewHits` property) can be configured in the display profile (see Code Example 16-2 on page 366.)

Code Example 16-2 DiscussionLiteProvider Channel Display Profile XML Fragment

```
<Channel name="DiscussionLite" provider="DiscussionProvider">
    <Properties>
        <String name="title" value="Recent Discussions"/>
        <String name="description" value="This is a DiscussionLite provider example"/>
        <String name="contentPage" value="discussionLiteContent.jsp"/>
        <String name="editPage" value=""/>
        <String name="processPage" value=""/>
        <String name="width" value="thin"/>
        <String name="searchServer" value=""/>
        <String name="db" value="discussion"/>
        <Integer name="viewHits" value="20"/>
    </Properties>
</Channel>
```

The following JSPs are used by the DiscussionLite channel:

`discussionLiteContent.jsp` JSP content page.

`query.jsp`	Sets and executes search query.
`display.jsp`	Displays results.
`error.jsp`	Displays exceptions and error messages.

Discussions Channel

The Discussions channel includes a full view that:

- Shows detailed descriptions for the top eight discussions sorted in descending order. This can be reconfigured via the channel edit page.
- Includes pagination so that users can see all the discussions.
- Supports search. The search returns discussion and comment results.

The Discussions channel properties can be configured from the Sun Java System administration console.

Discussions are stored in the discussion database specified in the `dbname` property in the display profile. Search server host (`searchServer` property), database name (`dbname` property), and the number of discussions to be displayed (`viewHits` property) can be configured in the display profile (see Code Example 16-3.)

Code Example 16-3 Discussions Channel Display Profile XML Fragment

```
<Channel name="Discussions" provider="DiscussionProvider">
    <Properties>
        <String name="title" value="Discussions"/>
        <String name="description" value="This is a Discussion provider example"/>
        <String name="searchServer" value=""/>
        <String name="dbname" value="discussions"/>
        <Integer name="viewHits" value="8"/>
    </Properties>
</Channel>
```

The following JSPs are used by Discussions channel:

`discussionContent.jsp`	JSP content page
`discussionEdit.jsp`	The edit page
`discussionDoEdit.jsp`	The process edit page

declare.jsp	
portal.jsp	Extracts the display profile data
fullDiscussion.jsp	handles the full view presentation
fullDiscussionDisplay.jsp	User interface for the all discussions page
searchUI.jsp	Search form displayed on the all discussions page
viewDiscussion.jsp	View discussion
viewDiscussionBar.jsp	Center horizontal bar with all the filters on the view discussion page
viewDiscussionDisplay.jsp	User interface for the view discussion page
viewDiscussionHeader.jsp	Header comment display on the view discussion page
viewDiscussionNavigation.jsp	Navigation bar displayed above and below the header on the view discussion page
feedback.jsp	Provides comment, feedback, and rating functionality
feedbackDisplay.jsp	Displays the feedback
feedbackForm.jsp	Provides the feedback form
feedbackProcess.jsp	Processes the feedback
error.jsp	Displays exceptions and error messages
query.jsp	Formats and executes the search query
pageFooter.jsp	Provides pagination

Managing and Using the Channels

Administering the DiscussionProvider Channel

Administration of the DiscussionProvider channel is distributed between the Desktop display profile and the Search service in the Sun Java System Access Manager administration console. Provider specific information is stored in the display profile. Discussion document and database related administration must be done in the Search service.

Discussions are stored in the discussion database. The discussion database expects a specific schema for discussions and comments. New schema fields have been added for this feature in the`schema.rdm` file.The search CLI `rdmgr` can be used for database management and debugging. For example, to dump all the comments, type:

```
./run-cs-cli rdmgr -y discussion
```

The sample DiscussionProvider channels are configured to use the default search server. Some sample discussions which are imported in the discussion database and channel is ready for use.

The samples are located at *PortalServer-base*`/SUNWps/samples/discussions/` directory. They are:

`discussions.soif`	A sample SOIF file loaded in the discussion database.
`dp-org.xml`	Contains the discussion channel display profile XML fragments.
`dp-providers.xml`	Contains the discussion provider display profile XML fragments.
`dp-anon.xml`	Contains XML fragment for the authlessanonymous user, loaded at sample portal install time.

Access to discussions can be controlled (to read only or totally hidden) by the administrator.

To Create a Channel from DiscussionProvider

1. Log in to the Sun Java System Access Manager administration console and select Services from the View pull-down menu.

   The list of services is displayed in the left frame.

2. Select Desktop and Channel and Container Management.

   Note that the Manage Channels and Container link is available in the right frame.

3. Select the New button under Channels.

   The page to specify the type of channel to add is displayed.

4. Specify a name for the channel in the Channel Name text box and select DiscussionProvider from the Provider pull-down menu.
5. Select OK.

   This action creates a channel based on the specified provider. The Cancel button returns you to the Channel and Container Management page without creating any new channels.
6. Select the Edit link next to the newly created channel in the Channels table.

   The page to edit the default values of the channel is displayed.
7. Edit the properties and select the Save button to save the modified values.

   The following display profile properties are specific to this provider:

`searchServer`	Path to the search server. By default, `portal/search`.
`dbname`	Any valid database.
`viewHits`	Number of discussions to display.
`defaultDiscussionDisplay`	This can be set to flat or threaded to allow the comment subtree to be displayed as flat or threaded.
`defaultFilter`	Filter for searching and displaying discussions and control display of the subtree. It can be based on ratings such as irrelevant, routine, interesting, important, or must read. By default, its value is irrelevant; so all comments rated irrelevant and above are displayed. The Must read filter will highlight the highly rated comments.
`defaultExpansionThreshold`	This can be set to expand all or collapse all. By default, its value is set to collapse all. If set to expand all, it will expand all the filtered comments, show description, rating menu, and allow user to post reply via links.
`anonymousAuthor`	
`viewDiscussionWindow`	
`displaySearch`	
`showDescription`	For the Discussions channel, this is configurable.
`ratingText`	By default, discussions can be rated at irrelevant, routine, interesting, important, or must read.

Using the DiscussionProvider Sample Channels

To Start a New Discussion

1. Log in to the sample Desktop.
2. To start a new discussion from the:
   - Channel, select the Collaborations tab and select the link to Start A New Discussion.
   - Search channel, select the Start A New Discussion link next to the document.
3. Specify:
   - Title - A title for the discussion
   - Message - Content for discussion
   - Rating - Rate the discussion. It can be routine, interesting, important, or must read.
4. Select Submit Feedback button.

Chapter 17

Configuring the Communication Channels

This chapter provides information on the communication channels for Sun™ Java System Portal Server, starting with general descriptive information, moving to an explanation of the state of the communication channels after installation but before configuration, and finally leading into a description of various steps for configuring the communication channels according to a site's needs.

The information provided on configuration makes up the bulk of this chapter and includes administrator and end user configuration. End users have the ability to edit the configuration of each channel directly from the Portal Desktop by clicking the edit button accessible in each channel. This gives end users access to an edit page (or edit pages) that allows editing of specific server configuration information and that allows editing of specific features visible to the end user in the channel, such as the number of address book entries visible in the Address Book channel.

Administrators can limit or extend end users' editing options. Administrators can even preconfigure channels to work without the need for end user server configuration; for more information see "Administrator Proxy Authentication: Eliminating End-User Credential Configuration" on page 397.

Since administrators can design each channel's edit page, they can select which specific features end users will be able to edit; for more information see "Application Preference Editing: Configuring Communication Channel Edit Pages" on page 391.

Furthermore, if a site has more than one instance of a particular application available—for example, two or more instances of a mail application—administrators can allow end users to configure a second Mail channel on their Portal Desktops; For more information, see "Enabling End-Users to Set Up Multiple Instances of a Communication Channel Type" on page 396.

This chapter includes the following sections:

Overview of the Communication Channels

The Sun Java System Portal Server product offers four communication channels that are accessible by end users directly in Portal Desktop. These channels allow end users access to corresponding applications—such as a mail application—which enable end users to organize, schedule, and communicate more effectively and efficiently.

The four communication channels are:

Address Book Channel The Address Book channel displays address book entries for end users to view. To access the address book in order to create and edit address book entries, first click Launch Address Book.

Calendar Channel The Calendar channel displays calendar events and tasks for end users to view. To access the calendar application in order to create new tasks and events, first click Launch Calendar.

Instant Messaging Channel The Instant Messaging Channel displays the presence status of other users with access to Sun™ Java System Instant Messenger. These contacts are from a list end users have created within the Instant Messenger application. Initiate a chat from the channel by clicking a presence status icon, which is one method of invoking Instant Messenger. To get presence updates directly from the channel, reload Portal Desktop. To receive presence updates as they occur, view contacts' presence status from Instant Messenger by invoking the application; therefore, click Instant Messenger.

Mail Channel The Mail channel displays mail messages sent to end users for them to view. To access the mail application in order to read and compose messages, click Launch Mail.

Supported Software for the Communication Channels

The Sun Java System Portal Server software supports the following resource server platforms for the Communication Channels:

- Sun™ Java System Messaging Server 5.2, 6.0, 6 2004Q2
- Sun™ Java System Calendar Server 5.1.1, 6.0, 6 2004Q2
- Sun™ Java System Instant Messaging Server 6.1, 6 2004Q2
- IBM Lotus Notes 5.0.6
- Microsoft Exchange Server 2000

The Installer and the Communication Channels

The Sun Java System Portal Server installer performs several tasks involving the communication channels. General communication channel configuration tasks are also handled by the installer. More detailed configuration is then required by administrators and end users depending up the needs of the site and of the individuals.

Sun Java System Portal Server Installer Tasks

The Sun Java System Portal Server Installer:

- installs the following packages, `SUNWpsso`, `SUNWpsap`, `SUNWpsmp`, `SUNWpscp`, and `SUNWiimps` which are deployed to the default Sun Java System Portal Server instance. Therefore, the installer does not install the communication channels on all of the Sun Java System Portal Server instances. For information on multi-server deployments, see “Multiple Instance Deployments” on page 376.
- creates the channels, Address Book, Calendar, Instant Messaging, and Mail. The installer places channels for Sun Java System servers into the My Front Page Tab panel container for the sample organization. Therefore, the communication channels are installed only when the sample portal is installed. Microsoft Exchange Server and IBM Lotus Notes server are not automatically placed in a container. An administrator would need to add these channels to a container, if desired.

The default configurations for the Calendar and Mail channels work after only basic configuration by end users; therefore, they do not require further configuration by administrators. The Address Book and Instant Messaging channels require further configuration by both administrators and end users.

- creates and configures the single sign-on (SSO) Adapter service which enables single sign-on with the Sun Java System Calendar Server and Sun Java System Messaging Server.

Multiple Instance Deployments

If you have a multi Sun Java System Portal Server deployment, manually deploy the communication channels to each additional instance of Sun Java System Portal Server and restart each instance. To deploy, type:

PortalServer-base`/SUNWps/bin/deploy redeploy -instance` *instancename* `-deploy_admin_password` *deployadminpassword*

Where *instancename* is the name for that particular non-default instance and *deployadminpassword* is the administrator password for the web container (web server or application server). The web container administrator password is only needed when the web container is Sun™ Java System Application Server or BEA WebLogic Server™. It is not problematic if you include the password when using one of the other acceptable web containers: Sun™ Java System Web Server or IBM WebSphere® Application Server; however, in those cases the password will be ignored.

Code Example 17-1 lists the commands for manually deploying communication channels to two non-default Sun Java System Portal Server instances and for restarting those instances, where `myinstance1` and `myinstance2` are non-default Sun Java System Portal Server instance names and `Admin` is the web container administrator password.

Code Example 17-1 Deploying Communication Channels to a Non-Default Instance

```
portalServer-base/SUNWps/bin/deploy redeploy -instance myinstance1
    -deploy_admin_password Admin
portalServer-base/SUNWps/bin/deploy redeploy -instance myinstance2
    -deploy_admin_password Admin
```

Configuration Tasks for the Communication Channels

The following are the high-level tasks involved in setting up the communication channels. Not all tasks are applicable to all sites. You need to determine if a task is applicable to your site according to your site's business requirements.

- Enabling Access to Mail and Calendar Applications
- Configuring the Services for the Default Organization
- Configuring End-User Channel Settings
- Application Preference Editing: Configuring Communication Channel Edit Pages
- Enabling End-Users to Set Up Multiple Instances of a Communication Channel Type
- Administrator Proxy Authentication: Eliminating End-User Credential Configuration
- Configuring a Read-Only Communication Channel for the Authentication-Less Portal Desktop
- Configuring Microsoft Exchange Server or IBM Lotus Notes
- Creating a New User Under the Default Organization
- Configuring the Mail Provider to Work with an HTTPS Enabled Messaging Server

If you already have Sun Java System Messaging Server and Sun Java System Calendar Server installed either on the same server or on different servers, specify the respective URL when you create a channel.

Enabling Access to Mail and Calendar Applications

Messaging Server and Calendar Server both verify the Internet Protocol (IP) address of the host where the browser requests a login session ID. If the IP address differs from the host IP address where the session ID is issued, Messaging Server and Calendar Server reject the session with a session time out message.

You must change the value of the parameter that enables and disables an IP security check to allow the user to access mail through Portal Server. The parameter that specifies whether to restrict session access to the login IP address, is:

`service.http.ipsecurity`

To Disable ipsecurity for Messaging Server

To disable `ipsecurity` for Messaging Server, perform the following steps in the command line on the machine running the mail server.

1. Log in to the Messaging Server.
2. Type the following command:

   MessagingServer-base`/sbin/server5/msg-`*messaging-server-hostname*`/configutil -o service.http.ipsecurity -v no`
3. Change to root using the su command.
4. Stop Messaging Server using this command

   MessagingServer-base`/sbin/server5/msg-`*messaging-server-hostname*`/stop-msg`
5. Start Messaging Server using this command:

   MessagingServer-base`/sbin/server5/`*msg-messaging-server-hostname*`/start-msg`

To Disable ipsecurity for Calendar Server

To disable `ipsecurity` for Calendar Server, perform the following steps in the command line on the machine running the Calendar Server:

1. Log in to the Calendar Server.
2. Assuming calendar server is installed in `/opt/SUNWics5`, type the following:

   ```
   cd  /opt/SUNWics5/cal/config/
   ```
3. Edit the `ics.conf` file and set ipsecurity to no. For example:

   ```
   service.http.ipsecurity = "no"
   ```
4. Assuming calendar server is installed in `/opt/SUNWics5`, restart Calendar Server by typing:

   ```
   /opt/SUNWics5/cal/sbin/stop-cal
   ```

   ```
   /opt/SUNWics5/cal/sbin/start-cal
   ```

Refresh or re-authenticate to the Portal Desktop; the “Launch Calendar” link should work.

Configuring the Services for the Default Organization

After the communication channels have been installed, the Instant Messaging and Address Book channels require more detailed configuration as explained subsequently. However, the Calendar and Mail channels have sample or default settings that can work without further configuration by an administrator. Site-specific issues can exist for any of the communication channels—including the Calendar and Mail channels—that deserve attention and might require configuration by an administrator before the channels will work according to the needs of the site.

The following sections provide important information relating to the configuration of the communication channels.

Communication Channel Configuration Information

Configuring the Instant Messaging Channel

Configuring the Address Book Channel

Communication Channel Configuration Information

Regarding All the Communication Channels

End-User Configuration

Unless you configure the communication channels with proxy authentication—see “Administrator Proxy Authentication: Eliminating End-User Credential Configuration” on page 397 for more information—end users will still need to go to each channel’s edit page by clicking the edit button in the respective communication channel to further configure the channel.

CAUTION—Undetected Error: Missing Launch Link

If a client port number is entered incorrectly for any of the communication channels, end users will not receive an error message. The error manifests itself by not displaying the launch link for the respective channel, which does not aid end users in identifying the root cause of the problem. Both administrators and end users can enter an incorrect client port number, but since end users can only edit the client port number for the Calendar and Mail channels, those are the only channels where they can create this problem.

CAUTION—Undetected Error: Missing Channel

Various situations can cause end users *not* to see a communication channel and *not* to see an error message explaining the problem. The cause might be a misconfigured template or configuration name, which doesn't allow the template or configuration to be found. A communication channel does not display when any of the following conditions is true:

- The SSOAdapter template is not found.
- The SSO Adapter configuration is not found.
- The `display.template` file is not found.

Regarding the Mail Channel

HTTPS Enabled Messaging Server

If the Mail channel is connected to—a more secure—HTTPS enabled messaging server instead of the basic HTTP enabled messaging server, then you will need to make some security-related adjustments for the Mail channel to work as intended. For more information, see "Configuring the Mail Provider to Work with an HTTPS Enabled Messaging Server" on page 420.

Configuring the Instant Messaging Channel

Sun Java System Instant Messaging Server is installed during the installation of Sun Java System Portal Server if the Enable IM in Portal Server option is selected during the Sun Java System Instant Messaging Server installation.

While the Instant Messaging Portal channel is designed to work right out of the box, other configuration might be necessary depending upon your site's needs. Therefore, after following the steps in "To Configure the Instant Messaging Channel," see "Additional Configuration for the Instant Messaging Channel," to determine if any of that section's subsections apply to your installation.

The Instant Messaging channel is based on a Portal Server content provider called `IMProvider`. The `IMProvider` is an extension of the `JSPProvider` in the Portal Server. As an extension of the `JSPProvider`, `IMProvider` uses the JSP files to generate the content page and the edit page for the Instant Messaging channel. The JSP files are also used to generate the pages used to launch the Instant Messenger. The `IMProvider` also defines an instant messaging-specific tag library and this tag library is used by the JSP files. The JSP files and the tag library use the channel properties that are defined by the `IMProvider`.

For more information on Sun Java System Instant Messaging Server, see *Instant Messaging Administrator's Guide*. For information specific to the Sun Java System Portal Server Instant Messaging Channel tag library and the customization of Instant Messaging Channel through the editing of JSP files, see *Sun Java System Portal Server 6 2005Q1 Desktop Customization Guide*. Furthermore, administrators and end users can access information about Sun Java System Instant Messaging Server by visiting the URL used in the codebase property for the Instant Messaging Channel configuration.

To Configure the Instant Messaging Channel

1. From an Internet browser, log into the Sun™ Java System Access Manager admin console at `http://`*`hostname:port`*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`
2. Click the Identity Management tab to display the View drop down list in the navigation pane (the lower left frame).
3. Select Services in the View drop down list to display the list of configurable services.
4. Under the Portal Server Configuration heading, click the arrow next to Portal Desktop to bring up the Portal Desktop page in the data pane (the lower right frame).
5. Click the Manage Channels and Containers.
6. Scroll down to the Channels heading and click Edit Properties next to IMChannel to display the Instant Messaging service panel, which includes Basic Properties.

   The following is a partial list of the properties displayed in the Edit IMChannel page with example values provided for each property.

Property	Example Value
`authMethod`	`idsvr`
`authUsernameAttr`	`uid`
`clientRunMode`	`plugin`
`codebase`	`imapplet.example.com`
`contactGroup`	`My Contacts`
`mux`	`imserver.example.com`
`muxport`	`49909`

Property	Example Value
netletRule	IM
password	(not applicable when idsvr is used for authmethod)
port	49999
server	imserver.example.com
username	(not applicable when idsvr is used for authmethod)

7. In the text field next to each property you want to input, enter the desired value. Table 17-1 describes the properties and the type of information to enter as a value.

Table 17-1 Property and Value Description for Edit IMChannel Page

Property	Value
authMenthod	It is usually preferable to enter idsvr as the value, which indicates that the authentication method to be used is the Sun Java System Access Manager authentication method. Two values are possible, idsvr or ldap. The idsvr value enables Single Sign-On to work. It also removes the username and password fields from the Instant Messaging channel edit page.
authUsernameAttr	Enter the name of the attribute to use for the user name when authenticating using the idsvr authentication method.
clientRunMode	Enter the method for running the Instant Messaging client: plugin or jnlp (which is used for Java Web Start).
codebase	Enter the URL prefix from which the Instant messaging client is downloaded.
contactGroup	Enter the name of the contact group that is displayed in the Instant Messaging channel.
mux	Enter the hostname of the Sun Java System Instant Messaging Multiplexor to be used when the Instant Messaging client is launched by the channel.
muxport	Enter the port number associated with the Sun Java System Instant Messaging Multiplexor. The default port number is 49909.
netletRule	Enter the name of the netlet rule that is used with the Instant Messaging client when in secure mode via the Secure Remote Access (SRA) gateway.

Table 17-1 Property and Value Description for Edit IMChannel Page

Property	Value
password	Enter the password to use when authenticating using the LDAP method. When stored in the display profile, this property is obfuscated using the `AMPasswordUtil` class.
port	Enter the port number associated with the Sun Java System Instant Messaging Server to be used by the channel. The default port number is 49999.
server	Enter the hostname of the Sun Java System Instant Messaging Server to be used by the channel.
username	Enter the username to use when authenticating using the LDAP method.

8. Scroll as needed and click Save.

Additional Configuration for the Instant Messaging Channel

Steps Might be Required to Allow Multiple Organizations

When a Portal Server instance serves multiple organizations but uses a single Instant Messaging server additional steps must be taken.

Access Manager and Portal Server allow administrators to set up users with the same User ID (uid) across an organization. For example, an organization could have two suborganizations that each have an end user named `enduser22`. This creates a conflict when these two end users attempt to access their respective accounts through the Instant Messaging channel.

To avoid this potential conflict, one set of JSP launch pages per organization must be created to contain a pass-in-the-parameter domain set to the value of the organization's attribute `sunPreferredDomain`. The default launch pages are:

```
/etc/opt/SUNWps/desktop/default/IMProvider/jnlpLaunch.jsp
```

```
/etc/opt/SUNWps/desktop/default/IMProvider/pluginLaunch.jsp
```

Inserting Instant Messenger Links in an Organization

By default the Instant Messenger links are added to the Application channel—which provides the links to launch various applications—in the default organization. The Instant Messenger links allow the Instant Messenger to be launched from the Application channel. You need to add Instant Messenger links manually, if:

- you want to add these links for another organization.

- you do not have the sample portal installed.
- you are using the AppProvider for another channel.

The contents for the Instant Messenger links are in the file *PortalServer-base*/SUNWps/samples/InstantMessaging/dp-IMChannel.xml. The dp-IMChannel.xml file also contains the sample IMChannel.

Edit a copy of the file dp-IMChannel.xml to add the Instant Messenger links information to the display profile for another organization and install the file using the dpadmin command as follows:

1. Change to the following directory:

   PortalServer-base/SUNWps/bin/

2. Create a copy of the dp-IMChannel.xml file as follows:.

   ```
   cp dp-IMChannel.xml newfile.xml
   ```

3. To modify the Application channel, type the following dpadmin command:

```
dpadmin modify -u ADMIN_DN -w PASSPHRASE -d ORG_DN -m newfile.xml
```

Where,

ADMIN_DN - Replace with LDAP administrator DN. For example: amadmin

PASSPHRASE - Replace with the administrator's password.

ORG_DN - Replace with the DN of the Organization where the links are to be added. For example: o=example.com, o=isp

The URL for launching the Instant Messenger using Java Plug-in will be a reference to the Instant Messaging channel with a launch argument. For example:

/portal/dt?action=content&provider=IMChannel&launch=plugin&username=sam

The URL for launching the Instant Messenger applet with Java Web Start will be:

/portal/imlaunch?channel=IMChannel&launch=jnlp&username=sam

Enabling Secure Mode for Sun Java System Instant Messenger in Sun Java Server Portal Server

Netlet facilitates secure communication between the Instant Messenger and the server.

NOTE The Instant Messaging channel automatically uses the secured mode when accessed through the Secure Remote Access gateway. The Instant Messaging channel does not use the secured mode when it is not accessed through the gateway.

To enable the secure mode, you need to add the Netlet Rule.

To add the Netlet Rule:

1. From an Internet browser, log into the Access Manager admin console at `http://`*`hostname:port`*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`
2. Click the Identity Management tab to display the View drop down list in the navigation pane.
3. Select Services in the View drop down list to display the list of configurable services.
4. Scroll down to SRA Configuration and select Netlet.
5. Click the arrow icon beside Netlet. The Netlet Rules are displayed in the right panel.
6. Click Add under Netlet Rules.
7. Type `IM` in the Rule Name field.

NOTE The Netlet rule name can be different. You can configure the Instant Messaging channel to use a different Netlet rule.

8. Remove the default value in the URL field and leave the field blank.

9. Select the Download Applet check box and enter the following string:

   $IM_DOWNLOAD_PORT:$IM_HOST:$IM_PORT

   For example:

   49916:company22.example.com:80

   where,

   IM_DOWNLOAD_PORT. The port on which Instant Messaging resources are downloaded using Netlet.

   IM_HOST. The host name of the web container serving Instant Messenger. For example: company22.example.com

   IM_PORT. The port number of the web container serving the Instant Messenger. For example, 80.

10. Select the default value in the Port-Host-Port List and click Remove.

11. Enter the local host port on which Netlet will run in the Client Port field. For example: 49916.

12. Enter the Instant Messaging Multiplexor host name in the Target Host(s) field.

13. Enter the Instant Messaging Multiplexor port in the Target Port(s) field.

NOTE The values for Netlet Port, Instant Messaging Host, and Instant Messaging Port should be the same as the Instant Messaging service attributes mentioned in the Instant Messaging service panel as discussed in the final steps of "To Configure the Instant Messaging Channel" on page 381.

14. Click Add to List.

15. Click Save to save the Netlet Rule.

Disallowing Users from Launching Instant Messenger

You can remove the ability for users to use the Instant Messaging channel by removing the channel from the user's display profile. For example, to remove the sample IMChannel that is automatically installed, do the following:

1. From an Internet browser, log into the Access Manager admin console at `http://`*`hostname:port`*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`

2. Click the Identity Management tab to display the View drop down list in the navigation pane.
3. Select Services in the View drop down list to display the list of configurable services.
4. Click the arrow icon next to the Portal Desktop service.
5. Click the Manage Channels and Containers Link.
6. Select the check box to the left of the IMChannel channel.
7. Scroll as needed and click Delete to delete the channel.

Configuring the Address Book Channel

For the Address Book channel to work, you need to configure the defaults for the Address Book service. Because the AddressBookProvider is not pre-configured, any channel the user creates based on the AddressBookProvider will not appear on the user's Desktop or on the Content link unless the AddressBookProvider has been configured.

NOTE Creating channels based on the other communications channels in the pre- populated, user-defined channels set may result in the created channel displaying the message: "Please specify a valid configuration." Although the other Communication Channels are defined to a sufficient extent to appear on the user's Desktop, they require additional administrative tasks in order to ascertain which backend service to use.

Additionally, the communication channels require the desktop user to specify backend credentials (such as username and password) after the administrative tasks are completed. The desktop user can specify these values in the channel by using the channel's Edit button.

NOTE The userDefinedChannels set might need to be administered on a per install basis because this set includes references to backend services which might not apply to your particular setup. For example, all Lotus Providers in this set refer to interaction with Lotus backend services for the communication channels which do not apply if none in the Portal user base will be using Lotus backend services.

Configuring the Address Book Service Defaults

This section provides information about single sign-on (SSO) Adapter templates. These templates globally affect the display of the communication channels on users' portal Desktops. To alter the display profile of users for the communication channels, you will need to edit or create SSO Adapter templates and configurations.

This chapter only discusses templates for Address Book. Even for Address Book, the discussion here is very specific. For a broader explanation of SSO Adapters, SSO Adapter templates, and SSO Adapter configurations, see Appendix A, "SSO Adapter Templates and Configurations" on page 455.

To Configure the Address Book Service Defaults

1. From an Internet browser, log into the Access Manager admin console at `http://`*hostname:port*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`
2. Click the Service Configuration tab to display the list of configurable services in the navigation pane.
3. Scroll down the navigation pane to the Single Sign-on Adapter Configuration heading and click the arrow next to the item SSO Adapter, which brings up the SSO Adapter page in the data pane.
4. Click New under SSO Adapter Configuration to add an SSO adapter configuration.

   The New Configuration page appears.
5. Type a configuration name and select SUN-ONE-ADDRESS-BOOK from the menu.
6. Click Next.

   The Configuration Properties page appears.
7. Modify the properties as needed.
8. Scroll down the SSO Adapter page and click Save.
9. When done, click Save.

   For more information about the attributes in an SSO Adapter template string, see Appendix A, "SSO Adapter Templates and Configurations" on page 455.

Configuring End-User Channel Settings

1. Log into the Desktop as the new user:

   a. From an Internet browser, go to:

      `http://`*hostname*`.domain:port/portal/dt, for example`
      `http://psserver.company22.example.com:80/portal/dt`

   b. Enter the user ID and password.

   c. Click Login.

2. Click the Edit button of each channel to configure the server settings.

   - To configure the Mail channel settings:

     Server Name. Enter the host name of the mail server. For example, `mailserver.example.com`.

     IMAP Server Port. Enter the mail server port number.

     SMTP Server Name. Enter the name of the Domain Name Server (DNS) of the outgoing mail—Simple Mail Transfer Protocol (SMTP)—server.

     Client Port. Enter the port number configured for HTTP service.

     User Name. Enter the mail server user name.

     User Password. Enter the mail server user password.

     When sending a message place a copy in Sent Folder. Check this box to store copies of your outgoing messages in the Sent folder.

     Finished. Click this button to save the mail configuration.

     Cancel. Click this button to close the window without saving the configuration details.

   - To configure Address Book channel settings:

     The IMAP user ID and Password are the same as the User Name and User Password entered when configuring the mail channel settings. For details, refer to the previous bulleted item,"To configure the Mail channel settings:"

     User Name. Enter your User Name.

     Password. Enter you Password.

Finished. Click this button to save the server information.

Cancel. Click this button to close the window without saving the details.

- To configure the Calendar channel settings:

  Server Name. Enter the calendar server host name. For example, `Calserver.example.com`.

  Server Port. Enter the calendar server port number.

  User Name. Enter the calendar server user name.

  User Password. Enter the calendar server user password.

  Finished. Click this button to save the calendar configuration.

  Cancel. Click this button to close the window without saving the details.

- To configure the Instant Messaging channel settings:

  Contact List. Select the desired contact list whose contacts will be displayed in the Instant Messaging Channel.

  Launch Method. Select the desired launch method:
  `Java Plugin` or `Java Web Start`.

  Server. Enter the Sun Java System Instant Messaging Server name. For example:
  `IMserver.example.com`

  Server Port. Enter the Sun Java System Instant Messaging Server port number. For example:
  `49999`

  Multiplexor. Enter the Multiplexor name, which must be the same machine as the Sun Java System Instant Messaging server. For example:
  `IMserver.example.com`

  Multiplexor Port. Enter the Multiplexor port number. For example:
  `49909`

  User Name. (This field only appears when the authentication method is set to the Sun Java System Access Manager authentication method, `idsvr`) Enter the Sun Java System Instant Messaging user name.

User Password. (This field only appears when the authentication method is set to the Sun Java System Access Manager authentication method, `idsvr`) Enter the Sun Java System Instant Messaging user password.

Finished. Click this button to save the Sun Java System Instant Messaging Server configuration.

Cancel. Click this button to close the window without saving the details.

NOTE The Address Book, Calendar, and Mail channels each have display options that can be set by the user and by default cannot be overwritten by an administrator. After logging into the Portal Desktop, the user can change the display options for a channel by clicking the edit button in the panel for that channel. The display options are clearly marked and easily changed.

In Address Book, a display option that users can change is the Number of Entries option; in Calendar, a display option that users can change is the Display Day View option; in Mail, a display option that users can change is the Number of Headers option.

Changes made by users to the default communication channels display options take precedence. Any future changes made by administrators will not automatically take effect and a new channel added by administrators will not automatically be accessible by users.

Application Preference Editing: Configuring Communication Channel Edit Pages

You can configure the edit pages that end users will see after they click the edit button in a communication channel's toolbar for the Address Book, Calendar, and Mail channels. The Instant Messaging channel does not use application preference editing. For information about configuring the Instant Messaging Channel's edit page, see *Sun Java System Portal Server 6 2005Q1 Desktop Customization Guide.*

For the three communication channels that allow application preference editing, you can change which options are available for end users to edit, the names and wording that accompany those options, and the way the options are formatted. Configuration of the communication channels edit pages can be performed in the display profile, various HTML templates, and an SSO Adapter template. You might also need to access an SSO Adapter configuration. These items together are involved in the configuration of the edit pages.

This section gives only a brief explanation of application preference editing. Other chapters in this guide and the *Sun Java System Portal Server 6 2005Q1 Desktop Customization Guide* provide a more complete explanation of the template files and the display profile, including how they interact with each other and how you can access and edit them.

Display Profile Attributes for the Edit Pages

The communication channels have two collections in their display profile that drive the creation of the edit pages, `ssoEditAttributes` and `dpEditAttributes`.

You can edit these collections by accessing the Sun Java System Access Manager admin console. Either download the display profile—to edit the XML code before uploading it back to the directory server—or edit specific properties in these collections using only the admin console.

The `ssoEditAttributes` collection controls the editing of the attributes contained by the SSO Adapter service—such as `user name` and `user password`—while `dpEditAttributes` controls the editing for the display profile attributes—such as `sort order` and `sort by`, which are options that by default are editable by end users.

Therefore, these collections list the attributes that can be edited and also contain information on the type of input and the header for the input string to use. For example:

```
<String name="uid" value="string|User Name:"/>
<String name="password" value="password|User Password:"/>
```

The name in the collection must match the name of the corresponding display profile SSO Adapter attribute. The value portion of the item contains two pieces of information separated by the “|” character. The first part of the value string specifies what the display type is for the attribute. The second part of the attribute’s value string specifies the text that will be displayed next to the item. The list below specifies how the type relates to a corresponding HTML GUI item:

- string - Creates a text field where alphanumeric characters can be entered
- password - Creates a password field where the input is replaced with “*”
- check - Creates a checkbox
- select - Creates a select box. Every select item must have a corresponding collection with a list of values and display text

For every select display type you must have a corresponding collection that lists the value to be returned and the display value for the option. The collection name must be made up of the name value for the attribute and the text `SelectOptions`. For example, for the `sortOrder` attribute in the `MailProvider`, the collection name is `sortOrderSelectOptions`:

```
<Collection name="sortOrderSelectOptions" advanced="false" merge="replace"
lock="false" propagate="true">
    <String name="top" value="Most recent at top"/>
    <String name="bottom" value="Most recent at bottom"/>
</Collection>
```

HTML Templates for the Edit Pages

There are nine HTML templates used to create the edit pages for the communication channel providers. The templates were created to be very generic in order to correspond to specific browser GUI types. They mostly relate to specific HTML inputs in the edit pages. The `edit-start.template` and the `edit-end.template` are exceptions in that they contain most of the HTML that is used for page layout. Table 17-2 on page 393 contains a description of each template name and how it relates to the GUI types. Some of the templates are used to start, end and separate the attributes. These templates are available for each of the communication channels at:

`/etc/opt/SUNWps/desktop/default/`*`ChannelName_Provider`*`/html`

For example, the templates for the Calendar channel edit pages can be accessed at:

`/etc/opt/SUNWps/desktop/default/CalendarProvider/html`

Table 17-2 Templates for the Communication Channel Edit Pages

Template	Description
edit-start.template	Provides the starting HTML table for the edit page.
edit-checkbox.template	Provides a generic template for checkbox items.

Table 17-2 Templates for the Communication Channel Edit Pages *(Continued)*

Template	Description
edit-separate.template	Separates the display profile attributes from the SSO attributes.
edit-end.template	Ends the HTML table for the edit page.
edit-password.template	Provides a generic template for password items.
edit-string.template	Provides a generic template for text items.
edit-select.template	Provides a generic template for a select item.
edit-selectoption.template	Provides a generic template for a select option. This way the option can also be generated dynamically from the display profile.
edit-link.template	Provides a template to generate the link so the user can edit their client's display attributes.

A Display Profile Example

This example demonstrates how certain SSO Adapter attributes work together with their corresponding display profile attributes to give end users the ability to change the entries for specific features in a communication channel's edit page, thereby changing how the communication channels are configured and displayed on their Portal Desktops.

The SSO Adapter template in Code Example 17-2 on page 394 is for a sample mail channel. The SSO Adapter template contains two merged attributes:

- uid - User ID
- password - User password

A merged attribute is an attribute that end users can specify. Administrators decide which attributes are merged, therefore, which attributes they want end users to be able to edit.

Code Example 17-2 Sample SSO Adapter Template

```
default|imap:///&configName=MAIL-SERVER-TEMPLATE
    &encoded=password
    &default=protocol
    &default=clientProtocol
    &default=type
    &default=subType
    &default=ssoClassName
    &default=smtpServer
    &default=clientPort
    &default=host
```

Code Example 17-2 Sample SSO Adapter Template *(Continued)*

```
&default=port
&merge=username
&merge=userpassword
&clientProtocol=http
&type=MAIL-TYPE
&subType=sun-one
&ssoClassName=com.sun.ssoadapter.impl.JavaMailSSOAdapter
&smtpServer=example.sun.com
&clientPort=80
&host=company22.example.com
&port=143
```

Code Example 17-3 on page 395 contains the channel's display profile XML fragment for the channel's `ssoEditAttributes`.

After administrators have set an attribute to `merge` in an SSO Adapter template, they can then edit that attribute in the display profile in order to reconfigure how the attribute is displayed to end users in an edit page and how end users can edit it. Administrators can decide how end users are queried for the necessary information by editing the proper display profile collection. For example, in this example, administrators could replace `User Name` with the question, `What is your user name`? The use of the `string` attribute display type before the "`|`" symbol is the most likely choice. However, it's possible for an administrator to change this to the `password` type or to another type.

Code Example 17-3 Sample Mail Channel Display Profile XML Fragment

```
<Channel name="SampleMailChannel" provider="MailProvider">
<Properties>
<Collection name="ssoEditAttributes">
    <String name="username" value="string|User Name:"/>
    <String name="userpassword" value="password|User Password:"/>
</Collection>
```

For this example, in the Mail channel edit page, end users will see text fields titled:

- User Name:
- User Password:

Enabling End-Users to Set Up Multiple Instances of a Communication Channel Type

Multiple types of communication channels can be created by end users or administrators. For end users to create multiple types of communication channels, they will need to utilize the Create a new channel link found on the Content page.

Administrators can create multiple channels for an organization, role, or group. After administrators have made multiple instances of a particular component available—for example, a second instance of the address book component—they can allow end users to configure a second Address Book channel on their Portal Desktops.

Administrators can create an SSO Adapter template for each new communication channel type or they can use one SSO Adapter template and create multiple SSO Adapter configurations for each channel. For more information, see the SSO Adapter documentation in Appendix A, “SSO Adapter Templates and Configurations” on page 455.

Depending on the amount of configuration done by the administrator, the end users may not need to enter as many configuration settings. Administrators can configure these settings by utilizing the application preference editing feature (see “Application Preference Editing: Configuring Communication Channel Edit Pages” on page 391).

To create two Address Book channels, you make each refer to a different SSO adapter template. You can then add both Address Book channels to the visible page you just came from. Likewise, you can create one SSO Adapter template and two SSO Adapter configurations (dynamic). The SSO Adapter template would define the server settings as user definable values (`merge`) and the SSO Adapter configuration would then specify those server settings.

To configure the address book for different servers where end users can configure the servers as needed:

1. Specify the server information as user definable, merge, in the SSO Adapter template. For more information, see Appendix A, “SSO Adapter Templates and Configurations” on page 455.
2. Specify which attributes are editable in the channel’s display profile `ssoEditAttributes` collection. For more information, see “Application Preference Editing: Configuring Communication Channel Edit Pages” on page 391 and for specific information about the display profile, see the *Sun Java System Portal Server 6 2005Q1 Desktop Customization Guide*.

Administrator Proxy Authentication: Eliminating End-User Credential Configuration

You can enable administrator proxy authentication for the Address Book, Calendar, and Mail channels. Extending support for proxy authentication between the Sun Java System Portal Server and Sun Java System Messaging Services (Messaging Server and Calendar Server) eliminates the need for end users to visit a channel's edit page in order to enter their credentials: user name and user password. An administrator's credentials are used instead of end-users' credentials and they are stored in the SSO Adapter template. Within the template, the administrator's User ID is stored as a value for the `proxyAdminUid` attribute while the administrator's password is stored as a value for the `proxyAdminPassword` attribute. Every time a user launches a channel, these values are used to make a connection between a channel and its respective back-end server. A naming attribute for the user is also sent to the back-end server. For more information on the use of naming attributes for administrator proxy authentication, see the `userAttribute` property in Table 17-3 on page 399.

Proxy authentication cannot be configured for Sun Java System Instant Messaging Server, Microsoft Exchange Server, or IBM Lotus Notes server.

CAUTION—Potential for Multiple End Users to be Directed to One Mail Account
Access Manager and Portal Server allow administrators to set up users with the same User ID across an organization. For example, the organization could have two suborganizations that each have an end user named `enduser22`. If administrator proxy authentication is enabled for a Sun Java System communication channel and the end user naming attribute is set to the default, `uid`, then both users could potentially access the same back-end user account. Administrator proxy authentication enables administrators to change the user naming attribute in the SSO Adapter template. For example, you can change the attribute to an attribute that is unique for each employee, such as employee number, to ensure that portal end users access the correct back-end server account.

Overview of How to Configure Proxy Authentication

In order to enable administrator proxy authentication for the Address Book, Calendar, and Mail channels, you need to access the SSO Adapter templates through the Sun Java System Access Manager admin console and you need to access the Sun Java System communication servers. More specifically, you need to:

- Edit SSO Adapter Templates.

In the SSO Adapter Templates, you need to edit the strings that apply to the Address Book, Calendar, and Mail channels. One of the distinguishing factors of the strings is the protocol used:

- The Address Book channel uses the LDAP protocol
- The Calendar channel uses the HTTP protocol
- The Mail channel uses the IMAP or POP protocol.

- access Sun Java System Messaging Server to enable proxy authentication for the Address Book and Mail channels
- access Sun Java System Calendar Server to enable proxy authentication for the Calendar channel.

Proxy Authentication and Single Sign-On (SSO) Adapter Templates

To Edit SSO Adapter Templates For Enabling Administrator Proxy Authentication

1. From an Internet browser, log into the Sun Java System Access Manager admin console at `http://`*`hostname:port`*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`
2. Click the Service Configuration tab to display the list of configurable services in the navigation pane.
3. Select SSO Adapter to display the page for configuring the SSO Adapter in the data pane.
4. Click the string for the channel that you want to enable with administrator proxy authentication.
5. Click in the configuration description field.
6. Delete and key in the necessary information for administrator proxy authentication:

   Table 17-3 describes the properties that need to be edited in the SSO Adapter Template to enable support for administrator proxy authentication.

Table 17-3 SSO Adapter Template Properties for Administrator Proxy Authentication

Property	Value	Description
enableProxyAuth	true | false	The value associated with this attribute is a flag to indicate if proxy authentication is enabled or not. If true the SSO Adapter and Application Adapter will perform proxy authentication. For example, &enableProxyAuth=true
proxyAdminUid	(configurable)	The value associated with this attribute is the administrator's user name. For example, &proxyAdminUid=ServiceAdmin
proxyAdminPassword	(configurable)	The value associated with this attribute is the administrator's user password. For example, &proxyAdminPassword=mailpwd
userAttribute	(configurable)	The value associated with this attribute is the user's naming attribute. This value is mapped to an attribute on the user's record (the user's entry in the directory). A typical record has several attributes, including the User ID (uid) and employee number. By default, the naming attribute is set to uid. For example, &userAttribute=uid By editing the SSO Adapter template, you can map the naming attribute to another attribute, such as employee number.

The preceding four properties appear in the SSO Adapter template string again, as shown subsequently. The configuration of the properties can be set to either default or merge. In the following examples, they are all set to default.

Property	Value	Example
enableProxyAuth	default	&default=enableProxyAuth
proxyAdminUid	default	&default=proxyAdminUid
proxyAdminPassword	default	&default=proxyAdminPassword
userAttribute	default	&default=userAttribute

Proxy Authentication and Communication Servers

Setting Up Sun Java System Messaging Server for Administrator Proxy Authentication

1. Log in to the Sun Java System Messaging Server software host and become super user.

2. Type the following code:

   MessagingServer-base/*msg-instance-name*/**configutil -o service.http.allowadminproxy -v yes**

3. Restart the Messaging Server.

   See the *Sun Java System Messaging Server Administrator's Guide* for detailed instructions on running `configutil` and restarting the server.

Setting Up Calendar Server for Administrator Proxy Authentication

1. Log in to the Sun Java System Calendar Server software host and become super user.
2. Open the following file with the editor of your choice:

   CalendarServer-base/`cal/bin/config/ics.conf`

3. Set the following attribute as shown:

   `service.http.allowadminproxy = "yes"`

4. Restart the calendar server.

   See the *Calendar Server Administrator's Guide* for detailed instructions on restarting the server.

Configuring a Read-Only Communication Channel for the Authentication-Less Portal Desktop

The authentication-less (authless anonymous) Portal Desktop supports read-only communication channels.

Read-Only Communication Channels Facts and Considerations

You can configure read-only access to Address Book, Calendar, and Mail channels for the authless anonymous Portal Desktop. End users can access the information in a read-only communication channel by simply accessing the Portal Desktop; therefore, by entering the following URL in an Internet browser:

`http://`*hostname*`.domain:port/portal/dt, for example`
`http://psserver.company22.example.com:80/portal/dt`

Without logging in, end users have access to any read-only communication channels that administrators have configured. However, end users are usually prevented from editing these channels. For more information about the authentication-less Portal Desktop, including enabling anonymous log in, see the *Sun Java System Portal Server 6 2005Q1 Desktop Customization Guide*.

The calendar channel is the channel most commonly shared by multiple users; therefore, the following steps are for configuring a read-only calendar channel. In this example, the calendar being shared belongs to user *library*. The public read-only calendar is titled *Library Schedule*. The following calendar set up demonstrates one possible approach. For more information about setting up users for the Sun Java System Calendar Server, see the `create` *userid* option of the `csuser` command in the *Sun Java System Calendar Server Administrator's Guide*

To Set Up a Calendar User

1. Create a calendar user by issuing a command such as the following:

   ```
   csuser -g Library -s Admin -y libadmin -l en -m libadmin@library.com
   -c librarySchedule create libadmin
   ```

   Where user **`libadmin`** has a given name of **`Library`**, surname of **`Admin`**, password of **`libadmin`**, preferred language of **`en`** (English), email address of **`libadmin@library.com`**, and calendar ID of **`librarySchedule`**.

2. Set the access permissions to world readable for:

   `libadmin:librarySchedule`

   You can set the access permissions using the `cscal` utility or the end user can do this using Calendar Express.

To Configure a Read-Only Communication Channel

1. Configure the settings for the end user—which in this case is authless anonymous—and create a calendar SSO adapter configuration.

   a. From an Internet browser, log on to the Sun Java System Access Manager admin console at `http://`*hostname:port*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`

   b. Click the Identity Management tab to display the View drop down list in the navigation pane.

   c. Click Users in the View drop down list.

d. Scroll down as needed to the authless anonymous user and click the accompanying arrow to bring up the authlessanonymous page in the data pane.

Now you can add the SSO Adapter service to the authless anonymous user.

e. Click Services in the View drop down list within the authlessanonymous page to display the available services.

f. Click Add.

g. Click the checkbox for SSO Adapter

h. Click Save.

2. Create a calendar SSO Adapter configuration for the authless anonymous user.

a. If not already logged in, log into the Sun Java System Access Manager admin console.

b. Click the Identity Management tab to display the View drop down list in the navigation pane.

c. Select Services in the View drop down list to display the list of configurable services.

d. Click the arrow next to SSO Adapter to bring up the SSO Adapter page in the data pane.

e. In the blank configuration description field, type in a group-oriented SSO Adapter configuration string (with a User ID and password). A typical configuration has been provided subsequently for your reference. The attributes available in this string can vary depending upon how you configured the Sun Java System Portal Server SSO Adapter template. By default the SSO Adapter template expects the user to specify the following information:

- host
- port
- client port
- uid

- password

  If the configuration description field is not blank when you get to it, select all the text in the field and delete it before entering a string in the following format:

  default|undef://?*uid:password@host:port***/?**
  configName=*configuration-name*
  &configDesc=*configuration-description*

  For example:

  ```
  default|undef://?libadmin:libadmin@example.com:3080/?
  configName=sunOneCalendar_librarySchedule
  &configDesc=SUN-ONE-CALENDAR
  ```

f. Click Add.

g. Click Save.

3. Create a new calendar channel for the authless anonymous user that is based on the newly created SSO Adapter configuration.

a. If not already logged in, log into the Sun Java System Access Manager admin console.

b. Click the Identity Management tab to display the View drop down list in the navigation pane.

c. Click Users in the View drop down list.

d. Scroll down as needed to the authless anonymous user and click the accompanying arrow to bring up the authlessanonymous page in the data pane.

Now you can create a new calendar channel for the authless anonymous user.

e. Click Portal Desktop in the View drop down list within the authlessanonymous page to display the Edit link.

f. Click the Edit link.

g. Click the Channel and Container Management link.

h. Scroll down to the Channels section and click New.

i. Enter a name in the Channel Name field. For example:

```
LibraryScheduleChannel
```

j. Choose the correct provider from the provider drop down list. For this example the correct provider is Calendar Provider.

k. Click OK, which returns you to the Channel and Container Management page.

Now you can edit the channel properties.

l. Scroll down to the Channels section and click Edit Properties next to your newly created channel. For example:

`LibraryScheduleChannel`

m. Edit fields as appropriate. For example:

- title: Library Schedule
- description: Library Schedule
- ssoAdapter: sunOneCalendar_librarySchedule
- loadSubscribedCalendars: false (no checkmark)
- is editable: false (no checkmark)

n. Scroll as needed and click Save.

Now you can add the new calendar channel to Portal Desktop of the Authless Anonymous user.

o. Near the top of the page, click Top, which returns you to the Channel and Container Management page.

p. Scroll down the Container Channels section and click the link for the container that you want to add the new channel to. For example, `MyFrontPageTabPanelContainer`. Do not click the accompanying Edit Properties link.

q. Under the Channel Management heading, click the name of the channel you just created. For example, `LibraryScheduleChannel`, which is in the Ready For Use list.

r. Add the channel to the Available to End Users on the Content Page list or to the Visible on the Portal Desktop list.

Click the Add button above the list for which you want to add the channel.

s. Scroll back up the page to click Save under the Channel Management heading.

t. Restart the web container.

Configuring Microsoft Exchange Server or IBM Lotus Notes

Besides supporting Sun Java System Messaging Server and Sun Java System Calendar Server for the communication channels, Sun Java System Portal Server 6 also supports Microsoft Exchange Server and IBM Lotus Notes server.

You can configure Microsoft Exchange Server to work with Sun Java System Portal Server, giving end users access to the Microsoft Outlook Web Access solution. End users gain this access after clicking Launch Address Book, Launch Calendar, or Launch Mail in the respective channel on Portal Desktop.

Similarly, you can configure IBM Lotus Notes server to work with Sun Java System Portal Server, giving end users access to the IBM Lotus Domino Webmail solution through the Address Book, Calendar, and Mail channels.

NOTE Microsoft Exchange Server and IBM Lotus Notes server do not support administrator proxy authentication or single sign on. Because of the single sign on limitation, when end users launch a channel connected to one of these servers, they will need to reenter their credentials before being connected.

To Configure Microsoft Exchange 5.5 Server for Address Book, Calendar, and Mail

1. Log into your Primary Domain Controller (PDC) as an administrator of the domain.
2. Select Start, Programs, Administrative Tools, User Manager for Domains and create an account with user name MAXHost.
3. Select Groups and add MAXHost to the groups, Administrators, and Domain Admins.
4. Ensure that MAXHost can log on locally to the MAIL_HOST, Domain Controllers, and MAX_HOST.
5. Set the password.
6. Log in to your Exchange 5.5 (MAIL_HOST) as MAXHost.
7. Go to Start, Programs, Microsoft Exchange, Microsoft Exchange Administrator.
8. For each end user, set permissions to the mailbox.

9. To enable the permissions tab, go to Tools, Options, Permissions, and enable Show Permissions Page for All Objects.
10. Double-click on the user name.
11. Select the permissions tab and select Add from the permissions page to add MAXHost and leave role as User.

    Repeat steps 9 through 11 for each user who will be accessing the communication channels.
12. Unzip the `ocxhost.zip` file located in the following directory:

    PortalServer-base/SUNWps/export.

    When unzipping the file, you will see the following file format:

```
Archive: ocxhost.zip
creating: ocxhost
creating: ocxhost/international
inflating:ocxhost/international/ocxhostEnglishResourceDll.dll
inflating:ocxhost/ocxhost.exe
```

13. Register ocxhost as follows:
    a. Locate the `ocxhost.exe`.
    b. Select Start and Run.
    c. Type the following in the Run window:

       `ocxhost.exe /multipleuse`
14. To set the properties of ocxhost utility:
    a. Configure the necessary DCOM settings for the ocxhost utility using the `dcomcnfg` utility. That is:
       I. Select Start and Run.
       II. Type **`dcomcnfg`** and select OK.
    b. In the Distributed COM Configuration Properties dialog box:
       I. Select Default Properties tab:
       - Check the Enable Distributed COM on the computer check box.
       - Set the default Authentication Level to Connect.

- Set the default Impersonation Level to Identify.

II. Select the Applications tab.

III. Double-click the ocxhost utility in the Properties dialog.

The ocxhost properties window is displayed.

IV. Check Run Application on this Computer under the Location tab.

V. Set Use custom access permissions, Use custom launch permissions, and Use custom configuration permissions under the Security tab.

VI. Select Edit for the Access, Launch, and Configuration settings and ensure that the following users are included in the Access Control List (ACL):

- Interactive
- Everyone
- System

VII. Select a User under the Identity tab in the ocxhost properties window.

VIII. Select Browse and locate the MAXHost.

IX. Enter the password and confirm the password.

c. Select OK.

The ocxhost DCOM component is now configured and ready to communicate with the Exchange Servers.

To Configure Microsoft Exchange 2000 Server for Address Book, Calendar, and Mail

If the Portal Server should be setup to access Calendar data from an Exchange Server 2000 environment in a complex Windows 2000 Domain configuration `ocxhost.exe` should be installed on a dedicated System (called MAX_HOST).

This is useful for two reasons:

- It allows easier troubleshooting if a user can not access his calendar from the portal.
- It allows a more restrictive security setup if firewall exists between the Portal Server and the Windows Domain.

A “complex” Domain might be if the Exchange Server is a Cluster and / or front-end and a back-end Exchange Server is configured. Or if Windows User and Exchange Mailbox of the same end user are in different Domains.

The following instructions assume that:

MAX_HOST is the name of the dedicated Windows 2000 System running Outlook 2000 and `ocxhost.exe` is installed.

MAIL_HOST is the Exchange Server on which the mailboxes of the end users reside.

PORTAL is the Java Enterprise System Portal Server 2005Q1

DOMAIN is the Windows Domain that has MAX_HOST and MAIL_HOST

When setting up the dedicated Windows 2000 System (MAX_HOST) note the following requirements and assumptions:

- Windows 2000 Server SP3 or Windows 2000 Professional.
- Microsoft Outlook 2000 with CDO enabled.
- The Operating System and Outlook 2000 is installed. Assign an IP Address and bring the new Host in the same Domain as the Exchange Server.

1. Create a User MAXhost in the Domain.
   a. Log into your Host (MAX_HOST) as an administrator of the domain.
   b. Select Start, Programs, Administrative Tools, Active Directory Users and Computers and create an domain account with user name MAXHost.
   c. Select User->Properties->Member of and add the group Administrators (local)
   d. Ensure that MAXHost can log on locally to the MAIL_HOST and MAX_HOST.
   e. Set the password.
2. Configure Outlook for MAXHost user.
   a. Log in to your MAX_HOST System as Domain user MAXHost
   b. Configure the Outlook Profile for the user MAXHost by starting Outlook (refer to Microsoft Documentation if required).
   c. Close Outlook after completing the Outlook setup for MAXHost user.

      Note: Outlook may not run concurrently with `ocxhost.exe`.

3. Configure Microsoft Exchange Server for Address Book, Calendar, and Mail.
   a. Log in to your Exchange 2000 Server (MAIL_HOST) as MAXHost.

      If you are using an Exchange 2000 Front-End Server log in to your front-end Server as MAXHost.

   b. Go to Start, Programs, Microsoft Exchange, Active Directory Users and Computers.

   c. For each end user, set permissions to the mailbox.

   d. Select View->Advanced Features

   e. Double-click on the user name.

   f. Select the Exchange Advanced tab and select Mailbox Rights.

   g. Add MAXHost and give MAXHost full access.

      Repeat steps Step d through Step g for each user who will be accessing the communication channels.

4. Install `ocxhost.exe` on the MAX_HOST.
   a. Log in to MAX_HOST as domain user MAXhost.

   b. Unzip the `ocxhost.zip` file located in the following directory:

      PortalServer-base`/SUNWps/export`.

      When unzipping the file, you will see the following file format:

      - `Archive: ocxhost.zip`
      - `creating: ocxhost`
      - `creating: ocxhost/international`
      - `inflating:ocxhost/international/ocxhostEnglishResourceDll.dll`
      - `inflating:ocxhost/ocxhost.exe`

   c. Register ocxhost as follows:

      I. Locate the `ocxhost.exe` file.

      II. Select Start and Run.

      III. Type ocxhost.exe /multipleuse and select OK.

Note: Perform this registration only once, because each time this command is executed the DCOM settings described in the next step are cleared and need to be reconfigured.

5. Configure the necessary DCOM settings for the `ocxhost` utility using the `dcomcnfg` utility.

   a. Select Start and Run.

   b. Type dcomcnfg and select OK.

   c. In the Distributed COM Configuration Properties dialog box select Default Properties tab and use the following settings:

      - Check the Enable Distributed COM on the computer check box.
      - Set the default Authentication Level to Connect.
      - Set the default Impersonation Level to Identify.

   d. Select the Applications tab.

   e. Double-click the ocxhost utility in the Properties dialog.

      The ocxhost properties window is displayed.

   f. Check Run Application on this Computer under the Location tab.

   g. Set Use custom access permissions, Use custom launch permissions and Use custom configuration permissions under the Security tab.

   h. Select Edit for the Access, Launch, and Configuration settings and ensure that the following users are included in the Access Control List (ACL):

      - Interactive
      - Everyone
      - System

   i. Select a User under the Identity tab in the ocxhost properties window.

   j. Select Browse and locate the MAXHost.

   k. Enter the password and confirm the password.

   l. Select OK.

   The ocxhost DCOM component is now configured and ready to communicate with the Exchange Servers. It will be launched by RPC call when the first access from the Portal Server occurs.

6. Change MAXHost users group.

   For security reasons you may want to remove the domain user from the Administrators group:

   a. Log out and login again as Administrator on MAX_HOST.

   b. Remove the user MAXHost from local Administrators group, (and assign it to Domain User Group).

NOTE A firewall should not be used between the Portal and the MAX_HOST.

(RPC calls using dynamic ports are used for the communication from Portal Server to `ocxhost.exe`)

A firewall should not be used between the MAX_HOST and the MAIL_HOST.

SSO Adapter for Calendar

Set up SSO Adapter for Calendar if you are using a dedicated Server for `ocxhost.exe` (MAX_HOST).

1. Create an SSO Adapter template.

   a. Log in to the Access Manager administration console.

   b. Select the Service Configuration Tab.

   c. Select SSOAdapter

   d. Select New.

   e. Enter a name for your new template and select the existing EXCHANGE-CALENDAR from the list.

   f. Select Next.

   g. In the line for the ocxHost enter the dns-name or IP-Address of the system were `oxchost.exe` resides, in this case MAX_HOST.

   h. Select Save.

2. Create an SSO Adapter configuration for your organization.

   a. From the Identity Management tab, select your organization.

   b. Select Services from the scroll down menu

c. Select SSOAdapter.

d. Under SSO Adapter Configurations, select New.

e. Enter a name for the configuration and select the previously created Template.

f. Select Next.

g. Modify the properties as needed.

You can provide a default Host name which is your MAIL_HOST (DNS name or IP-Address), or you can leave it blank

h. Select Save and note the message Changes Saved.

Instructions on using SSO Adapter Templates and Configurations can also be found at Appendix A, "SSO Adapter Templates and Configurations."

To Uninstall ocxhost.exe

Unregister ocxhost as follows:

1. Locate the ocxhost.exe.
2. Select Start and Run.
3. Type the following in the Run window:

```
ocxhost.exe /unregserver
```

4. Delete the files `ocxhost.exe` and `ocxhostEnglishResourceDll.dll`

To Configure Lotus Domino Server for Address Book, Calendar, and Mail

1. Open the Lotus Administrator by selecting Start, Programs, Lotus Applications, and Lotus Administrator.
2. Go to Administration, Configuration, Server, Current Server Documents.
3. In the Security tab, set the following settings:
   a. Under Java/COM Restrictions, set Run restricted Java/Javascript/COM and Run unrestricted Java/Javascript/COM to *.
   b. Under Security Settings, set:
      - Compare Notes Public keys against those stored in Directory to No.
      - Allow anonymous Notes connections to No.

- Check Passwords on Notes IDs to Disabled.

c. Under Server Access, set Only allow server access to users listed in this Directory to No.

d. Under Web Server Access, set Web Server Authentication to More Name Variations with lower security.

4. In the Ports tab:

a. Select the Notes Network Ports tab and ensure that TCPIP is ENABLED.

b. Select Internet Ports tab and the Web tab.

I. Ensure that TCP/IP port status is Enabled.

II. Under Authentication options, ensure that Name and password and Anonymous are Yes.

c. Select the Directory tab and ensure that:

- TCP/IP port status is Enabled.
- Authentication options items Name and Password and Anonymous are Yes.
- SSL port status is Disabled.

d. Select the Mail tab and ensure that:

- TCP/IP port status is Enabled.
- Authentication options Name and Password and Anonymous are set as follows:

	Mail (IMAP)	Mail (POP)	Mail (SMTP Inbound)	SMTP (Outbound)
Name and Password	Yes	Yes	No	N/A
Anonymous	N/A	N/A	Yes	N/A

e. Select the IIOP tab and ensure that:

- TCP/IP port status is Enabled.
- Authentication options items Name and Password and Anonymous are Yes.

- TCP/IP port number is not set to `0`. It should be `63148`.
- SSL port status is Disabled.

5. Select the Internet Protocols tab and the IIOP sub-tabs. Ensure that the Number of threads is at least 10.
6. Save and close.
7. Restart the server by typing the following in the Domino server console:

   `restart server`

   Restarting the server enables the settings to take effect.
8. Enable DIIOP server by typing the following command in the console:

   `load diiop`
9. Check to see if `diiop_ior.txt` has been generated at location:

   `C:\Lotus\Domino\Data\domino\html\diiop_ior.txt`
10. Enable HTTP service by typing the following command in the console:

    `load http`

    - If there is another service using port 80, the HTTP service will not start. Stop the service running on port 80 and retype the following in the console: **`load http`**

      Or

    - Use the existing service. To do this, copy the `diiop_ior.txt` file into the root or home directory of the web server running on port 80. You can include both the HTTP service and the DIIOP service in the `notes.ini` file to ensure that both services start when you start the server.

Configuration for Lotus Notes

To access a Lotus Notes system using the Sun Java System Portal Server Mail and Calendar channels, you need to add another file to the Sun Java System Portal Server. This file is called `NCSO.jar`. It must be obtained from the Lotus Notes product CD or the IBM web site.

It is available with the Domino Designer and Domino Server products from IBM in the `domino\java` subdirectory. It is also available in a Web download from the following Web site:

`http://www-10.lotus.com/ldd/toolkits`

Go to the Lotus Domino Toolkit link and then to the Java/Corba R5.0.8 update link.

NOTE The download file is a .exe file, which performs the extraction of this file and other files.

Place the NCSO.jar file in the global class path of the web container (web server or application server) as described in the subsequent sections about each of the four possible web containers. For three of the four web containers, the NCSO.jar file is placed in /usr/share/lib. The following table summarizes the steps that follow. The table outlines the process of placing the JAR file in the global class path by indicating where the NCSO.jar file can be placed: in the System Classpath or in the Portal WAR. The table also indicates if special instructions are needed. If so, they are included later in this section.

Web Container	System Classpath	Portal WAR	Special Instructions
Sun Java System Web Server	Yes	Yes	N/A
Sun Java System Application Server	Yes	Yes	N/A
BEA WebLogic Server	Yes	No	How to update system classpath
IBM WebSphere Application Server	No	Yes	How to prune JAR file

For the following steps, you need administrative rights to the web container. Also, you should have access to the web container documentation in order to reference detailed information on various web container processes and commands. For more information concerning the Sun Java System web containers, see *Sun Java System Application Server Administrator's Guide* or *Sun Java System Web Server, Enterprise Edition Administrator's Guide*.

Sun Java System Web Server

Sun Java System Application Server

BEA WebLogic Server

IBM WebSphere Application Server

Sun Java System Web Server

1. Place the `NCSO.jar` in the following Sun Java System Portal Server directory:

   `/usr/share/lib`

2. Update the web container class path to include:

   `/usr/share/lib/NCSO.jar`

   a. Launch the Sun Java System Web Server admin console.

   b. Select the Sun Java System Web Server instance.

   c. Click Manage.

   d. Select the Java tab.

   e. Select the JVM Path Settings.

   f. Add `/usr/share/lib/NCSO.jar` to the classpath suffix.

   g. Select ok

   h. Select Apply

3. Restart the Sun Java System Web Server; though often not mandatory, this is a good practice.

Optional Placement of the NCSO.jar File

1. Place the `NCSO.jar` file in the following directory:

   PortalServer-base`/SUNWps/web-src/WEB-INF/lib`

2. Redeploy the web application with the following command:

   PortalServer-base`/SUNWps/bin/deploy redeploy`

3. Restart the web container.

Sun Java System Application Server

1. Place the `NCSO.jar` in the following Sun Java System Portal Server directory:

   `/usr/share/lib`

2. Update the web container class path to include `/usr/share/lib/NCSO.jar` using the Sun Java System Application Server admin console.

   a. Launch the Sun Java System Application Server admin console.

   b. Select the domain.

   c. Select the server instance.

d. Select the JVM Settings tab in the server instance view.

e. Select Path Settings under the JVM Settings tab.

f. Add `/usr/share/lib/NCSO.jar` in the Classpath Suffix list.

g. Select Save.

h. Select Apply Changes under the General tab of the instance.

i. Select Restart.

Optional Placement of the NCSO.jar File

1. Place the `NCSO.jar` file in the following directory:

PortalServer-base`/SUNWps/web-src/WEB-INF/lib`

2. Redeploy the web application with the following command:

***PortalServer-base*`/SUNWps/bin/deploy redeploy`**

Where *PortalServer-base* represents the directory in which the Portal Server was originally installed.

3. Restart the web container.

BEA WebLogic Server

1. Place the `NCSO.jar` in the following Sun Java System Portal Server directory:

`/usr/share/lib`

2. Update the web container class path to include `/usr/share/lib/NCSO.jar` using the command line.

a. Change directories to the web container install directory:

WebContainer-base`/bea/wlserver6.1/config`

Where *WebContainer-base* represents the directory in which the web container was originally installed.

b. Change directories to the directory that contains the domain instance:

`mydomain`

c. Edit the `startWebLogic.sh` file using the editor of your choice.

d. Add `/usr/share/lib/NCSO.jar` to the end of the CLASSPATH.

NOTE The `startWebLogic.sh` file may contain multiple CLASSPATH definitions. Locate the last definition of the variable and add the following string to the very end of the CLASSPATH:

`/usr/share/lib/NCSO.jar`

e. Restart the web container.

IBM WebSphere Application Server

1. Prune the classes under `org/w3c/dom/` and `org/xml/sax/` from the `NCSO.jar` file and rejar.

The classes should include the following:

- `org/w3c/dom/Document.class`
- `org/w3c/dom/Node.class`
- `org/xml/sax/InputSource.class`
- `org/xml/sax/SAXException.class`

There are many ways to perform this task. Two examples are provided for you here. Follow the method that suits you best:

- The following method requires you to manually unjar and rejar the file:

a. Download and place the file in the following directory:

`/tmp/ncsoprune/work`

b. Unjar the file while it is in that directory.

c. Remove the preceding four classes.

d. Rejar the file.

- The following method requires you to run a script that automates the jar and unjar logic.

a. Download and place the file in the following directory:

`/tmp/ncsoprune/work`

b. Run the following script:

```
#!/bin/ksh
JAR=/usr/j2se/bin/jar
JAR_FILE=NCSO.jar
RM=/usr/bin/rm
BASE_DIR=/tmp/ncsoprune
WORK_DIR=${BASE_DIR}/work
# cd to director of jar file
cd $WORK_DIR
# unjar
$JAR xvf $JAR_FILE
# prune classes
$RM $WORK_DIR/org/w3c/dom/Document.class
$RM $WORK_DIR/org/w3c/dom/Node.class
$RM $WORK_DIR/org/xml/sax/InputSource.class
$RM $WORK_DIR/org/xml/sax/SAXException.class
# jar
$JAR cvf $BASE_DIR/$JAR_FILE META-INF com lotus org
```

2. Place the re-jarred `NCSO.jar` file in the following directory:

   PortalServer-base`/SUNWps/web-src/WEB-INF/lib`

3. Redeploy the web application with the following command:

   ***PortalServer-base*/SUNWps/bin/deploy redeploy**

   Where *PortalServer-base* represents the directory in which the Portal Server was originally installed.

4. Restart the web container.

Creating a New User Under the Default Organization

1. From an Internet browser, log on to the Sun Java System Access Manager admin console at `http://`*hostname:port*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`
2. Click the Identity Management tab to display the View drop down list in the navigation pane.
3. Select Users in the View drop down list to display the User page.
4. Click New to display the New User page in the data pane.

5. Select the services to be assigned to the user.

   Select at a minimum Portal Desktop and SSO Adapter.

6. Enter the user information.

7. Click Create.

   The new user's name appears in the Users list in the navigation pane.

Configuring the Mail Provider to Work with an HTTPS Enabled Messaging Server

The Mail channel automatically supports the HTTP protocol; it does not automatically support the more secure HTTPS protocol. However, if your Sun Java System Messaging Server is enabled for HTTPS, you can follow the steps in this section to configure the Mail provider to work properly with the Sun Java System Messaging Server. These steps do not apply to Microsoft Exchange Server and IBM Lotus Notes server.

Web Container Facts and Considerations

In terms of configuring the mail provider for HTTPS for Sun Java System Messaging Server, the steps regarding the web container differ depending upon which web container you are using: Sun Java System Web Server, Sun Java System Application Server, BEA WebLogic Server, or IBM WebSphere Application Server. You need administrative rights to the web container regardless of which one you use. Also, you should have access to the web container documentation in order to reference detailed information on initializing a trust database, adding certificates, and restarting the web container. For more information on these tasks and other security-related issues concerning the Sun Java System web containers, see *Sun Java System Application Server Administrator's Guide to Security* or *Sun Java System Web Server, Enterprise Edition Administrator's Guide*.

To Configure the Mail Provider to Work with an HTTPS Enabled Messaging Server

1. Initialize the trust database for the web container running Sun Java System Portal Server. For more information, refer to the proper documentation as discussed in the preceding paragraph.

2. Install the SSL certificate for the Trusted Certificate Authority (TCA) if it is not already installed.

3. Restart the web container; though often not mandatory, this is a good practice.
4. Add a new SSO Adapter template specifically for HTTPS. The name of the template used in this example is `SUN-ONE-MAIL-SSL`, which is descriptive since the security protocol, SSL, is included in the name

.

NOTE You can configure an SSO Adapter template and related SSO Adapter configurations in many ways. The steps presented to you subsequently explain a typical configuration. These steps describe how to create a new template and a new configuration since this is a safer practice than simply editing existing templates and configurations.

If you feel comfortable with the editing option, then proceed in that manner. However, if you change the name of the SSO Adapter template and SSO Adapter configuration as part of the edits you make, you will also need to change the SSO Adapter name by editing the properties of the Mail channel.

The two items you would need to edit in the SSO Adapter template or SSO Adapter configuration are:

- `clientProtocol`
- `clientPort`

In creating a new SSO Adapter Template for this example, the `clientProtocol` attribute is set as a `default` attribute. Therefore, it appears in an SSO Adapter template not in an SSO Adapter configuration. The `clientProtocol` attribute must be changed from `http` to `https`. The edited template fragment for this attribute appears as follows:

```
clientProtocol=https
```

For this example, the `clientPort` attribute is set as a `merge` attribute. Therefore, it appears in an SSO Adapter configuration (see Step 5 on page 423). If the `clientPort` attribute were set as a `default` attribute, it would appear in an SSO Adapter template. The client port should be changed to a port reserved exclusively for HTTPS. Here port `443` is used since the HTTPS protocol uses this port number as the default. The edited template fragment for this attribute appears as follows:

```
&clientPort=443
```

a. From an Internet browser, log into the Sun Java System Access Manager admin console at `http://`*hostname:port*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`

b. Click the Service Configuration tab to display the list of configurable services in the navigation pane.

c. Click the arrow next to SSO Adapter to bring up the SSO Adapter page in the data pane.

d. Type a template name and select an existing template from the menu.

e. Click Next.

f. The Template Properties page appears.

g. Modify the properties as needed.

Code Example 17-4 is a typical configuration which has been provided for your reference. The template you enter will probably have different information. For example, you will probably enter a different value for the `configName` property type unless you want to use the name `SUN-ONE-MAIL-SSL`. Furthermore, the attributes you set as `default` and `merge` will probably differ from this example, depending upon the needs of your site.

h. When done, click Save.

Code Example 17-4 Mail SSO Adapter Template for an HTTPS Messaging Server

```
default|imap:///?configName=SUN-ONE-MAIL-SSL
    &encoded=password
    &default=protocol
    &default=clientProtocol
    &default=type
    &default=subType
    &default=enableProxyAuth
    &default=proxyAdminUid
    &default=proxyAdminPassword
    &default=ssoClassName
    &merge=host
    &merge=port
    &merge=uid
    &merge=password
    &merge=smtpServer
    &merge=clientPort
    &clientProtocol=https
    &enableProxyAuth=false
    &proxyAdminUid=[PROXY-ADMIN-UID]
    &proxyAdminPassword=[PROXY-ADMIN_PASSWORD]
    &type=MAIL-TYPE
    &subType=sun-one
    &ssoClassName=com.sun.ssoadapter.impl.JavaMailSSOAdapter
    &default=enablePerRequestConnection
    &enablePerRequestConnection=false
```

At this point, there may be more than one string that begins with the IMAP protocol. This is acceptable.

5. Add a new SSO Adapter configuration specifically for HTTPS. The name of the configuration used in this example is `sunOneMailSSl` because it is similar to the name used for the respective SSO Adapter template.

NOTE See the Note from the preceding step, Step 4 on page 421.

   a. From an Internet browser, log on to the Sun Java System Access Manager admin console at `http://`*hostname:port*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`

   b. Click the Identity Management tab to display the View drop down list in the navigation pane.

   c. Click Services in the View drop down list.

d. Scroll down the navigation pane to the Single Sign-on Adapter configuration heading and click the arrow next to SSO Adapter to bring up the SSO Adapter page in the data pane.

e. Click in the blank configuration description field—which is just above the Add and Remove buttons.

f. Click New under SSO Adapter Configuration to add an SSO adapter configuration.

g. The New Configuration page appears.

h. Type a configuration name and select an SSO Adapter template from the menu.

i. Click Next.

j. The Configuration Properties page appears.

k. Modify the properties as needed.

l. When done, click Save.

6. Add a new Mail channel to Portal Desktop.

Step 4 and Step 5 explained how to create a new SSO Adapter template and SSO Adapter configuration; those are the steps for creating a new channel. In this step you will make the channel available to end users.

The criteria for choosing a name for the new channel is simply one that is descriptive; therefore the example name chosen here is `SunOneMailSSLChannel`.

a. From an Internet browser, log on to the Sun Java System Access Manager admin console at `http://`*`hostname:port`*`/amconsole`, for example `http://psserver.company22.example.com:80/amconsole`

b. Click the Identity Management tab to display the View drop down list in the navigation pane.

c. Select Services in the View drop down list to display the list of configurable services.

d. Under the Portal Server Configuration heading, click the arrow next to Portal Desktop to bring up the Portal Desktop page in the data pane

e. Scroll as needed and click the Manage Channels and Containers link.

f. Scroll down to the Channels heading and click New.

g. In the Channel Name field, type your site's name for the new channel. For example, `SunJavaMailSSLChannel`.

h. In the Provider drop down menu, select MailProvider.

i. Click OK, which returns you to the Channel and Container Management Web page where the channel you just created now exists.

j. Scroll down to the Channels heading and click Edit Properties next to the name of the channel you just created, which for this example is `SunOneMailSSLChannel`.

k. Scroll down to the title field, select and delete any words that currently exist, for example `mail`, and type a provider title. A possible name is `SSL Mail Account`.

l. In the description field, select and delete any words that currently exist, for example `mail`, and type a provider description. The same example is used here for description as for the title in the preceding substep: `SSL Mail Account`.

m. Scroll down the page; select and delete any words that currently exist in the SSO Adapter field, for example `sunOneMail`; and type the same SSO Adapter configuration name used in Step 5 on page 423, which for this example is `sunOneMailSSL`.

n. Scroll down and click Save.

o. Scroll back up the page to click the word `top`, which is the first item following the words `Container Path`.

p. Scroll down to the Container Channels heading and click the link for the container that you want to add the new channel to. For example, `MyFrontPageTabPanelContainer`. Do not click the accompanying Edit Properties link.

q. Scroll down to the Channel Management heading, scroll as needed in the Ready For Use frame, and click the name of your newly created channel to select it.

Remember, for this example the channel name is `SunOneMailSSLChannel`.

r. Add the channel to the Available to End Users on the Content Page list or to the Visible on the Portal Desktop list.

Click the Add button above the list for which you want to add the channel.

s. Scroll back up the page and click Save under the Channel Management heading.

You should now be able to log in and use an HTTPS enabled messaging server.

Managing the Portal Server System

This chapter describes the various administrative tasks associated with maintaining the Sun Java™ System Portal Server system.

This chapter contains these sections:

Configuring Secure Sockets Layer (SSL)

You can configure Secure Sockets Layer (SSL) with Portal Server and associated components in the following ways:

- Portal Server—If you configure SSL for just the Portal Server system and not a gateway, then your intranet is “open.”

  You can use SSL between the Portal Server user interface node (where the Sun Java™ System Access Manager administration console, Desktop, servlets, and so on run) and gateway node; and between the Portal Server user interface node and end user computers.

- Sun Java™ System Directory Server—You can configure SSL for the Sun Java System Directory Server and use a secure connection between Sun Java System Access Manager and the Portal Server. See Chapter 6, “Basic Configurations” in the *Sun Java System Access Manager Installation and Configuration Guide* at the following URL for information on enabling SSL on the directory server:

  `http://docs.sun.com/source/816-5626-10/contents.html`

NOTE If you have configured SSL on a directory server, you must disable SSL before uninstalling the directory server with the Portal Server installation script. In addition, to use the `dpadmin` command at the command line, you must also disable SSL.

- Sun Java™ System Portal Server: Secure Remote Access —When you configure SSL for the gateway, your intranet is “secure.” See the *Sun Java System Portal Server: Secure Remote Access 6 2004Q2 Administration Guide* for the steps to configure SSL on the gateway.

To Configure SSL with Portal Server

Use this procedure if you chose to run SSL on your machine during the Portal Server installation.

1. Create a trust database for the web server on which you installed Portal Server.

   See Chapter 5, “Creating a Trust Database” in the *Sun Java System Web Server 6 2004Q2, Enterprise Edition Administration Guide* at the following URL for more information:

   `http://docs.sun.com/source/816-5682-10/index.htm`

2. Request a certificate for the web server on which you installed Portal Server software and install the certificate on the web server instance.

   See Chapter 5, “Requesting and Installing a VeriSign Certificate” or “Requesting and Installing Other Server Certificates” in the *Sun Java System Web Server 6 2004Q2, Enterprise Edition Administration Guide* for more information.

3. Turn on encryption for the Portal Server web server instance.

   In the web server administration console, select the Preferences tab, select Add Listen Socket, then select Edit Listen Socket and turn on security.

   See Chapter 5, "Turning Security On," in the *Sun Java System Web Server 6 2004Q2, Enterprise Edition Administration Guide* for more information,

4. Click Apply and Apply Changes in the web server administration console.

5. Restart the web container.

   See your web container documentation for instructions on starting the web container.

6. The system prompts you for the password to get to the certificate database.

NOTE To avoid having to type the passphrase on each reboot, create a file named `.wtpass` that contains the web server passphrase and place it in the *AccessManager-base*`/SUNWam/config` directory. If you reboot the system with a secure web server without having this file, you must type in the passphrase at the system console.

7. Verify that you can now log on to the Portal Server portal using SSL:

   - To log on to the Sun Java System Access Manager administration console, type:
     `https://`*server*`:`*port*`/amconsole`
   - To log on as a user to the Desktop, type:
     `https://`*server*`:`*port*`/`*deploy_uri*

     for example,
     `https://sesta:80/portal/dt`

To Modify an Existing Portal Server Installation to Use SSL

Use this procedure if you answered **n** when asked "Do you want to run SSL on *hostname*?" during the Portal Server installation. See the *Sun Java System Portal Server 6 2004Q2 Installation Guide* for more information.

1. Log in to the Sun Java System Access Manager admin console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

3. Click the properties arrow next to Platform.

   The Platform attributes appear in the data pane.

4. In the server list, change `http` to `https`.

5. Click Save to save your changes.

6. Install the certificate on the web server.

   See Step 1 through Step 4 in "To Configure SSL with Portal Server" on page 428 for details.

7. Copy the `server.xml` and `magnus.conf` files from `/`*AccessManager-base*`/SUNWam/servers/https-`*hostname-domain*`/conf_bk` directory to the `/`*AccessManager-base*`/SUNWam/servers/https-`*hostname-domain*`/config` directory.

8. Add the following line to the `/`*AccessManager-base*`/SUNWam/lib/AMConfig.properties` file if the root CA is not installed for your certificate.

   ```
   com.sun.am.jssproxy.trustAllServerCerts=true
   ```

   This option tells JSS to trust the certificate.

9. In the `/`*AccessManager-base*`/SUNWam/lib/AMConfig.properties` file, change `http` to `https` for the following:

   ```
   com.sun.am.server.protocol
   com.sun.am.naming.url
   com.sun.am.notification.url
   com.sun.am.session.server.protocol
   com.sun.services.cdsso.CDCURL
   com.sun.services.cdc.authLoginUrl
   ```

10. Restart the web container.

11. The system prompts you for the password to get to the certificate database.

See Chapter 11, “Managing SSL” in the *Sun Java System Directory Server Administration Guide* for more information.

To Configure a Portal Server Instance to Use SSL

1. Log in to the Sun Java System Access Manager admin console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

3. Click the properties arrow next to Platform.

   The Platform attributes appear in the data pane.

4. In the server list, change `http` to `https`.

5. Click Save to save your changes.

6. Install the certificate on the web server.

   See Step 1 through Step 4 in “To Configure SSL with Portal Server” on page 428 for details.

7. If this server is part of a multi-instance installation, copy the `server.xml` and `magnus.conf` files from */AccessManager-base*`/SUNWam/servers/https-`*instance_nickname*`/conf_bk` directory to the */AccessManager-base*`/SUNWam/servers/https-`*instance_nickname*`/config` directory.

8. Add the following line to the */AccessManager-base*`/SUNWam/lib/AMConfig-`*instance_nickname*`.properties` file if the root CA is not installed for your certificate.

   ```
   com.sun.am.jssproxy.trustAllServerCerts=true
   ```

   This option tells JSS to trust the certificate.

9. In the */AccessManager-base*/SUNWam/lib/AMConfig-*instance_nickname*.properties file, change http to https for the following:

   com.sun.am.server.protocol

   com.sun.am.naming.url

   com.sun.am.notification.url

   com.sun.am.session.server.protocol

   com.sun.services.cdsso.CDCURL

   com.sun.services.cdc.authLoginUrl

10. Restart the web container.

11. The system prompts you for the password to get to the certificate database.

See Chapter 11, "Managing SSL" in the *Sun Java System Directory Server Administration Guide* for more information.

Backing Up and Restoring Portal Server Configuration

The Portal Server user and service configuration is stored on the directory server in an LDAP Directory Information Tree (DIT). This allows you to back up and restore configuration information via a Lightweight Directory Interchange Format (LDIF) file.

To Back Up a Portal Server Configuration

To back up Portal Server configuration information use the db2ldif command. This command is available in the slapd-*hostname* directory within the base directory of the directory server. For example, if the directory server was installed to the default install directory (/usr/ldap) on the server sesta, the base directory would be /usr/ldap/slapd-sesta.

1. Change directories to the directory server base directory containing the db2ldif command.

   cd *DirectoryServer-base*/slapd-HOSTNAME

2. Save the configuration to an LDIF file using the `db2ldif` command with the `-s` option specifying the top level of the DIT for Portal Server. For example, to save a configuration in which the top level of the DIT is `isp`, type the following:

   `./db2ldif -s "o=isp"`

   The data are saved to an LDIF file. The command saves the file to a the current directory. The following format is used to name the file:

   `YYYY_MM_DD_HHMMSS.ldif`

   After the file is saved, the following example output displays:

```
[16/May/2002:14:11:25 -0700] - Backend Instance: userRoot
ldiffile: /usr/ldap/slapd-sesta/ldif/2002_05_16_141122.ldif
[16/May/2002:14:11:28 -0700] - export userRoot: Processed 178 entries (100%).
```

To Restore a Portal Server Configuration

You can restore the Portal Server configuration information you have backed up via the `db2ldif` command using the `ldif2db` command. This command is available in the `slapd-`*hostname* directory within the base directory of the directory server. For example, if the directory server was installed to the default install directory (`/usr/ldap`) on the server `sesta`, the base directory would be `/usr/ldap/slapd-sesta`.

1. Change directories to the Directory Server base directory containing the `ldif2db` command by entering:

   `cd` *DirectoryServer-base/slapd-HOSTNAME*

2. Stop the directory server by entering:

   `./stop-slapd`

3. Restore the configuration from the LDIF file to the directory server using the `ldif2db` command with the `-s` option specifying the top level of the DIT for Portal Server and the `-i` option specifying the file name. For example, to restore the LDIF file saved in the previous procedure to the top level of the DIT of `isp`, type the following:

   `./ldif2db -s "o=isp" -i`
   `/usr/ldap/slapd-sesta/ldif/2002_05_16_141122.ldif`

   After the configuration is restored, the following example output displays:

```
importing data ...
```

```
[16/May/2002:16:37:02 -0700] - Backend Instance: userRoot

[16/May/2002:16:37:03 -0700] - import userRoot: Index buffering enabled
with bucket size 13

[16/May/2002:16:37:03 -0700] - import userRoot: Beginning import job...

[16/May/2002:16:37:03 -0700] - import userRoot: Processing file
"/usr/ldap/slapd-sesta/ldif/2002_05_16_141122.ldif"

[16/May/2002:16:37:04 -0700] - import userRoot: Finished scanning file
"/usr/ldap/slapd-sesta/ldif/2002_05_16_141122.ldif" (178 entries)

[16/May/2002:16:37:05 -0700] - import userRoot: Workers finished;
cleaning up...

[16/May/2002:16:37:08 -0700] - import userRoot: Workers cleaned up.

[16/May/2002:16:37:08 -0700] - import userRoot: Cleaning up producer
thread...

[16/May/2002:16:37:08 -0700] - import userRoot: Indexing complete.
Post-processing...

[16/May/2002:16:37:08 -0700] - import userRoot: Flushing caches...

[16/May/2002:16:37:08 -0700] - import userRoot: Closing files...

[16/May/2002:16:37:09 -0700] - import userRoot: Import complete.
Processed 178 entries in 6 seconds. (29.67 entries/sec)
```

4. Restart the directory server by entering:

   ./start-slapd

Changing Portal Server Network Settings

To physically move a server running Portal Server software from one network to another, you need only change the fully qualified domain name mapping the IP address in the /etc/hosts file. There are no other hardcoded addresses that need to be changed.

Managing a Multiple UI Node Installation

When you install Portal Server software onto multiple UI nodes, you need to make a configuration change to the Platform attributes in the Sun Java System Access Manager administration console. You edit the Server List attribute to include the URLs for each UI node.

The Sun Java System Access Manager naming service reads the Server List attribute at initialization time. This list contains the Sun Java System Access Manager session servers in a single Sun Java System Access Manager configuration. For example, if two Sun Java System Access Manager servers are installed and should work as one, they must both be included in this list. If the host specified in a request for a service URL is not in this list, the naming service will reject the request. The first value in the list specifies the host name and port of the server specified during installation. Additional servers can be added using the format *protocol*://*server*:*port*.

To Add Additional Portal Servers to the Server List

1. Log in to the Sun Java System Access Manager admin console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

   The global services appear in the navigation pane.

3. Click the properties arrow next to Platform.

   The Platform attributes appear in the data pane.

4. Edit the Server List attribute.

   For each server functioning as a UI node, type the server URL, for example, `http://host1.sesta.com:80` and then click the Add button. The URL then appears in the Server List.

5. Click Save.

6. Restart the web container.

Configuring a Portal Server Instance to Use an HTTP Proxy

If the Portal Server software is installed on a host that cannot directly access certain portions of the Internet or your intranet, you might want to configure the instance to use an HTTP proxy.

The Portal Server is configured to use an HTTP proxy by setting the http.proxyHost and http.proxyPort Java Virtual Machine (JVM) system properties in the web container that is running the Portal Server web application. The method for setting JVM system properties varies on different web containers. The procedure described in this section is specifically for configuring the Sun Java System Web Server instance to use an HTTP proxy.

1. Change directories to the Web Server base directory containing the configuration for the instance by entering:

   `cd /`*`WebServer-base`*`/SUNWam/servers/https-`*`hostname-domain`***`/config`**

2. Edit the `server.xml` file within this directory and add the following lines:

   <*JVMOPTIONS*>`-Dhttp.proxyHost=proxy_host`</*JVMOPTIONS*>

   <*JVMOPTIONS*`-Dhttp.proxyPort=proxy_port`</*JVMOPTIONS*>

   where *proxy_host* is the fully-qualified domain name of the proxy host and *proxy_port* is the port on which the proxy is run.

NOTE If the `server.xml` file has a proxy set up (using the `http.proxyHost=` and `http.proxyPort=` options) you may want to add the `http.nonProxyHosts=`*proxy_host* option. It is possible that the portal server may not be accessible through the proxy server, unless the portal server is added to the proxy server access list.

Managing Portal Server Logs

You can configure Portal Server logging to log information to a flat file or to a database. When logging to a database, the JDBC protocol is used.

To Configure Logging to a File

1. Log in to the Sun Java System Access Manager admin console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

   The global services appear in the navigation pane.

3. Click the properties arrow next to Logging.

   The Logging attributes appear in the data pane.

4. Select File as the Logging Type attribute.

5. Specify the directory path for the log files in the Log Location attribute.

6. Specify the maximum file size in bytes for the log file in the Max Log Size attribute.

7. Specify the number of backup logs in the Number of History Files attribute.

8. Click Save.

To Configure Logging to a Database

1. Log in to the Sun Java System Access Manager admin console as administrator.

   By default, Identity Management is selected in the location pane and All created organizations are displayed in the navigation pane.

2. Choose Service Configuration in the location pane.

   The global services appear in the navigation pane.

3. Click the properties arrow next to the Logging service in the navigation pane.

   The Logging attributes appear in the data pane.

4. Select DB as the Logging Type attribute.

5. Specify a user name and password with which to connect to the database in the Database User Name and Database User Password attributes.

6. Specify the driver to use for logging in the Database Driver Name attribute.

7. Click Save.

Debugging Portal Server

This section describes how to set the debug level to help you troubleshoot various Portal Server components.

To Set the Debug Level for Sun Java System Access Manager

The debug level allows you to define the types of messages sent to the debug log. The following levels are supported:

- off—No messages are sent to the debug log.
- error—Error messages are sent to the debug log.
- warning—Warning and error messages are sent to the debug log.
- message—Status, warning, and error messages are sent to the debug log.

By default, debug messages are sent to log files in the `/var/opt/SUNWam/debug` directory.

To set the debug level:

1. Define the debug level in the following line of the `/etc/opt/SUNWps/desktopconfig.properties` file:

   `debugLevel=`*value*

2. Restart the web container.
3. Examine the various log files under `/var/opt/SUNWam/debug` as well as the Sun Java System Web Server log file.

Part III

Tuning the Sun Java System Portal Server

Chapter 19

Tuning the Portal Server

This chapter describes the configuration parameters for optimizing the performance and capacity of the Sun Java™ System Portal Server. The `perftune` script (in *PortalServer-base*`/SUNWps/bin` directory), bundled with Portal Server, automates most of the tuning process discussed in this chapter.

Updates to the perftune script for this release include:

- A safeguard that prevents the script from tuning a system that does not have enough memory.
- The ability to tune Portal Server and Access Manager instances that are installed on separate machines.
- New JVM™ tuning parameters.

NOTE If you are tuning either the Portal Server or Access Manager in a separated instance, select only the option to tune the component that is installed on the system. For example, if running the perftune script on a machine that has just the Portal Server instance installed, do not tune the Access Manager.

Introduction

The `perftune` script:

- Tunes the Solaris™ Operating System Kernel and TCP settings (see Solaris Tuning)
- Modifies the following configuration files as part of:
  - Sun Java System Web Server 6.1 Tuning:

- *WebServer-base*/SUNWwbsvr/*webserver-instance*/config/magnus.conf
- *WebServer-base*/SUNWwbsvr/*webserver-instance*/config/server.xml

❍ Sun Java System Application Server 7.0 Tuning:

- *Deploy_Domain*/*Deploy_Instance*/config/init.conf
- *Deploy_Domain*/*Deploy_Instance*/config/server.xml
- *Deploy_Domain*/*Deploy_Instance*/config/server.policy

Deploy_Domain = Application Server's domain directory and the application server domain, for example, /var/opt/SUNWappserver7/domains/domain1

Deploy_Instance = Application Server instance, for example, server1.

❍ Sun Java System Directory Server Tuning:

- /var/opt/mps/serverroot/slapd-*hostname*/config/dse.ldif

❍ Sun Java System Access Manager Tuning:

- /etc/opt/SUNWam/config/serverconfig.xml
- /etc/opt/SUNWam/config/AMConfig*properties

❍ Portal Server Desktop Tuning

- /etc/opt/SUNWps/desktop/desktopconfig.properties

- Modifies properties of the Portal Server Desktop service and Sun Java™ System Access Manager authentication service.

Tuning Instructions

To run the perftune script:

1. Log in to the machine and become super user.

   You need root access to run this script.

2. Change directories to *PortalServer-base*/SUNWps/bin.
3. Enter:

   ./perftune.

The perftune script performs start and stop operation of servers during tuning process. It creates backup copies of modified files in *filename*-orig-*date*-*pid* format. Reboot the system after running the script to take effect tuning changes.

Solaris Tuning

Kernel Tuning

To the /etc/system file, the script appends the following setters:

- File Descriptor Limits - Number of open files limits
  - set rlim_fd_max=16384
  - set rlim_fd_cur=16384
- Stream queue Size - The depth of the syncq (number of messages) before a destination streams queue generates a QFULL
  - set sq_max_size=0
- TCP Connection Hash Size (<= file descriptors)
  - set tcp:tcp_conn_hash_size=8192

The original file (/etc/system) will be copied to a file of the format:

/etc/system-orig-'$DATE+%y%m%d'-$$

NOTE Most of the files get backed up. The script creates backup copies of modified files in respective directories in the following format: *filename*-orig-*date*-*pid*

TCP Parameters Tuning

Changes to TCP parameters (shown within parenthesis) in /dev/tcp include:

- TCP Time Wait Interval (tcp_time_wait_interval) - The amount of time a TCP socket will remain in the TIME_WAIT state (after the connection is closed) is set to 60000
- TCP Fin Wait 2 Interval (tcp_fin_wait_2_flush_interval) - The amount of time a TCP socket will remain in the FIN_WAIT_2 state (after the connection is closed) is set to 67500

- TCP Maximum Connection Size (tcp_conn_req_max_q) - The maximum number of fully established connection is set to 8192
- TCP List Queue (tcp_conn_req_max_q0) - The size of the queue containing unestablished connections is set to 8192
- TCP Packet Drop Time (tcp_ip_abort_interval) - The amount of time before a packet is dropped is set to 60000
- TCP Keep Alive Interval (tcp_keepalive_interval) - This is set to 90000
- TCP Maximum Retransmit Interval (tcp_rexmit_interval_max) - This is set to 6000
- TCP Minimum Retransmit Interval (tcp_rexmit_interval_min) - This is set to 3000
- TCP Initial Retransmit Interval (tcp_rexmit_interval_initial) - This is set to 500
- TCP Smallest Anonymous Port (tcp_smallest_anon_port) - This is set to 1024
- TCP Initial Packets for Slow Start Algorithm (tcp_slow_start_initial) - This is set to 2
- TCP Transmit / Receive Buffer Size Limit (tcp_xmit_hiwat and tcp_recv_hiwat) - These are set to 32768 each

In order to execute the ndd commands automatically when the system is rebooted, the perftune script copies the S99ndds_tcp file into /etc/rc2.d/ directory.

Sun Java System Access Manager Tuning

Directory Server Connection Pool

Changes made to the /etc/opt/SUNWam/config/serverconfig.xml file are as follows:

- Increases the minimum connection pool size to 10
- Increases the maximum connection pool size to 90

NOTE The /etc/opt/SUNWam/config/serverconfig.xml file is backed up in the format:

filename-orig-*date*-*pid*

LDAP Authentication Service

- Updates LDAP connection pools default size (min:max) to 10:90

Sun Java System Access Manager Services Configuration Parameters

Changes are made to the `/etc/opt/SUNWam/config/AMConfig.properties` file as follows:

- Specifies `com.iplanet.am.logstatus` to `INACTIVE`
- Increases `com.iplanet.am.session.maxSession` (default 50000) if expected number of concurrent sessions exceeds this value
- Disables `com.iplanet.am.session.httpSession.enabled`
- Specifies `com.iplanet.am.sdk.cache.maxSize=DSAME_MAX_CACHE_SIZE` where `DSAME_MAX_CACHE_SIZE` is based on Access Manager and Portal Server tuning guide recommendations.
  `DSAME_MAX_CACHE_SIZE=(MAX_CONCURRENT_SESSIONS) * (2 + services registered)` where `MAX_CONCURRENT_SESSIONS`=7000, and services registered=3 (out of the box default).
- Specifies `com.iplanet.am.stats.interval` with the value 60.
- Specifies `com.iplanet.am.session.purgedelay` with the value 5.
- Specifies `com.iplanet.services.stats.state` with the value `file`.
- Specifies `com.iplanet.services.states.directory` with the value `/var/opt/SUNWam/debug`.

Polling mode is enabled if Access Manager and Portal Server are installed on separate machines. Polling mode provides the following options:

- option to specify polling mode
- option to set interval for polling mode

The following threadpool properties in the `/opt/SUNWam/lib/AMConfig.properties` file are exposed in Portal Server 6:

- `com.iplanet.am.notification.threadpool.threshold`. This property indicates the maximum size of the task queue in the thread pool. The thread pool will reject further requests if the number of unprocessed tasks in the queue exceeds that threshold value. This number depends on the system memory resource. Each task requires about 3k. You should decide how many tasks can be queued given the size of thread pool. A task is queued only when no thread in the pool is available.

The default value is set at 200. This might be high for your particular usage, and can be adjusted. For example use a value of 40 for a 4-CPU Ultra Sparc II or III machine.

- `com.iplanet.am.notification.threadpool.size`. This parameter allows reliable authentication for Portal Server on Sun Java™ System Application Server under a heavy load. The default value is 50 but can be changed. For example, a value of 50 should be used for a 4-CPU Ultra Sparc II or III machine.

Sun Java System Directory Server Tuning

If the Sun Java™ System Directory Server is shared by other applications, you may need to verify that those parameters are not conflicting with the other application's parameters tuning.

Enough virtual memory space must be provisioned for `/tmp/slapd-`*DSinstance*`1` and the total amount of used memory, including the allocated for database caching, should not exceed the size of physical memory to avoid paging. In any events, the cumulative values of `nsslapd-dbcachesize + nsslapd-cachememsize + fixed memory` used for `slapd` process itself cannot exceed the 4 GB of process address space. `Nsslapd` is a 32-bit application.

With regard to the sizing of resources pooling (connections and threads), Sun Java System Directory Server provides best performance with a concurrency level of around 15 for search type of operations.

The `perftune` script tunes `ns-slapd` threading, `db cache` and database file system mapping in the `/var/opt/mps/serverroot/slapd-`*hostname*`/config/dse.ldif` file as follows:

- Under `dn: cn=config` LDAP entry:
  - Adds the line `nsslapd-threadnumber` to `nThreads`. In most cases, default value (30) should be fine unless a fair amount of profile changes (LDAP writes) is expected, in which case, the script applies the following formula:

    `nThreads = 30 for 1 CPU, nThreads = 45 for 2 CPUs, nThreads = 60 for 3 CPUs, nThreads = 75for 4 CPUs.`
  - Specifies `nsslapd-accesslog-logging-enabled` to off to disable access log
- Under `dn: cn=config,cn=ldbm database,cn=plugins,cn=config LDAP` entry:
  - Adds the line `nsslapd-db-home-directory to /tmp/slapd-dsame1`
  - Changes the line `nsslapd-maxthreadsperconn` to 20

- Modifies the line nsslapd-dbcachesize to newSize where newSize = 1.2 * size of all db3 files located under /var/opt/mps/serverroot/slapd-*hostname*/db/userRoot.

- Under dn: cn=userRoot,cn=ldbm database,cn=plugins,cn=config LDAP entry, modifies the line nsslapd-cachememsize to newSize where newSize = 3 * the size of id2entry.db3.
- If NEED_REBOOT is yes, the nsslapd-maxdescriptors is set to 16384.

NOTE The file /var/opt/mps/severroot/slapd-hostname/config/dse.ldif is backed up in the format:

filename-orig-*date*-*pid*

Sun Java System Web Server 6.1 Tuning

The following describe the JVM tuning offered by the perftune script to help tune Sun Java™ System Web Server for Portal Server performance.

1. Specifies the following in magnus.conf located at *WebServer-base*/SUNWwbsvr/https-*hostname*/config
   - RqThrottle 256
   - RqThrottleMin 128
   - StackSize 393216
   - ThreadIncrement 20
   - ConnQueueSize 4096
   - ListenQ 4096

NOTE The *WebServer-base*/SUNWwbsvr/https-*hostname*/config/magnus.conf file is backed up in the format:

filename-orig-*date*-*pid*

2. Specifies the following in server.xml file at *WebServer-base*//https-*hostname*//config for JVM Tuning:

- `-Xms3G` (This value is set to 3G if memory is available, otherwise, by default, the Xms value is set to 128)
- `-Xmx3G` (This value is set to 3G if memory is available, otherwise, by default, the Xmx value is set to 128)
- `-Xss128K`
- `-Xloggc:/var/opt/SUNWappserver7/domains/domain1/server1/logs/gclog`
- `-XX:NewSize=384M`
- `-XX:MaxNewSize=384M`
- `-XX:MaxPermSize=64M`
- `-XX:PermSize=64M`
- `-XX:+UseParNewGC`
- `-XX:+UseConcMarkSweepGC`
- `-XX:MaxTenuringThreshold=1`
- `-XX:SoftRefLRUPolicyMSPerMB=1`
- `-XX:+CMSClassUnloadingEnabled`
- `-XX:+CMSPermGenSweepingEnabled`
- `-XX:+PrintGCTimeStamps`
- `-XX:+ShowMessageBoxOnError`
- `-XX:+OverrideDefaultLibthread`
- `-XX:+DisableExplicitGC`
- `-XX:+PrintGCDetails`
- `-XX:+PrintClassHistogram`

Sun Java System Application Server 7.0 Tuning

When deploying the Portal Server on the Sun™ Java System Application Server, the minimum and maximum heap size for the application server instance is set to 3 Gbytes.

The perftune script now includes a safeguard that prevents the perftune script from tuning a system that does not allow for a 3 Gbyte heap for the JVM.

The following describe the JVM tuning offered by the perftune script to help tune Sun Java™ System Application Server for Portal Server performance.

1. Specifies the following in init.conf located at *Deplaoy_Domain*/*Deploy_Instance*/config
   - RqThrottle 256
   - RqThrottleMin 128
   - StackSize 393216
   - ThreadIncrement 20
   - ConnQueueSize 4096
   - ListenQ 4096

NOTE The *Deplaoy_Domain*/*Deploy_Instance*/config/init.conf file is backed up in the format:

filename-orig-*date*-*pid*

2. Specifies the following JVM parameters in *Deplaoy_Domain*/*Deploy_Instance*/config/sever.xml:

- -Xms3G (This value is set to 3G if memory is available, otherwise, by default, the Xms value is set to 128)
- -Xmx3G (This value is set to 3G if memory is available, otherwise, by default, the Xmx value is set to 128)
- -Xss128K
- -Xloggc:/var/opt/SUNWappserver7/domains/domain1/server1/logs/gclog
- -XX:NewSize=384M
- -XX:MaxNewSize=384M
- -XX:MaxPermSize=64M
- -XX:PermSize=64M
- -XX:+UseParNewGC
- -XX:+UseConcMarkSweepGC

- -XX:MaxTenuringThreshold=1
- -XX:SoftRefLRUPolicyMSPerMB=1
- -XX:+CMSClassUnloadingEnabled
- -XX:+CMSPermGenSweepingEnabled
- -XX:+PrintGCTimeStamps
- -XX:+ShowMessageBoxOnError
- -XX:+OverrideDefaultLibthread
- -XX:+DisableExplicitGC
- -XX:+PrintGCDetails
- -XX:+PrintClassHistogram

NOTE The *Deplaoy_Domain*/*Deploy_Instance*/config/server.xml file is backed up in the format:

filename-orig-*date*-*pid*

NOTE If -Djava.security.policy is present in the file *Deplaoy_Domain*/*Deploy_Instance*/config/server.policy, -Djava.security.policy is replaced by -Djava.security.policy=*Deplaoy_Domain*/*Deploy_Instance*/config/server.policy.NEVERUSED

Setting Additional Sun Java System Application Server Parameters for Gateway Reliability

To achieve optimal performance using Secure Remote Access, configure your implementation as follows:

1. Modify the *AccessManager-base*/SUNWam/lib/AmConfig.properties file to set the notification threadpool size for the application server. At the top of the file just below the following lines:

```
 Sun, Sun Microsystems, the Sun logo, and iPlanet
 * are trademarks or registered trademarks of Sun Microsystems,
 * Inc. in the United States and other countries.
```

add the following lines to set the threadpool size to 200:

```
/*Notification Thread Pool Size*/
com.iplanet.am.notification.threadpool.size=200
```

2. Log into the Portal Server administration console with the user name amadmin and the passphrase you entered during the installation.
3. Select Service Management in the View menu.
4. Select SRA Configuration and then Gateway.
5. Select the default server and click Edit.
6. Check the Enable HTTP Connections checkbox.
7. In the HTTP Port field, type 80 and click Save.
8. Log in to the Sun Java System Application Server administration console as administrator (admin) by entering `http://`*fullservername*`:`*port* in your browser's web address field. The default port is 4848. Use the password you entered at installation.
9. Select the application server instance where you installed the Access Manager.
10. Click JVM Settings and then JVM Options.
11. In the JVM Option field, enter the following string:

    ```
    -Dhttp.keepAlive=false
    ```

12. Click Add and then Save.
13. Select the application server instance on which you will install Portal Server.

    The right pane shows that the configuration has changed.
14. Click Apply Changes.
15. Click Restart.
16. The application server should automatically restart.
17. On the server where the gateway is installed, go to the `/opt/SUNWps/bin/perf` directory and enter the following to run a script that will set tuning parameters for Secure Remote Access:

```
./perftune
```

18. Modify the *AccessManager-base*/SUNWam/lib/AmConfig.properties file to set the notification threadpool size for the gateway. At the top of the file just below the following lines:

```
Sun, Sun Microsystems, the Sun logo, and iPlanet
* are trademarks or registered trademarks of Sun Microsystems,
* Inc. in the United States and other countries.
```

add the following lines to set the threadpool size to 200:

```
/*Notification Thread Pool Size*/
com.iplanet.am.notification.threadpool.size=200
```

19. Go to the /opt/SUNWps/bin directory and modify the gateway file to set the -Dhttp.keepAlive option to false and to increase the settings for the -Xms and -Xmx heap size options.

20. Define the CMD settings options as follows:

NOTE Define the CMD settings as one line.

```
CMD="$JAVA_HOME/bin/java -server -Xms3G -Xmx3G
-XX:+OverrideDefaultLibthread -Xss128K
-XX:MaxPermSize=64M -XX:PermSize=64M -XX:MaxNewSize=512M
-XX:NewSize=512M -XX:+UseParNewGC -XX:+UseConcMarkSweepGC
-XX:MaxTenuringThreshold=1
-XX:SoftRefLRUPolicyMSPerMB=1
-XX:+CMSClassUnloadingEnabled -XX:+CMSPermGenSweepingEnabled
-XX:+PrintGCDetails
-XX:+PrintGCTimeStamps -XX:+PrintClassHistogram
-XX:+ShowMessageBoxOnError -XX:+DisableExplicitGC
-Xloggc:/var/opt/SUNWps/debug/gclog.$GW_INSTANCE -classpath
$CLASSPATH $DEFINES1 $DEFINES2 $DEFINES3 $DEFINES4 $DEFINES5
$PROXY_DEFINES $BOOT_CLASSPATH com.sun.portal.netlet.eproxy.EProxy"
```

21. Modify the /etc/opt/SUNWps/platform.conf.default file to set the gateway.protocol parameter to http and the gateway.port parameter to port 80 as follows:

```
gateway.protocol=http
```

```
gateway.port=80
```

22. Restart the gateway for the changes to take effect by typing the following command:

    PortalServer-base/SUNWps/bin/gateway -n default start

    where default is the default gateway profile created during installation.

Portal Server Desktop Tuning

The caller parameters are used to size the thread pool to render content through the providers. The caller pool is initialized to size 0. Items are added to the pool as they are used and returned. The caller pool can expand to a very large size, however, in the normal case it will only be as big as the number of channels on the user's Portal Desktop. In cases where there are multiple concurrent threads with the same sid, the pool may expand to an size that is n * m, where n = the number of concurrent same-sid threads and m = the number of channels on the Portal Desktop for the given sid.

The perftune script changes the following parameters for optimizing the Provider Caller Resource Pooling, in the /etc/opt/SUNWps/desktop/desktopconfig.properties file:

- Increases callerPoolMinSize to 128
- Increases callerPoolMaxSize to 512
- Increases callerPoolPartitionSize to 16
- Increases templateScanInterval to 3600

To minimize unnecessary memory growth due to spawning of Portal Desktop caller threads when performing long-run tests, these properties (except for templateScanInterval) should be changed back to their original default values.

Make the following changes to these properties:

- Change callerPoolMinSize back to 0
- Change callerPoolMaxSize back to 0
- Change callerPoolPartitionSize back to 0
- Increase the templateScanInterval property from 30 to 3600

Appendix A

SSO Adapter Templates and Configurations

This appendix describes how to configure the single sign-on (SSO) adapter in order to adjust options available to end users.

This appendix contains the following sections:

Overview of the Single Sign-On Adapter

The single sign-on adapter service allows end users to use applications, such as a portal server provider or any other web application, to gain authenticated access to various resource servers after signing in once. The resource servers that can be accessed depend on the implementations of the SSO Adapter interface that are available in the system. Currently, Sun™ Java System Portal Server provides SSO Adapters for the following resource servers: Address Book, Calendar, and Mail. Single Sign-On for the Instant Messaging channel is not achieved through SSO Adapter but through the use of the Sun Java System Identity Server authentication method. For information on this method, see the `authMethod` property in Table 17-1 on page 382. The Address Book, Calendar, and Mail services are available through the products:

- Sun™ Java System Calendar Server 5.1.1, 6.0, 6 2004Q2
- Sun™ Java System Messaging Server 5.2, 6.0, 6 2004Q2

Resource servers are typically accessed by an application using a standard application programming interface (API), such as JavaMail for accessing a mail server. To create an authenticated connection using the API, the API must be provided the configuration data for the connection. The purpose of the SSO Adapter is to provide this configuration data, and the SSO Adapter service is used to store that data.

The SSO Adapter service defines two levels of data, templates and configurations. An SSO Adapter template defines a class of connections that are going to be made available to users. A single template is used by many users. It defines data values that are the same for all users that use the template including default values and identification of what values can be edited by a user. Therefore, SSO Adapter templates are defined at a global service level.

An SSO Adapter configuration builds upon a template by providing data values that are specific to an organization, role, or user. A configuration references a template, and takes data values from the template for those properties that are not editable by the user. When an end user changes the user-editable properties of an SSO Adapter configuration, that configuration would then apply only to that one user.

A Sun Java System Portal Server communication channel that uses the SSO Adapter service references either a template or a configuration to get data values needed to obtain a connection to a resource server. If the channel references a template, and the user saves configuration information, the reference is changed to refer to a configuration instead. The configuration then references the template.

SSO Adapter Attributes Page

You can use the SSO Attributes page to administer:

- SSO Adapter Templates—You can create an SSO Adapter template, delete an SSO Adapter template, or modify the properties of an SSO Adapter template.
- SSO Adapter Configurations—You can create an SSO Adapter configuration, delete an SSO Adapter configuration, or modify the properties of an SSO Adapter template.

To Create an SSO Adapter Template

To Create an SSO Adapter Configuration

To Edit SSO Adapter Template Properties

To Edit an SSO Adapter Configuration Property

To Create an SSO Adapter Template

1. Log in to the Access Manager Administration Console.
2. Select the Service Configuration Tab.
3. Click New under SSO Adapter Template to add a SSO adapter template.

   The New Template page appears.
4. Type a template name and select an existing template from the menu.
5. Click Next.

   The Template Properties page appears.
6. Modify the properties as needed.
7. When done, click Save.

To Create an SSO Adapter Configuration

1. Click New under SSO Adapter Configuration to add an SSO adapter configuration.

   The New Configuration page appears.
2. Type a configuration name and select an SSO Adapter template from the menu.
3. Click Next.

   The Configuration Properties page appears.
4. Modify the properties as needed.

NOTE You can provide a default Host name which is your MAIL_HOST (DNS name or IP Address), or you can leave it blank.

5. When done, click Save.

To Edit SSO Adapter Template Properties

An SSO Adapter template can have the following property types

- Default — You can create, select, or delete a property that is provided by the system on behalf of a the user.
- Merge — You can create, select, or delete a property that is provided by the user.
- Encoded — You can select, or delete a property that is encoded

1. Click the Edit Properties link beside the SSO Adapter template to be modified.

   The Template Properties page appears.

2. Modify the properties as needed.

   You can modify the property types by clicking the link under Type.

3. When done, click Save.

To Edit an SSO Adapter Configuration Property

1. Click the Edit Properties link beside the SSO Adapter configuration to be modified.

   The Configuration Properties page appears.

2. Modify the properties as needed.
3. When done, click Save.

Glossary

Refer to the Java Enterprise System glossary (http://docs.sun.com/doc/816-6873) for a complete list of terms that are used in this documentation set.

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

X