# BEA WebLogic Enterprise Security™®

## WebLogic Server v8.1 Installation

# Contents

## 1. Overview

## 2. Preparing to Install

# 3. Installing

# 4. Post Installation Tasks

# 5. Integrating WebLogic Enterprise Security with WebLogic Portal

# 6. Uninstalling

# Overview

The document is organized as follows:

- Chapter 1 (this chapter) provides an introduction to the WebLogic Enterprise Security architecture component architecture.

- Chapter 2, "Preparing to Install," discusses system requirements (software and hardware) that you need to ensure are met before installing the product.

- Chapter 3, "Installing," describes how to install the product.

- Chapter 4, "Post Installation Tasks," describes the tasks that must be performed after you install the product.

- Chapter 5, "Integrating WebLogic Enterprise Security with WebLogic Portal," describes how to integrate WebLogic Enterprise Security using the sample portal domain shipped with WebLogic 8.1 Platform.

- Chapter 6, "Uninstalling," describes how to uninstall the product.

This section covers the following topics:

- "About this Document" on page 1-1

- "Product Overview" on page 1-3

## About this Document

This document provides security administrators with the information needed to install the BEA® WebLogic® Enterprise Security WebLogic Server Security Service Module.

This section covers the following topics:

- "Audience for This Guide" on page 1-2
- "Prerequisites for This Guide" on page 1-2
- "Product Documentation on the dev2dev Web Site" on page 1-2
- "Related Information" on page 1-3

## Audience for This Guide

It is assumed that reader understands web technologies and has a general understanding of the Microsoft Windows or UNIX operating systems being used, and WebLogic Server. The general audience for this installation guide includes:

- Security Administrators—Administrators who are responsible for installing and configuring the WebLogic Server Security Service Module and designing policy who work with other administrators to implement and maintain security configurations, authentication and authorization schemes, and to set up and maintain access to deployed application resources. Application Administrators have a general knowledge of security concepts and the WebLogic Server security architecture. They understand Java, XML, deployment descriptors, and can identify security events in server and audit logs

- Application Developers—Developers who focus on designing applications and work with other engineers, quality assurance (QA) technicians, and database teams to implement applications.

## Prerequisites for This Guide

Prior to reading this guide, you should read the *Introduction to BEA WebLogic Enterprise Security*. This document describes how the product works and provides conceptual information that is helpful to understanding the necessary installation components.

Additionally, BEA WebLogic Enterprise Security includes many unique terms and concepts that you need to understand. These terms and concepts—which you will encounter throughout the documentation—are defined in the *Glossary*.

## Product Documentation on the dev2dev Web Site

BEA product documentation, along with other information about BEA software, is available from the BEA dev2dev web site:

http://dev2dev.bea.com

To view the documentation for a particular product, select that product from the Product Centers menu on the left side of the screen on the dev2dev page. Select More Product Centers. From the BEA Products list, choose WebLogic Enterprise Security 4.2. The home page for this product is displayed. From the Resources menu, choose Documentation 4.2. The home page for the complete documentation set for the product and release you have selected is displayed.

## Related Information

The BEA corporate web site provides all documentation for BEA WebLogic Enterprise Security. Other BEA WebLogic Enterprise Security documents that may be of interest to the reader include:

- *Introduction to BEA WebLogic Enterprise Security*—This document provides an overview, conceptual, and architectural information for the WebLogic Server Security Service Module products.

- *BEA WebLogic Enterprise Security Administration Application Guide*—This document provides a complete overview of the product and includes step-by-step instructions on how to perform various administrative tasks.

- *BEA WebLogic Enterprise Security Policy Managers Guide*—This document defines the policy model used by BEA WebLogic Enterprise Security, and describes how to import and export policy data.

- *Developing Security Providers for BEA WebLogic Enterprise Security* —This document provides security vendors and security and application developers with the information needed to develop custom security providers.

- *Javadocs for Security Service Provider Interfaces*—This document provides reference documentation for the Security Service Provider Interfaces that are provided with and supported by this release of BEA WebLogic Enterprise Security.

## Product Overview

The WebLogic Server 8.1 Security Service Module is a security enhancement product that supports WebLogic Server, Version 8.1 (with SP3). The WebLogic Server Security Service Module provides an application programming interface (API) that allows security developers to insert security into their applications. These interfaces support the most commonly required security functions and are organized into services that are logically grouped by functionality. After you use the WebLogic Server Security Service Module interfaces to implement security

functions in your WebLogic Server, you can deploy and run your application on any instance of a WebLogic Server Security Service Module that supports the configuration requirements of your application. All WebLogic Server security-related functions remain available, but those functions are provided through the Security Service Module.

This section covers the following topics:

- "Security Environment" on page 1-4
- "Security Architecture Functional Description" on page 1-5

## Security Environment

Figure 1-1 shows the major components that make up the BEA WebLogic Enterprise Security environment.

**Figure 1-1  WebLogic Enterprise Security Environment**
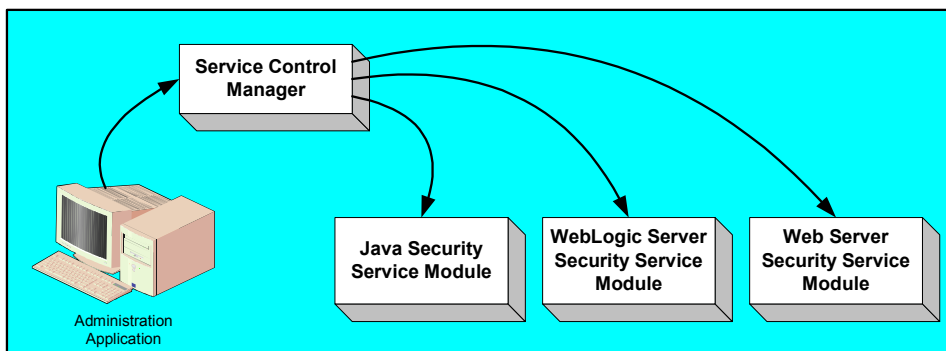


- Administration Application

  The Administration Application allows you to manage and configure multiple Security Service Modules. While Security Service Modules specify and consume configuration data and then services security requests accordingly, the Administration Application allows you to configure and display the security providers that are deployed in the Security Service Modules and modify the configuration data for those providers.

- Service Control Manager

  The Service Control Manager is an essential component of the BEA WebLogic Enterprise Security configuration provisioning mechanism and a key component of a fully-distributed

security enforcement architecture. A Service Control Manager is a machine agent that exposes a provisioning interface to the administration application to facilitate the management of a potentially large number of distributed Security Service Modules. A Service Control Manager can receive and store metadata updates, both full and incremental, initiated by the Administration Application.

The Administration Application uses the provisioning mechanism to distribute configuration and policy data to each service module where it is consumed locally (see Figure 1-2). Security Service Modules (which can be distributed throughout an enterprise) can be embedded in Java applications, application servers, and web servers.

After you use the Administration Application to configure an instance of a Security Service Module with configuration and policy data, the module requires no any additional communication with the Service Control Manager to perform security functions. However, the Service Control Manager maintains communication with the Security Service Module to distribute full and incremental updates.

**Figure 1-2  Deploying Configuration and Policy Data**
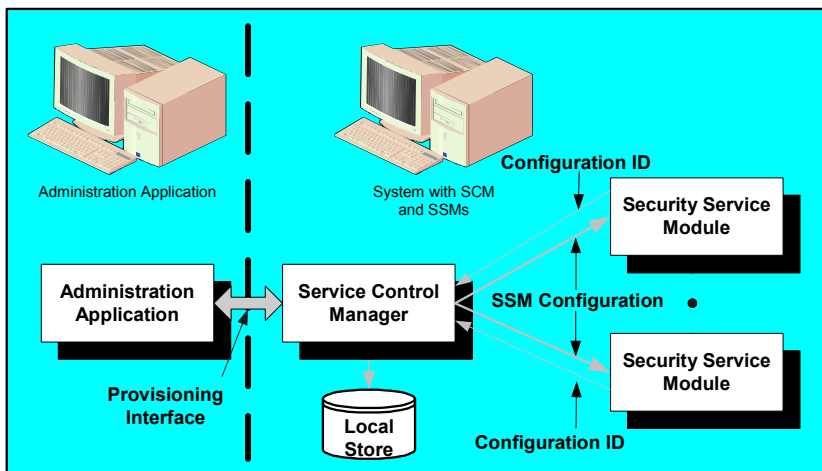


# Security Architecture Functional Description

Figure 1-3 shows the major components of the BEA WebLogic Enterprise security architecture. The WebLogic Server Security Service Module comprises the Security Services APIs, the security framework, and the security providers.

The following topics describe these components:

**Figure 1-3  Architectural Components**
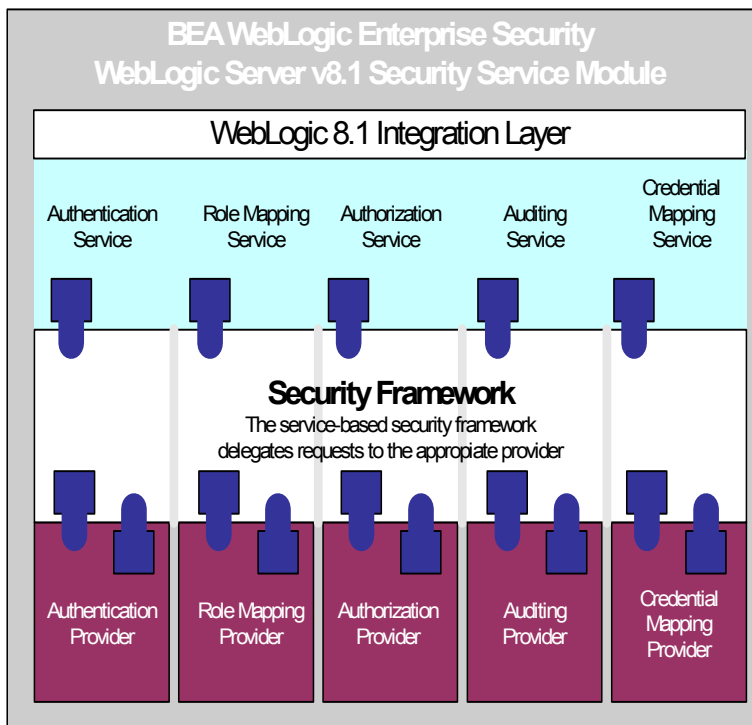


## Security Service APIs

The WebLogic Server 8.1 Security Service Module supports the following security service application programming interfaces (APIs):

- Authentication Service API

- Role Mapping Service API

- Authorization Service API

- Auditing Service API

- Credential Mapping Service API

For descriptions of these APIs, see Java Security Service Module API*s* in *Programming Security for Java Applications*. For Javadoc for these APIs, see Javadocs for Java Applications.

## Security Framework

The primary function of the Security Framework is to provide an application programming interface (API) that security and application developers can use to implement security functions in WebLogic Server. Within that context, the Security Framework also acts as an intermediary between security functions that you implement using the configuration tools and the security providers that are configured into the Security Service Module.

## Security Providers

When you install the Administration Application or a Security Service Module, a JAR file is deployed that contains all the security providers that ship with the product. However, before any of the security providers can be used, you must configure them through the Administration Application. You have the option of configuring either the security providers that ship with the product or custom security providers, that you develop or purchase from third-party security vendors.

**Note:** To use custom security providers with a Security Service Module, you must deploy the security providers MBean JAR file (MJF) to both the providers directory on the machine on which you install the Security Service Module product and on the Administration Server.

The Administration Application supports the following types of security providers:

- Authentication provider

- Identity Assertion provider

- Principal Validation provider

- Authorization provider

- Credential Mapping provider

- Role Mapping provider

- Auditing provider

For more information on security providers, see *Introduction to BEA WebLogic Enterprise Security*.

For more information on how to develop custom security providers, see *Developing Security Providers for BEA WebLogic Enterprise Security*.

# Preparing to Install

This section provides the information needed to install the BEA WebLogic Server, Version 8.1 Security Service Module, including system requirements and prerequisite software and hardware. It does not include information for installing the BEA WebLogic Enterprise Security Administration Application or other Security Service Modules.

This section covers the following topics:

## Installation and Distribution

BEA WebLogic Enterprise Security products are distributed and installed using the BEA Installation and Distribution System, which provides a complete framework for the following:

- Distribution of BEA products by download from the BEA web site.

- Installation and uninstallation of the WebLogic Server, Version 8.1 Security Service Module including documentation.

BEA WebLogic Enterprise Security is distributed on both the BEA web site and on CD-ROM.

# Web Distribution

If you want to install the product by downloading it from the BEA web site, contact BEA Sales at http://www.bea.com/framework.jsp?CNT=sales1.htm&FP=/content/about/contact/ and request a download.

The package installer downloads a stand-alone version of the installation program that contains the complete WebLogic Server v8.1 Security Service Module. The package installer is approximately 55 MB.

Documentation is available from the product documentation home page. Be sure to download the most up-to-date information from the BEA web site at:
http://e-docs.bea.com/wles/docs42/download.html.

# CD-ROM Distribution

If you purchased BEA WebLogic Enterprise Security from your local sales representative, you will find the following items in the product box:

Four CD-ROMs:

- Disk 1 of 4 contains the following BEA WebLogic Enterprise Security products:
  - Administration Application software for Microsoft Windows platforms
  - Security Service Modules software for Microsoft Windows platforms
  - Documentation in both PDF and HTML format

- Disk 2 of 4 contains the following BEA WebLogic Enterprise Security products:
  - Administration Application software for Linux and Sun Solaris
  - Security Service Modules software for Linux and Sun Solaris

- Disks 3 of 4 contains the BEA WebLogic Enterprise Security metadirectory software for Microsoft Windows platforms. This product is used with the Administration Application to integrate user repositories.

- Disks 4 of 4 contains the BEA WebLogic Enterprise Security metadirectory software for Linux and Sun Solaris platforms.

The following printed documents:

- Introduction to BEA WebLogic Enterprise Security

- BEA Software License and Limited Warranty pamphlet

● Customer Support Quick Reference and Other Important Information card

# Installation Prerequisites

The WebLogic Server, version 8.1 Security Service Module requires certain software components to operate properly. Review these requirements before installing the product. For additional information on the BEA WebLogic Enterprise Security products, see: http://www.bea.com/wles.

● "System Requirements" on page 2-3

● "Licensing" on page 2-5

● "Requirements for Reinstalling the SSM" on page 2-5

## System Requirements

Table 2-1 lists the system requirements for the machine on which you install the WebLogic Server, Version 8.1 Security Service Module.

**Note:** The machine on which you install the WebLogic Server Security Service Module must have a static IP address. The IP address is used by the Security Service Module and Service Control Manager for connectivity. Also, on a Windows platform, the file system used must be NTFS, not FAT. To check the file system format, open Windows Explorer and right-click the hard drive on which you intend to do the installation and select `Properties`.

**Table 2-1 System Requirements**

| Use | Component and Version |
| --- | --- |
| WebLogic Server 8.1 with Service Pack 3 or Service Pack 4 | You can download this product from this location:<br><br>http://commerce.bea.com/showallversions.jsp?family=WLS<br><br>**Note:** The BEA WebLogic Enterprise Security installation program requires a Sun Microsystems Java 2 Platform Standard Edition (J2SE), Version 1.4 Java run-time environment (JRE). The JRE provided by WebLogic Server 8.1 SP3 and SP4 satisfies this requirement. |
| Platforms supported | The WebLogic Server, Version 8.1 Security Service Module runs on any of the following platforms:<br>• Intel Pentium compatible with Microsoft Windows NT 4.0<br>• Intel Pentium compatible with Microsoft Windows 2000 Professional<br>• Intel Pentium compatible with Microsoft Windows 2000 Server/Advanced Server<br>• Intel Pentium compatible with Microsoft Windows XP Server/Advanced Server<br>• SUN Microsystems Sparc with Solaris versions 8 and 9<br>• Linux Red Hat Advanced Server 2.1 and 3.0 (Update 4) |
| A Configured WebLogic Server Domain | If you do not have domain configured for the WebLogic Server, use the WebLogic Server Configuration Wizard to configure a domain. Make a note of the domain name and its location. This information is required to install the WebLogic Server, Version 8.1 Security Service Module. |
| BEA WebLogic Enterprise Security Administration Application | You must install the Administration Application before you install the WebLogic Server, Version 8.1 Security Service Module software distribution. |

**Table 2-1  System Requirements (Continued)**

| Use | Component and Version |
| --- | --- |
| Memory | With the Sun Java SDK— 64 MB of RAM minimum, 256 MB or more is recommended for each instance. |
| Hard Disk Space | For installation on a Microsoft Windows platform—About 100 MB of free storage space is required for the installed product and about 100 MB of temporary storage space is required by the installer. |
| | For installation on Unix systems—About 100 MB of free storage space is required for the installed product and about 100 MB of temporary storage space is required by the installer. |

## Licensing

The product software cannot be used without a valid license. When you install WebLogic Server, Version 8.1 Security Service Module, the installation program creates an evaluation license. The evaluation license expires in 90 days.

To use the WebLogic Server, Version 8.1 Security Service Module in a production environment, you must purchase a license. For information about purchasing a license, contact your BEA Sales Representative.

## Requirements for Reinstalling the SSM

If you are installing the security Service Module on a computer on which a WebLogic Enterprise Security SSM was previously installed, refer to "Uninstalling" on page 6-1 and make sure all of the uninstall steps were completed; otherwise the installation may fail.

# Selecting Directories for the Installation

During installation, you need to specify locations for the following directories:

- "BEA Home Directory" on page 2-5
- "Product Installation Directory" on page 2-7

## BEA Home Directory

The files and directories in the BEA Home directory are described in your WebLogic documentation. When you install the product, you are prompted to specify a BEA Home

directory. You should specify the same BEA Home directory that you specified when you installed WebLogic Server 8.1. The BEA Home directory is a repository for common files that are used by multiple BEA products installed on the same machine. For this reason, the BEA Home directory can be considered a "central support directory" for the BEA products installed on your system.

The files in the BEA Home directory are essential to ensuring that BEA software operates correctly on your system. They perform the following types of functions:

- Ensure that licensing works correctly for the installed BEA products

- Facilitate checking of cross-product dependencies during installation

If you choose the default product installation directory, you will see additional directories in the BEA Home directory, such as weblogic81 (the WebLogic Server installation directory) and user_projects (a folder for WebLogic domains that you create). Although the default location for the WebLogic Enterprise Security installation directory is within the BEA Home directory, you can select a different location outside the BEA Home directory.

During installation, you are prompted to choose an existing BEA Home (BEA_HOME) directory or specify a path to create a new BEA Home directory. If you choose to create a new directory, the installation program automatically creates the directory for you.

**Note:** For a BEA Home directory, you are allowed to install each version of a BEA product that uses the BEA Home directory convention only once. For example, you can install WebLogic Server 8.1 and associate it with a BEA Home directory, and that BEA Home directory can also be associated with an installation of BEA WebLogic Enterprise Security. It cannot be associated with another installation of WebLogic Server 8.1.

## Understanding the Functions of the BEA Home Directory

The files and directories in the BEA Home (BEA_HOME) directory are described in your WebLogic documentation. Although it is possible to create more than one BEA Home directory, BEA recommends that you avoid doing so. In almost all situations, a single BEA Home directory is sufficient. There may be circumstances, however, in which you prefer to maintain separate development and production environments on a single machine, each containing a separate product stack. With two directories, you can update your development environment (in a BEA Home directory) without modifying the production environment until you are ready to do so.

# Product Installation Directory

The product installation directory contains all the software components used to administer BEA WebLogic Enterprise Security. During installation, you are prompted to choose a product installation directory. If you accept the default, the software is installed in the following directory:

`c:\bea\wles42-ssm\wls-ssm` (Windows)

`/opt/bea/wles42-ssm/wls-ssm` (Sun Solaris and Linux)

where `c:\bea` is the `BEA_HOME` directory and `wles42-ssm\wls-ssm` is the product installation directory. You can specify any name and location on your system for your product installation directory and there is no requirement that you name the directory `wles42-ssm\wls-ssm` or create it under the BEA Home directory.

# Installing

The following sections provide the information you need to install and uninstall the WebLogic
Server 8.1 Security Service Module:

**Note:**   For installation information on other security service modules, see the associated
installation guides.

- "Before you Begin" on page 3-1
- "Starting the Installation Program" on page 3-2
- "Running the Installation Program" on page 3-6
- "What's Next" on page 3-9

## Before you Begin

Before you begin this installation procedure, make sure you do the following:

**Note:**   If you start the installation process from the command line or from a script, you can
specify the -log option to generate a verbose installation log. For instructions on how to
generate a verbose log file during installation, see "Generating a Verbose Installation
Log" on page 3-2.

- Install WebLogic Server 8.1.
- Download and read the Release Notes from:
  `http://e-docs.bea.com/wles/docs42/relnotes/index.html`
- Install the Administration Application and related components.
- Ensure the system requirements are met as described in "Installation Prerequisites" on
  page 2-3.

# Generating a Verbose Installation Log

If you start the installation process from the command line or from a script, you can specify the -log option to generate a verbose installation log. The installation log lists messages about events during the installation process, including informational, warning, error, and fatal messages. This can be especially useful for silent installations.

**Note:** You may see some warning messages during in the installation log. However, unless there is a fatal error, the installation program will complete the installation successfully. The installation user interface will indicate the success or failure of the installation, and the installation log file will include an entry indicating that the installation was successful.

To generate a verbose log file during installation, include the -log=/full_path_to_log_file option in the command line or script. For example:

**For Windows:**

```
wles422ssm_win32.exe -log=D:\logs\wles_install.log
```

**For Sun Solaris:**

```
wles422ssm_solaris32.bin -log=/opt/logs/wles_install.log
```

**For Linux:**

```
wles422ssm_linux32.bin -log=/opt/logs/wles_install.log
```

The path must be the full path to a file name. If the file does not exist, all folders in the path must exist before you execute the command or the installation program will not create the log file.

# Starting the Installation Program

The procedure for starting the installation program varies depending the platform on which install BEA WebLogic Enterprise Security. Therefore, separate instructions are provide for each supported platform.

**Note:** In a production environment, BEA recommends that you install the Security Service Modules on machines other than the machine on which the Administration Server is installed.

To start the installation program, refer to the appropriate section listed below:

- "Starting the Installation Program on a Windows Platform" on page 3-3

- "Starting the Installation Program on a Sun Solaris Platform" on page 3-4

- "Starting the Installation Program on a Linux Platform" on page 3-5

# Starting the Installation Program on a Windows Platform

**Note:** Do *not* install the software from a network drive. Download the software distribution to a local drive on your machine and install it from there. Also, on a Windows platform, the file system used must be NTFS, not FAT. To check the file system format, open Windows Explorer and right-click the hard drive on which you intend to do the installation and select `Properties`.

To install the application in a Microsoft Windows environment:

1. Shut down any programs that are running.

2. Log in to the local Administrators group.

3. If you are installing from a CD-ROM, go to step 4. If you want to install the product by downloading it from the BEA web site:

   a. Contact BEA Sales at
   http://www.bea.com/framework.jsp?CNT=sales1.htm&FP=/content/about/contact/ and request a download.

   b. Go to the directory where you downloaded the installation file and double-click `wles421w18_win32.exe`.

   The BEA Installer - Security Service Module for WLS 8.1 window appears (see Figure 3-1).

   c. Proceed to "Running the Installation Program" on page 3-6.

4. If you are installing from a CD-ROM:

   a. Insert Disk 1 into the CD-ROM drive.

   If the installation program does not start automatically, open Windows Explorer and double-click the CD-ROM icon.

   b. From the installation CD, double-click `wles421w18.exe`.

   The BEA Installer - Security Service Module for WLS 8.1 window appears (see Figure 3-1).

   c. Proceed to "Running the Installation Program" on page 3-6.

# Starting the Installation Program on a Sun Solaris Platform

To run graphical-mode installation, your console must support a Java-based GUI. If the installation program determines that your system cannot support a Java-based GUI, the installation program automatically starts console-mode installation.

1. Shut down any programs that are running.

2. Log in to the machine as root (or su root).

3. Open a command-line shell.

4. If you are installing from a CD-ROM, go to step 5. If you want to install the product by downloading it from the BEA web site:

   a. Contact BEA Sales at
      http://www.bea.com/framework.jsp?CNT=sales1.htm&FP=/content/about/contact/ and request a download.

   b. Go to the directory where you downloaded the file and change the protection on the install file:

      ```
      chmod u+x wles422ssm_solaris32.bin
      ```

   c. Start the installation: `wles422ssm_solaris32.bin`

      The BEA Installer - Security Service Module for WLS 8.1 window appears (see Figure 3-1).

   d. Proceed to "Running the Installation Program" on page 3-6.

5. If you are installing from a CD-ROM:

   a. Insert the Disk 1 into the CD-ROM drive.

   b. In a command shell, go to the directory where you installed the CD-ROM and change the protection on the install file:

      ```
      chmod a+x wles422ssm_solaris32.bin
      ```

   c. Enter this command to start the installation: `wles422ssm_solaris32.bin`

      The BEA Installer - Security Service Module for WLS 8.1 window appears (see Figure 3-1).

   d. Proceed to "Running the Installation Program" on page 3-6.

# Starting the Installation Program on a Linux Platform

To run graphical-mode installation, your console must support a Java-based GUI. If the installation program determines that your system cannot support a Java-based GUI, the installation program automatically starts console-mode installation.

1. Shut down any programs that are running.

2. Log in to the machine as root (or su root).

3. Set your DISPLAY variable if needed.

4. Open a command-line shell.

5. If you are installing from a CD-ROM, go to step 6. If you want to install the product by downloading it from the BEA web site:

   a. Contact BEA Sales at
      http://www.bea.com/framework.jsp?CNT=sales1.htm&FP=/content/about/contact/ and
      request a download.

   b. Go to the directory where you downloaded the file and change the protection on the install file:

      ```
      chmod u+x wles422ssm_linux32.bin
      ```

   c. Start the installation: wles422ssm_linux32.bin

      The BEA Installer - Security Service Module for WLS 8.1 window appears (see Figure 3-1).

   d. Proceed to "Running the Installation Program" on page 3-6.

6. If you are installing from a CD-ROM:

   a. Insert the Disk 1 into the CD-ROM drive.

   b. In a command shell, go to the directory where you installed the CD-ROM and enter this command to change the protection on the install file:

      ```
      chmod u+x wles422ssm_linux32.bin
      ```

   c. Enter this command to start the installation: wles422ssm_linux32.bin

      The BEA Installer - Security Service Module for WLS 8.1 window appears (see Figure 3-1).

d. Proceed to "Running the Installation Program" on page 3-6.

**Figure 3-1  BEA Installer - WebLogic Server, Version 8.1 Security Service Module Window**
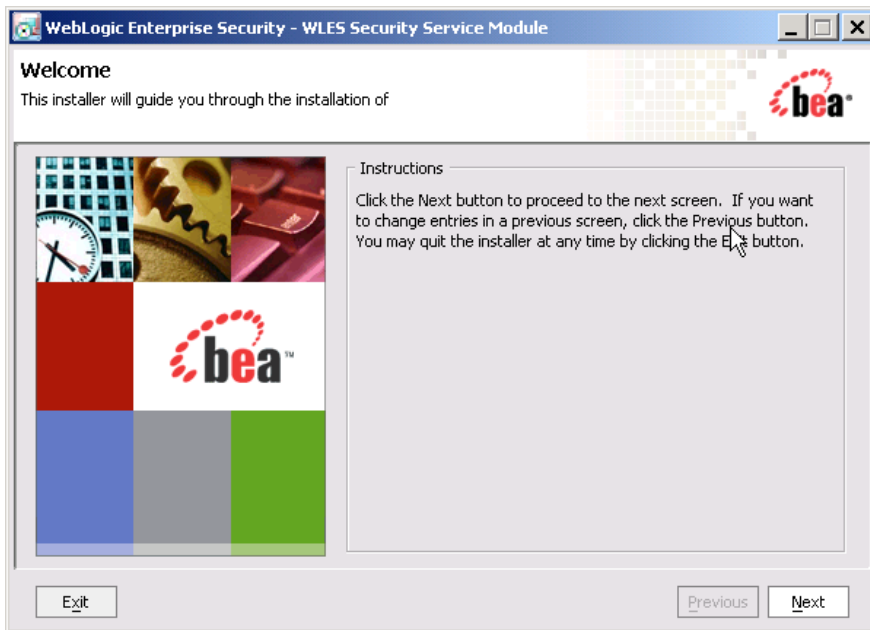


# Running the Installation Program

The installation program prompts you to enter specific information about your system and configuration as described in Table 3-1. To complete this procedure you need the following information:

- Name of the BEA_HOME directory

- Name of the product directory

**Note:**  If this is the first WebLogic Enterprise Security product you have installed on this machine, the Service Control Manager is also included as part of the installation (which requires additional inputs, such as the Service Control Manager directory).

**Table 3-1  Running the Installation Program**

| In this Window: | Perform this Action: |
|---|---|
| Welcome | Click Next to proceed, or cancel the installation at any time by clicking Exit. |
| BEA License Agreement | Read the BEA Software License Agreement, and then select Yes to indicate your acceptance of the terms of the agreement. To continue with the installation, you must accept the terms of the license agreement, click Yes, and then click Next. |
| Choose BEA Home Directory | Specify the BEA Home directory that serves as the central support directory for all BEA products installed on the target system. If you already have a BEA Home directory on your system, you can select that directory (recommended) or create a new BEA Home directory. If you choose to create a new directory, the installer program automatically creates the directory for you. For details about the BEA Home directory, see "BEA Home Directory" on page 2-5. |
| Choose Product Directory | Specify the directory in which you want to install the product software, and then click Next. You can accept the default product directory (C:\bea\wles42-wls-ssm) or you can create a new product directory. |
| | **Note:** If you are installing on a machine with existing BEA WebLogic Enterprise products or on a machine that you intend to install other BEA WebLogic Enterprise products (for example, the Administration Application or another Security Service Module) you *must* select a different directory. |
| | For additional information and a description of the resulting directory structure, see "Product Installation Directory" on page 2-7. |
| | If you choose to create a new directory, the installation program automatically creates the directory for you, if necessary. |
| | When you click Next, the installation program begins copying the components you specified to your system. If you have installed other products then you will see Installation Complete. Otherwise, continue installing the Service Control Manager. |
| Choose Service Control Manager Directory | Specify the directory in which to install the Service Control Manager. You can accept the default directory (wles42-scm) or you can create a new one. |
| | Click Next to continue. |

**Table 3-1  Running the Installation Program (Continued)**

| In this Window: | Perform this Action: |
|---|---|
| Select Users and Groups | Specify the user names and group names to use for the Service Control Manager and Administration Application. You can accept the default settings or create a new ones.<br><br>**Note:** When installing this product for use in a production environment, BEA recommends that you set these passwords to known values; otherwise you will not be able to modify them later. For example, you may want to modify these passwords to comply with organizational requirements.<br><br>**Admin User** (`asiadmin`)—A local user account used to start the Administration Application components.<br><br>**Admin Group** (`asiadgrp`)—Administration Application group. Members of this group have full access to Administration Application and log files; they can start and stop the Administration Application components.<br><br>**SCM User** (`scmuser`)—A local user account used to start the Service Control Manager.<br><br>**Security Group** (`asiusers`)—Service Control Manager Group. Members of this group are allowed to use the WebLogic Enterprise Security products.<br><br>Click Next to continue. |
| Confirm User Selection | If the users and groups do not exist, they are created for you. If you specified users and groups other than the defaults, verify the values you entered are correct, and then click Next. |
| User Passwords (Windows only) | Specify the password for the Administration Application User and Service Control Manager User. You can also choose the default passwords that are randomly generated.<br><br>**Note:** If any of the users exist you must enter their passwords; the passwords are not generated randomly. Passwords are case sensitive. If you are installing the Administration Application in a production environment, BEA recommends using secure user names and passwords, and not those that are randomly generated.<br><br>Click Next to continue. |

**Table 3-1  Running the Installation Program (Continued)**

| In this Window: | Perform this Action: |
|---|---|
| Choose Network Interface | Select the network interfaces to which to bind the Service Control Manager. This is the IP Address used to listen for requests to provision policy and configuration data. |
| | **Note:** If you are installing the security service module in a production environment with more than one network card, you want to select a protected (internal) interface; you do not want to expose the Service Control Manager through a public address. |
| | Click Next to continue. |
| Configure Enterprise Domain for Service Control Manager | **Enterprise Domain Name**—The enterprise domain name is used to link all of the WebLogic Enterprise Security components. |
| | **Note:** This is same enterprise domain name that you entered when you installed the BEA WebLogic Enterprise Security Administration Application. |
| | **SCM Logical Name—**The name you assign to the Service Control Manager during this installation. |
| | **SCM Port**—Port used by the Service Control Manager to receive configuration and policy data from the Administration Application; may not be used by any other server. |
| | **Note:** The SCM values are different the SCM values defined when you installed the BEA WebLogic Enterprise Security Administration Application. |
| | **Primary Server URL**—The address used by your Administration Application. |
| | **Backup Server URL**—If you have a second Administration Application installed for the purpose of failover or backup, enter its address here. This field is optional and may be left blank. |
| Installation Complete | Indicates that the installation completed successfully. Click Done to finish the installation. |

# What's Next

Now that you have installed the necessary software, you must enroll the Service Control Manager, create an instance of the Security Service Module and enroll the instance, and then start the services. For additional instructions, see "Post Installation Tasks" on page 4-1.

Installing

# Post Installation Tasks

This section describes each task you must perform after you install the product software and discusses other considerations.

**Note:** If you want to use the WebLogic Server 8.1 Security Service Module to integrate WebLogic Enterprise Security with WebLogic Portal server and portal applications, skip this section and go to "Integrating WebLogic Enterprise Security with WebLogic Portal" on page 5-1.

**Note:** Some of the procedures described here require basic knowledge of both WebLogic Server and WebLogic Enterprise Security products. If you need assistance with any task, see the Administration Console online help or the *Administration Application Guide* for more details. It is assumed that you know the location of the products you have installed, including the WebLogic Server, the Security Service Module, and the Administration Server.

# Enrolling the Service Control Manager

This section describes how to enroll the Service Control Manager. Each machine on which you install a Security Service Module must have one (and only one) enrolled Service Control Manager. You only need to follow this procedure if you installed the Security Service Module on a machine other then the one that contains the Administration Application.

**Note:** While you can use the demonstration digital certificate to enroll in a development environment, you should never use it in a production environment.

To enroll the Service Control Manager, perform the following steps:

1. Open a command window and go to the Service Control Manager `/bin` directory, for example:

   ```
   BEA_HOME/wles42-scm/bin
   ```

   Where:

   `BEA_HOME` is the directory where your BEA products are installed.

   `wles42-scm` is the directory where you installed the Service Control Manager.

2. Run the following script:

```
enrolltool demo
```

The Enrollment menu appears.

3. Type: 5 and press <ENTER>, and do one of the following:

   – If the domain you to which you want to enroll the SSM is listed, go to step 4.

   – If the domain you want to use is not listed, type: 3, press <ENTER> to register the domain, enter the following information, Type: 5 and press <ENTER> again:

   ```
   Enter Enterprise Domain Name :> (For example: asi)
   Enter Primary Admin URL :> (For example:
   https://adminmachine:7010/asi)
   Secondary Admin URL :> (This value is optional. Same format as primary
   URL)
   SCM name :> (For example: ssmmachinename_ssm)
   SCM port :> (Default: 7010)
   ```

4. Select the domain you want to use and press <ENTER>.

5. Enter the admin username and password. This is the username and password of the security administrator that is enrolling the SCM.

6. Enter and confirm the following passwords:

   – **Private key password**—Protects the identity of the Service Control Manager you are creating

   – **identity.jks password**—Protects the `ssl\identity.jks` keystore. This keystore contains the identities for all the components you are enrolling.

   – **peer.jks password**—Protects the `ssl\peer.jks` keystore. This keystore contains the certificates of components with which this Security Service Module can communicate.

   – **trust.jks password**—Protects the `ssl\trust.jks` keystore. This keystore contains the WebLogic Enterprise Security CA certificate used for enrollment.

# Configuring a Service Control Manager

You configure a Service Control Manager (SCM) for each of the machines on which you have installed one of more Security Service Modules (SSM). Each machine must have one (and only one) configured Service Control Manager. For example, if you install an SSM on the same machine as the Administration Application, you must use the `adminconfig` SCM, which was configured for you when you installed the Administration Application.

**Note:** When you use the Instance Wizard to create an instance of a SSM on a machine, you link the instance to a SCM by name. When you install multiple SSMs of different types (Web

Server or Web Services, WebLogic Server 8.1, and Java) on the same machine, they all must use the same SCM.

To configure a SCM, see the Administration Application Console Help and use the WebLogic Enterprise Security Administration Console.

The instructions for performing this task are also available in "Configuring a Service Control Manager" in the *BEA WebLogic Enterprise Administration Application Guide*.

# Configuring and Binding a WebLogic 8.1 Security Service Module

Configure a SSM with the security providers that you require for the WebLogic 8.1 SSM and bind it to the SCM. You have the option of configuring either the default security providers that ship with the product or custom security providers, which you develop or purchase from third-party security vendors. The Java Security Service Module supports the following types of security providers:

- Authentication provider

- Authorization provider

- Auditing provider

- Credential mapping provider

- Identity asserter

- Principal validator

- Role mapping provider

To configure these providers and bind the configuration to the SCM, perform the following steps:

1. In the Administration Console, expand the Security Configuration node in the left pane, and click Unbound Configurations. The Unbound Security Service Module Configurations page displays.

2. Click Create a New Security Service Module Configuration. The Edit Security Service Module Configuration page displays.

3. In the Configuration ID text box, enter an identity for the SSM (for example, weblogic81_ssm) and click Create.

> **Note:** Later, when you use the Instance Wizard to create an instance of the SSM to which this secruity configuration will be applied, you will use the Configuration ID to link the SSM instance to this security configuration.

4. Click the Providers tab and create the desired providers.

5. Click on the SCM that you previously configured for this SSM. The Edit a Service Control Manager Configuration page displays.

6. Click on the Binding tab and bind the WebLogic 8.1 SSM configuration to the SCM.

# Creating an Instance of the WebLogic 8.1 Security Service Module

Before starting a WebLogic Server Security Service Module, you must first create an instance of the Security Service Module using the Instance Wizard. You can create any number of instances of the Security Service Module. You must then enroll each instance that you want to use. Each instance has its own set of providers.

To create an instance of a Security Service Module:

1. Start the Instance Wizard:
   - On Widows, click Start>Programs>BEA WebLogic Enterprise Security>WebLogic Server 8.1 Security Service Module>Create New Instance.
   - On UNIX, if you are using X-windows run: `instancewizard.sh`

     **Note:** If you are not using X-windows, use a console based installer.

2. In the Instance Name text box, enter the name to assign to this instance.

3. In the Authorization Engine port text box, enter the port number for the Authorization and Role Mapping engine to use.

4. In the Configuration ID text box, enter the configuration identifier to use with this instance. The Configuration ID was specified when you configured your module, as described in "Configuring and Binding a WebLogic 8.1 Security Service Module" on page 4-4.

5. From the Enterprise Domain drop-down box, select the domain to which this instance belongs.

6. Click Next.

7. In the Location text box, enter the location for this instance. The default instance is located within the installation directory of the Security Service Module.

8. Click Next.

9. Click Done when the instance wizard completes.

# Enrolling the Instance of the Security Service Module

You must have the Administration Server running prior to enrolling the Security Service Module.

**Note:**   While you can use the demonstration digital certificate in a development environment, you should never use it in a production environment.

To enroll the Security Service Module:

1. Open a command window and go to the Security Service Module instance `/adm` directory: `BEA_HOME/wles42-ssm/wls-ssm/instance/`*`instancename`*`/adm`, where *`instancename`* is the name you assigned to the instance when you created it.

2. Run the following script:

   ```
   enroll demo
   ```

3. Enter the `admin` username and password. This is the username and password of the Security Administrator doing the enrollment (if you used the default values and have not yet changed them, the default username is `system` and the password is `weblogic`).

4. Enter and confirm the following passwords:

   – **Private key password**—This password protects the identity of the Security Service Module that you are creating.

   – **identity.jks password**—This password protects the `ssl\identity.jks` keystore. This keystore contains the identities for all the components you are enrolling.

   – **peer.jks password**—This password protects the `ssl\peer.jks` keystore. This keystore contains the certificates of components with which this Security Service Module can communicate.

   – **trust.jks password**—This password protects the `ssl\trust.jks` keystore. This keystore contains the WebLogic Enterprise Security CA certificate used for enrollment.

# Creating a WebLogic Server Domain

The Security Service Module requires that you create a WebLogic Server domain in the following location:

```
BEA_HOME/user_projects/domains/mydomain
```

You can use the WebLogic Server Configuration Wizard to create a domain or create it manually. The domain includes a `startWebLogic` file, which you are instructed to modify in "Modifying the startWebLogic File" on page 4-7.

# Modifying the startWebLogic File

The WebLogic startup script does the following:

- Sets environment variables.

- Invokes the java weblogic.Server command, which starts a JVM that is configured to run a WebLogic Server instance.

Before you can start a WebLogic Server that uses BEA WebLogic Enterprise Security, you must edit the `startWebLogic` file that is located in the WebLogic Server domain directory, for example:

```
BEA_HOME/user_projects/domains/mydomain
```

where:

BEA_HOME is the directory where your BEA products are installed.

user_projects is the directory where your WebLogic Server user projects are located.

domains is the directory where your WebLogic Server domain instances are located.

mydomain is the name of the WebLogic Server domain instance you are using.

See Listing 4-1 for an example of a modified `startWebLogic.cmd` file. To edit the `startWebLogic` file, do the following:

1. Before the CLASSPATH is set, add a call to the set-wls-env script file in your the bin directory for your instance. For example:

```
BEA_HOME/wles42-ssm/wls-ssm/instance/wls81ssm/bin
```

Where:

BEA_HOME is the directory where your BEA products are installed.

wles42-ssm is the directory where you installed the Security Service Module.

instance is the directory where all instances are stored.

wls81ssm is the name of the Security Service Module instance you created earlier.

2. For example, if you created an instance called myInstance, the call looks like this:

On Windows:

```
        call
        "C:\bea\wles42-ssm\wls-ssm\instance\myInstance\bin\set-wls-env.bat"
```

On Unix:

```
    . "/bea/wles42-ssm/wls-ssm/instance/myInstance/bin/set-wls-env.sh"
```

3.  Add the following line to the CLASSPATH:

    On Windows:

    ```
    %WLES_PRE_CLASSPATH% and %WLES_POST_CLASSPATH%
    ```

    On Unix:

    ```
    ${WLES_PRE_CLASSPATH} and ${WLES_POST_CLASSPATH}
    ```

4.  On Windows, add quotes to %JAVA_HOME%\bin\java in the weblogic.Server command.

    ```
    "%JAVA_HOME%\bin\java"
    ```

5.  Add the following line to the weblogic.Server command.

    On Windows:

    ```
    %WLES_JAVA_OPTIONS%
    ```

    On Unix:

    ```
    ${WLES_JAVA_OPTIONS}
    ```

**Listing 4-1   Modifying the startWebLogic.cmd File for Windows**

```
...

set SERVER_NAME=myserver

call "C:\BEA_HOME\wles42-ssm\wls-ssm\instance\myInstance\bin\set-wls-env.bat"

set CLASSPATH=%WLES_PRE_CLASSPATH%;%WEBLOGIC_CLASSPATH%;
%POINTBASE_CLASSPATH%;%JAVA_HOME%\jre\lib\rt.jar;
%WL_HOME%\server\lib\webservices.jar;%CLASSPATH%;
%WLES_POST_CLASSPATH%

@REM Call WebLogic Server
echo .
echo CLASSPATH=%CLASSPATH%
echo .
echo PATH=%PATH%
echo .
```

```
echo ***************************************************
echo *  To start WebLogic Server, use a username and   *
echo *  password assigned to an admin-level user.  For *
echo *  server administration, use the WebLogic Server *
echo *  console at http:\\[hostname]:[port]\console    *
echo ***************************************************

"%JAVA_HOME%\bin\java" %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS% %WLES_JAVA_OPTIONS%
-Dweblogic.Name=%SERVER_NAME%
-Dweblogic.ProductionModeEnabled=%PRODUCTION_MODE%
-Djava.security.policy="%WL_HOME%\server\lib\weblogic.policy" weblogic.Server

ENDLOCAL
```

# Defining Security Properties

You can use the security.properties file to set the necessary security properties. To set the security properties, create a security.properties file and put it in the WebLogic Server domain directory; for example:

> BEA_HOME/user_projects/domains/*mydomain*

Where:

> BEA_HOME is the directory where your BEA products are installed.

> user_projects is the directory where your WebLogic projects are located.

> domains is the directory where your WebLogic Server domain instances are located.

> *mydomain* is the name of the WebLogic Server domain instance you are using.

Include the information shown in Listing 4-2 in the security.properties file, where:

> wles.realm is the value of the Configuration ID entered for Security Service Module using the Administration Console (see Security Configuration in your Console Help or in the *Administration Application Guide*).

> wles.default.realm must be set to the same value as wles.realm.

**Note:** The security.properties file is not required if you add these parameters to Java Options.

**Listing 4-2  Security.properties File**

```
wles.realm=ConfigurationID
wles.default.realm=ConfigurationID
```

# Starting and Stopping Processes

After you install the Security Service Module, create the instance, and enroll it, you must start the necessary processes by running the appropriate batch or shell scripts. Before you start these processes, make sure that the Administration Server and all of its services are running.

For each machine, you must start the following processes:

- One Service Control Manager
- One Authorization and Role Mapping Engine (ARME) for each Security Service Module instance.

For instructions on how to start and stop the required processes, see "Starting and Stopping Processes for Security Service Modules" in the *Administration Application Guide.*

# Additional Post-Installation Considerations

When using the Database Authentication provider, ASI Authorization provider and ASI Role Mapping provider, refer to the following sections for important information:

- "Setting the Boot Login for WebLogic Server" on page 4-10
- "Creating a WebLogic Boot Policy" on page 4-11
- "Creating a WebLogic Console Policy" on page 4-14
- "Protecting Resources" on page 4-15

## Setting the Boot Login for WebLogic Server

The WebLogic Server uses the login information contained in the boot.properties file to start the server. This file contains a username and password that must match a username and password in the configured authentication policy. The boot.properties file is located in the WebLogic Server domain directory on the machine on which the Security Service Module is installed, for example:

```
BEA_HOME/user_projects/domains/mydomain
```

Where:

> `BEA_HOME` is the directory where your BEA products are installed.
>
> `user_projects` is the directory where your WebLogic user projects are located.
>
> `domains` is the directory where your WebLogic Server domain instances are located.
>
> `mydomain` is the name of the WebLogic Server domain instance you are using.

If you used a username of `system` and a password of `weblogic`, then modify WebLogic Server `boot.properties` in the domain as follows:

```
user = system
password = weblogic
```

The next time you start the WebLogic Server, the username and password you specified are encrypted.

# Creating a WebLogic Boot Policy

Before you can use the ASI Authorization provider with the WebLogic Server, you need to configure a boot policy, and then distribute it to the WebLogic Server 8.1 Security Service Module. If you need instructions on how to perform any of these tasks, see the Console Help for details. You may also want to refer to the *Policy Managers Guide* for information on how the policy language is constructed and how it appears in the console.

To configure and distribute a boot policy, perform the following tasks:

- "Creating the User Identity" on page 4-11

- "Creating Resources for the Defined User" on page 4-12

- "Creating a Policy to Protect the Resource" on page 4-12

- "Creating a Role with Resource Access Privileges" on page 4-13

- "Binding the Resource to the ASI Authorization Provider" on page 4-13

- "Distributing the Policy to the Security Service Module" on page 4-14

## Creating the User Identity

To create the user identity named `wlesusers`, perform these steps:

1. Using the Administration Console, create an Identity directory called `wlesusers`.

   a. Open the Identity folder and click Identity.

  b. Click New, in the Name text box, enter `wlesusers`, and then click OK.

2. Within this directory, create a user named `system` and set the password for system to `weblogic`. Replace `system` and `weblogic` with the values used in `boot.properties` file.

  a. Click Users, click New, enter `system`, and click OK.

  b. Click Edit, click Set Password, enter the `weblogic`, and click OK.

  c. Click OK.

## Creating Resources for the Defined User

To create resources for the defined user, `wlesusers`, create the following resources below the resource called `policy`:

 `wlsserver` as a bound application node.

 `wlsserver/shared` as virtual

 `wlsserver/shared/svr`

1. Click Resources and then click New.

2. In the Name box, type `wlsserver`, select Binding from the Type drop-down menu, and then click OK.

3. Select `wlsserver` and click Configure.

4. From the Type drop-down menu, select Binding Application, check Distribution Point, and then click OK.

5. Select `wlsserver`, click New, enter `shared` in the name box, and then click OK.

6. Select `shared`, click Configure, check Allow Virtual Resources, and then click OK.

7. Select `shared`, click New, enter `svr` in the name box, and then click OK.

## Creating a Policy to Protect the Resource

Create the following policy:

 `grant(any, //app/policy/`*`wlsserver`*`/shared/svr, //role/Admin) if true;`

1. Click Policy and click New.

2. In the Create Rule page, click `any` in the Select Privileges from Group list box, and then click Add.

3.  Select the Resources tab, expand the `wlsserver` and `shared` nodes in the Child Resources list box, select `svr`, and then click Add.

4.  Select the Policy Subjects tab, select `Admin` from the Roles List list box, click Add, and click OK.

## Creating a Role with Resource Access Privileges

Create the following role:

```
grant(//role/Admin, //app/policy/wlsserver, //user/wlesusers/system/)
    if true;
```

1.  Open the Role folder, click Role Policy, and then click New.

2.  In the Create Role Policy page, click Roles, select Admin from the Available Roles list box, and click Add.

3.  Click the Resources tab, select `wlsserver` in the Child Resources list box, and click Add.

4.  Click the Identities tab, select Users from the drop-down menu, change the directory to `wlesusers`, select `system` from the users list box, click Add, and click OK.

## Binding the Resource to the ASI Authorization Provider

To bind the resource `//app/policy/wlsserver` to the ASI Authorization provider for this Security Service Module, perform the following steps:

1.  Open the Security Configuration and Security Control Manager folders.

2.  Open the Security Service Module folder and click Authorization.

3.  The Authorization page appears.

4.  Click Create a new ASI Authorization Provider.

5.  The Edit ASI Authorization Provider page appears.

6.  Enter a name for the provider in the Name text box, and then click Create.

7.  Click the Details tab, set the Identity Directory to `wlesusers`, set the Application Directory Parent to `//app/policy/wlsserver`.

8.  Click Apply.

9.  Click the Bindings tab and select the resource you want to bind to the provider from the Bind drop-down menu, and then click Bind.

### Distributing the Policy to the Security Service Module

Distribute the policy to the WebLogic Server v8.1 Security Service Module.

For information on how to distribute policy, see Deployment in the Administration Console Help or in the *Administration Application Guide*. Make sure to verify the results of your distribution.

# Creating a WebLogic Console Policy

Before you can login into the WebLogic Server Administration Console, you need to configure a console policy and then distribute it to the WebLogic Server 8.1 Security Service Module. This is needed if you want to access the WebLogic Server Administration Console.

To configure and distribute a WebLogic Server Administration Console policy, do the following on the WebLogic Enterprise Security Administration Console:

1. Create the following resource below `//app/policy/wlsserver/console`.

    a. Click Resources. The Resources page appears.

    b. Select `wlsserver`, click New, enter `console` in the name box, and then click OK.

    c. Select `console`, click Configure, check Allow Virtual Resources, and then click OK.

2. Create the following rule:

   `grant(any, //app/policy/wlsserver/console, //role/Admin) if true;`

    a. Click Policy and then click New.

    b. In the Create Rule page, click `any` in the Select Privileges from Group list box, and then click Add.

    c. Select the Resources tab, expand the `wlsserver` in the Child Resources list box, select `console`, and then click Add.

    d. Select the Policy Subjects tab, select `Admin` from the Roles List list box, click Add, and then click OK.

3. Distribute the policy to the WebLogic Server 8.1 Security Service Module. For information on how to distribute policy, see Distributing Policy in your Console Help or in the *Administration Application Guide*. Make sure to verify the results of your distribution.

## Protecting Resources

When you secure an EJB using a WebLogic Server 8.1 Security Service Module, you must follow these steps if you want to use the WebLogic Enterprise Security providers instead of the default WebLogic providers.

1.  Modify the EJB deployment descriptor (`ejb-jar.xml`) so that the assembly-descriptor does not have any method-permissions set to unchecked or excluded.

    If either of these settings is present in the deployment descriptor, then the EJB container enforces them rather than calling into the security subsystem.

2.  Set the following system property to true, indicating that the EJB container delegates other security checks to the security subsystem.

    ```
    weblogic.security.fullyDelegateAuthorization=true
    ```

3.  Add this line to the `WLES_JAVA_OPTIONS` in the `set-wls-env.sh` script.

# Protecting a Cluster of WebLogic Servers

If you want to protect a cluster of WebLogic Servers using WebLogic Enterprise Security, you must make some addition changes to the security configuration and resource configuration. For information on how to protect cluster of WebLogic Servers, see the following topics:

"Security Configuration" on page 4-15

"Resource Configuration" on page 4-17

"Policy Configuration" on page 4-18

## Security Configuration

Figure 4-1 shows a Security Service Module configuration named `myrealm`, located under a Service Control Manager named `adminconfig` in the WebLogic Enterprise Security Administration Console. Your actual Security Service Module configuration will vary from this example based on the needs of your WebLogic domain.

**Figure 4-1  Service Control Manager Configuration**



[Figure 4-2](#) shows a configuration for a cluster of four WebLogic Servers: one administration server (adm) and three managed servers (svr1, svr2, svr3), with one Security Service Module instance for each server. The Service Control Manager on both machines must use the same Configuration Name (adminconfig). Each Security Service Module must have a unique Instance Name and Port number per machine, but always shares a common Configuration ID (myrealm) across all machines. Thus, each server uses the same security provider configuration and receives the same policy.

**Figure 4-2  WebLogic Server Clusters**
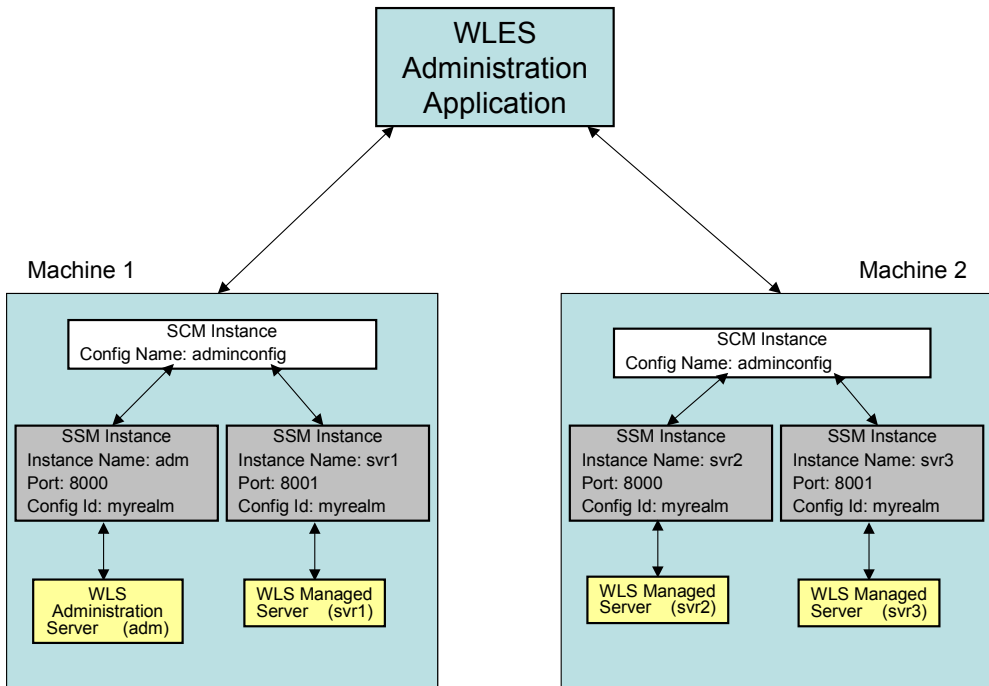


## Resource Configuration

You must also create the following two resources shown in Figure 4-3, setting them both to virtual.

The myrealm/wl_management_internal1 resource is accessed on the cluster's administration server by the WebLogic Admin Console to view WebLogic Server related log files.

The myrealm/wl_management_internal2 resource is accessed on the cluster's administration server by a managed server during bootstrap and file distribution operations.

The myrealm/bea_wls_internal is accessed when one managed server is synchronizing with another managed server.

The `myrealm/wl_management_internal1`, `myrealm/wl_management_internal2` and `myrealm/bea_wls_internal` resources must be configured to allow virtual resources.

**Figure 4-3  Resources for Managing WebLogic Server Clusters**

```
policy
    MedrecApp
    myrealm
        bea_wls_internal
        console
        portalApp
        shared
        testportalapp
        wl_management_internal1
        wl_management_internal2
```

# Policy Configuration

You must create the policy listed in Table 4-1. Also, ensure that there is a role policy that maps the `Everyone` role to the group `allusers` in your identity directory.

**Table 4-1  Policy Configuration**

| Privileges | Resources | Policy Subjects | Conditions |
|---|---|---|---|
| any | myrealm/bea_wls_internal | role/Everyone | none |
| any | myrealm/wl_management_internal1, myrealm/wl_management_internal2 | role/Everyone | none |

# What's Next?

You have completed the installation and configuration of the WebLogic Server 8.1 Security Service Module. Your Security Administrator can now configure additional security services using the security providers for your Security Service Module, through the WebLogic Enterprise Security Administration Console. If you configured the providers as part of the post install, you can now make changes to your configuration using the console.

Before you continue to configure security services, read the information on security configuration in the Administration Console help or in the *Administration Application Guide*. This section

provides additional information on how to configure the Service Control Manager, the Security Service Module, and the providers, and then deploy your changes.

Post Installation Tasks

# Integrating WebLogic Enterprise Security with WebLogic Portal

This section covers the following topics:

## Introduction

Integrating WebLogic Enterprise Security with WebLogic Portal server and portal application results in an enhanced set of security services for use in protecting WebLogic Portal (see Figure 5-1). WebLogic Enterprise Security participates in the authoring and management of policy for WebLogic Portal resources. Once WebLogic Enterprise Security is integrated with WebLogic Portal, you use WebLogic Enterprise Security Administration Application to manage resources related to portal desktops, books, pages, and portlets.

Therefore, the intent is that you use WebLogic Enterprise Security for authorization of the resources associated with a portal application as well as standard WebLogic Server J2EE resources. The benefit of using WebLogic Enterprise Security to manage visitor entitlements is that it offers fine-grained, dynamic role-based authorization. Additionally, WebLogic Enterprise Security allows you to have common security policies for a heterogeneous environment. For example, you may have a single security infrastructure that supports WebLogic Portal, WebLogic Server, and custom applications.

The WebLogic Enterprise Security security service does not replace all of the management functionality provided by the Portal Administration Tools. For example, as shown in Figure 5-1,

WebLogic Enterprise Security is not used to manage administrative users and resources associated with Portal Delegated Administration and Portal Content Management; use the Portal Administration Tools for those management tasks.

**Figure 5-1  Portal Integration Overview**
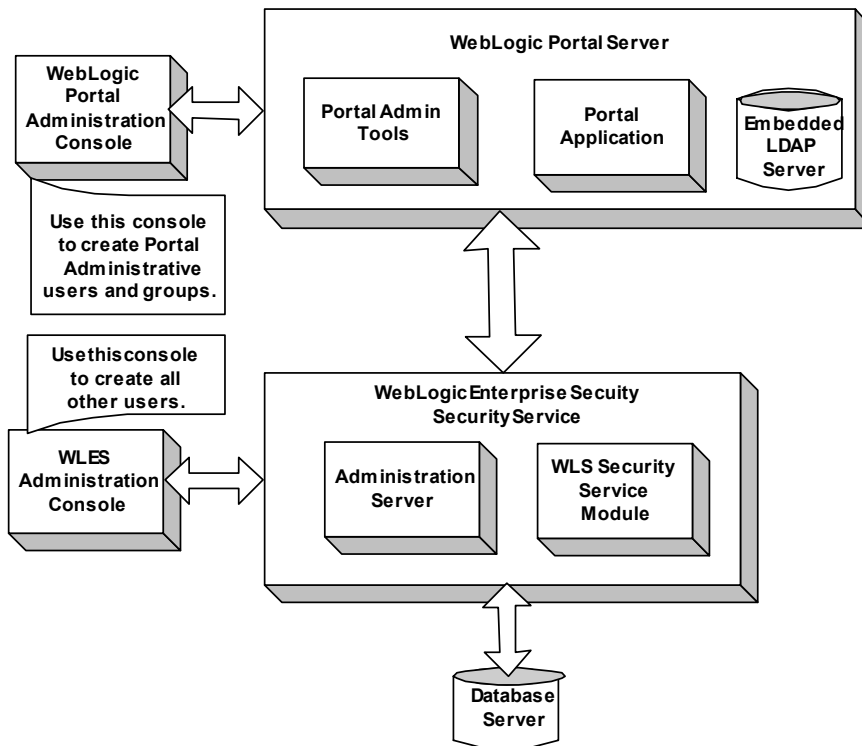


WebLogic Enterprise Security enables you to write, deploy, and manage fine-grained policy for the authorization of WebLogic Portal application resources. You can use WebLogic Enterprise Security to protect portal desktops, books, pages, portlets, and application look and feels.

For more information, see the following topics:

- "Integration Features" on page 5-3
- "Supported Use-case Scenario" on page 5-3
- "Constraints and Limitations" on page 5-4

# Integration Features

WebLogic Enterprise Security can be use with either WebLogic Server 8.1 Service Pack 3 or Service Pack 4. While several different security providers can be used with WebLogic Portal, the following security providers include enhanced WebLogic portal support:

- The ASI Authorization provider—Enables you to use WebLogic Enterprise Security to write, deploy, and manage fine-grained authorization of WebLogic Portal application resources related to desktops, books, pages, portlets, and application look and feels.

- The Database Authentication provider—Enables user and group controls. This provider is integrated into the Portal Administration Tools environment so as to handle user and group queries.

  **Note:** Use of the Database Authentication provider with WebLogic Portal is not mandatory. You may use other authentication providers as well.

- ASI Role Mapper—Enables dynamic mapping of principals (users and groups) to security roles at runtime. Role Mapping determines which security roles to apply to the principals stored in a subject when the subject is attempting to perform an operation on a portal application resource. Because this operation usually involves gaining access to the portal application resource, the ASI Role Mapper is used in conjunction with the ASI Authorization provider.

# Supported Use-case Scenario

The following use-case scenario is supported when you integrate WebLogic Enterprise Security with WebLogic Portal:

- The WebLogic Enterprise Security Administration Application assumes responsibility for management and policy of resources related to Portal visitor entitlements.

- The WebLogic Enterprise Security Administration Application is responsible for management of J2EE application resources associated with Portal applications and portal administration tools.

- The Portal Administration Tools continue to be responsible for the rest of Portal Management and Administration, including the creation and management of portal administrative users and groups.

**Note:** To implement this use case scenario, you must define the security configuration as specified in "Creating the Portal Application Security Configuration" on page 5-6.

# Constraints and Limitations

When integrated with WebLogic Enterprise Security, WebLogic Portal has the following constraints and limitations:

- Portal application administrators will not use the WebLogic Portal Administration Tools to create and manage visitor entitlements.

  Use of WebLogic Enterprise Security with a Portal application implies that an administrator will not use the Portal Administration Tools to create "Visitor Entitlements" on portal desktops, books, pages, and portlets. Managing visitor entitlements from the Portal Administration Tools is not a supported use case.

- Users will not use application deployment descriptors to deploy policy.

  Use of Deployment descriptors to deploy policy is not supported in WebLogic Enterprise Security 4.2 SP1.

- Migration of existing portal application policy is not supported.

  WebLogic Enterprise Security does not to support the migration of visitor entitlements policy for existing portal applications. There are no facilities for migrating any information from the WebLogic Server embedded LDAP store.

- WebLogic Enterprise Security does not replace or in any way interfere with the use of the Portal Administration Tools for the management of resource structures associated with Portal Delegated Administration and Portal Content Management.

  You cannot use WebLogic Enterprise Security to manage the resources associated with Portal Delegated Administration and Portal Content Management. WebLogic Enterprise Security does not support Portal Unified User Profiles.

# Integration Pre-Requisites

Before you begin, you must ensure that the following pre-requisites are satisfied:

- The WebLogic Platform/Portal 8.1, with Service Pack 3 or Service Pack 4, must be installed on the local machine.

- The WebLogic Enterprise Security WebLogic Server v8.1 Security Service Module, Version 4.2, Service Pack 2 must be installed on the local machine.

- You must have access to an Administration Console that is running on the WebLogic Enterprise Security Administration Application, Version 4.2, Service Pack 2 on either the local machine or a remote machine.

● You have created a WebLogic Portal domain on the local machine and installed a portal application in that domain.

# Integrating with WebLogic Portal

This section describes how to integrate WebLogic Enterprise Security with WebLogic Portal. Once integrated, you can use the WebLogic Enterprise Security Administration Console to write and deploy security policy to protect WebLogic Portal application resources.

**Note:** While the instructions provided is this section use a WebLogic Portal server and the sample portal application that ships with the WebLogic Platform 8.1 software distribution, you can use this procedure to integrate WebLogic Enterprise Security with your WebLogic Portal server and portal application.

To integrate WebLogic Enterprise Security with WebLogic Portal, perform the following tasks:

# Creating the Portal Application Security Configuration

This section describes how to create a new security configuration named `myrealm`. A security configuration defines the set of security providers to use for adjudication, authentication, auditing, authorization, role mapping, and credential mapping services. The security configuration named `myrealm` matches the default security configuration for the WebLogic Portal sample portal application.

**Note:** To implement the use-case scenario described in "Supported Use-case Scenario" on page 5-3, you are required to defined the security configuration as described in this section. This security configuration is a requirement; it is not optional.

Refer to Table 5-1 and use the WebLogic Enterprise Security Administration Application to configure the security providers listed there. Set the Configuration ID to `myrealm`. For instructions on creating a security configuration, see "Configuring a Security Service Module" in the *Administration Application Guide* and the Console Help.

**Table 5-1  Portal Security Configuration**

| Security Provider | Configuration Settings |
|---|---|
| ASI Adjudication Provider | Uncheck the Require Unanimous Permit check box, and click Create. |
| Log4j Auditor | Accept the default settings, and click Create. |
| Database Authentication Provider | Set the Control Flag to SUFFICIENT, and click Create. For the Details tab settings, except for the Identity Scope, the parameters are populated automatically. Set the Identity Scope to `myusers`, and click Apply. |
|  | **Note:** Even though you set the Identity Scope to `myusers`, you do not actually create the `myusers` identity until you perform the steps in "Creating the Realm Resource" on page 5-13. |
| WebLogic Authentication Provider | Set the Control Flag to SUFFICIENT, and click Create. |
|  | **Note:** Make sure the authentication providers are configured in the following order: 1) Database Authenticator and 2) WebLogic Authenticator. |
|  | **Note:** The WebLogic Authentication provider can be replaced with another authentication provider that supports write access to users and groups. |
| ASI Authorization Provider | On the General tab, accept the default settings, and click Create. On the Details tab, set the Identity Scope to `myusers`, and click Apply. |

**Table 5-1  Portal Security Configuration (Continued)**

| Security Provider | Configuration Settings |
|---|---|
| WebLogic Authorization Provider | Uncheck the Policy Deployment Enabled check box, and click Create. |
| WebLogic Credential Mapper Provider | Uncheck the Credential Mapping Deployment Enabled check box, and click Create. |
| ASI Role Mapping Provider | On the General tab, accept the default settings, and click Create. On the Details tab, set the Identity Scope to myusers, and click Apply. |
| WebLogic Role Mapper Provider | Uncheck the Role Deployment Enabled check box, and click Create. |

# Binding the Security Configuration

The security configuration must be bound to a Service Control Manager.

To bind the myrealm security configuration, see "Binding a Security Service Module to a Service Control Manager" in the *Administration Application Guide* and the Console Help

# Distributing the Security Configuration

The myrealm security configuration must be distributed.

To distribute the myrealm security configuration, see "Distributing Configuration" in the *Administration Application Guide* and the Console Help.

# Creating an Instance of the Security Service Module

Before starting a WebLogic Server Security Service Module, you must first create an instance of the WebLogic Server 8.1 Security Service Module using the Create New Instance Wizard.

To create an instance of a WebLogic Server 8.1 Security Service Module, see "Creating an Instance of the WebLogic 8.1 Security Service Module" on page 4-5.

# Enrolling the Instance of the Security Service Module

You must have the Administration Server running prior to enrolling the Security Service Module.

**Note:**  While you can use the demonstration digital certificate in a development environment, you should never use it in a production environment.

To enroll a security service module, see .

# Modifying the Portal Server startWeblogic File

Before you can start a WebLogic Portal server that uses BEA WebLogic Enterprise Security, you must modify the `startWeblogic` file that is located in the WebLogic Portal domain that you are using for you WebLogic Portal server. These modifications are needed so that portal connects the WebLogic Portal domain to the distributed `myrealm` security configuration on startup.

The `startWeblogic` file for the WebLogic Portal domain named `portalDomain` is located at: `BEA_HOME\user_projects\domains\portalDomain`

To edit the `startWeblogic` file, perform the steps:

**Note:** This procedure assumes a Windows installation of WebLogic Portal in the directory `c:\bea` with an WebLogic Server 8.1 Security Service Module instance named `portalInstance`.

1. Before you modify the script, make sure to make a backup copy. For example, for Microsoft Windows, copy `startWeblogic.cmd` to `startWeblogic.cmd.original`.

2. Add a line to call the environment batch file `set-wls-env.bat`. For example, add it below the line: `set SAVE_JAVA_OPTIONS=`

   ```
   call
   "c:\bea\wles42-ssm\wls-ssm\instance\portalInstance\bin\set-wls-env.bat"
   ```

3. Add the WebLogic Enterprise Security `classpath` variables to the `classpath`. For example, add the following text before the line: `echo CLASSPATH=%CLASSPATH%`

   ```
   set CLASSPATH=%WLES_PRE_CLASSPATH%;%CLASSPATH%;%WLES_POST_CLASSPATH%
   ```

4. Add `%WLES_JAVA_OPTIONS%` to the server start command after `%JAVA_OPTIONS%`. Listing 5-1 shows, in bold text, where to make this change.

**Listing 5-1  Adding WLES_JAVA_OPTIONS to the startWebLogic File**

```
if "%WLS_REDIRECT_LOG%"=="" (
     echo Starting WLS with line:
     echo %JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS%
%WLES_JAVA_OPTIONS% -Dweblogic.Name=%SERVER_NAME%
-Djava.security.policy=%WL_HOME%\server\lib\weblogic.policy
```

```
%PROXY_SETTINGS% %SERVER_CLASS%
      %JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS%
%WLES_JAVA_OPTIONS% -Dweblogic.Name=%SERVER_NAME%
-Djava.security.policy=%WL_HOME%\server\lib\weblogic.policy
%PROXY_SETTINGS% %SERVER_CLASS%
) else (
      echo Redirecting output from WLS window to %WLS_REDIRECT_LOG%
      %JAVA_HOME%\bin\java %JAVA_VM% %MEM_ARGS% %JAVA_OPTIONS%
%WLES_JAVA_OPTIONS% -Dweblogic.Name=%SERVER_NAME%
-Djava.security.policy=%WL_HOME%\server\lib\weblogic.policy
%PROXY_SETTINGS% %SERVER_CLASS%  > "%WLS_REDIRECT_LOG%" 2>&1
)
```

## Creating the security.properties File

Create a text file named `security.properties` and place it in the portal domain directory. You use this file to define the WebLogic Enterprise Security realm and the default realm. Listing 5-2 shows the content of this file for the realm `myrealm`.

**Listing 5-2   security.properties File**

```
# WebLogic Enterprise Security Configuration File
#
# This file contains WebLogic Enterprise Security configuration
# properties. By default, the WebLogic Enterprise Security runtime
# looks for a property file called 'security.properties' in the
# working directory.
wles.realm=myrealm
wles.default.realm=myrealm
```

## Replacing the Portal p13n_ejb.jar File

To integrate WebLogic Enterprise Security with WebLogic Portal, you must replace the `p13n_ejb.jar` file in the top-level portal application directory with the version of that file that is provided in the WebLogic Enterprise Security software distribution. The WebLogic Enterprise

Security version of `p13n_ejb.jar` is located in `BEA_HOME/wles42-ssm/wls-ssm/lib` directory.

**Note:** BEA WebLogic Enterprise Security 4.2 SP2 includes two versions of p13n_ejb.jar, the WebLogic Server 8.1 SP3 version: p13n_ejb_81SP3.jar, and the SP4 version: p13n_ejb_81SP4.jar. Be sure to use the correct version.

**Note:** Because these instructions assume that you are using the sample portal application that ships with WebLogic Portal, this procedure instructs you to replace the `p13n_ejb.jar` in the sample portal application. To use a different portal application, replace `p13n_ejb.jar` in that application as well.

To replace `p13n_ejb.jar`, perform the following steps:

1. Rename the portal version of the `p13n_ejb.jar`. For example, rename it to `p13n_ejb.jar.original`. The portal application version of this file is located in `BEA_HOME/weblogic81/samples/portal/portalApp`.

2. Depending on which version of the WebLogic Server 8.1 you are using (SP3 or SP4), copy either `p13n_ejb_81SP3.jar` or `p13n_ejb_81SP4.jar` from `BEA_HOME/wles42-ssm/wls-ssm/lib/` to `BEA_HOME/weblogic81/samples/portal/portalApp` and `rename` it to `p13n_ejb.jar`.

## Replacing the Portal p13n_system.jar File

To integrate WebLogici Enterprise Security with WebLogic Portal, you must replace the `p13n_system.jar` file in the `BEA_HOME/weblogic81/p13n/lib` directory with the version of that file that is provided in the WebLogic Enterprise Security software distribution. The WebLogic Enterprise Security version of this file is located in `BEA_HOME/wles42-ssm/wls-ssm/lib` directory.

**Note:** BEA WebLogic Enterprise Security 4.2 SP2 includes two versions of `p13n_system.jar`, the WebLogic Server 8.1 SP3 version: `p13n_system_81SP3.jar`, and the SP4 version: `p13n_system_81SP4.jar`. Be sure to use the correct version.

**Note:** Once you replace `p13n_system.jar` in the `/lib` directory of the WebLogic Platform installation, all portal domains configured for that installation must be WebLogic Enterprise Security enabled.

To replace `p13n_system.jar`, perform the following steps:

1. Rename the portal version of the `p13n_system.jar`. For example, rename it to `p13n_system.jar.original`. The portal version of this file is located in `BEA_HOME/weblogic81/p13n/lib`.

2. Depending on which version of the WebLogic Server 8.1 you are using (SP3 or SP4), copy either `p13n_system_81SP3.jar` or `p13n_system_81SP4.jar` from `BEA_HOME/wles42-ssm/wls-ssm/lib/` to `BEA_HOME/weblogic81/p13n/lib` and rename it to `p13n_system.jar`.

# Replacing the DefaultAuthorizerInit.ldift File

WebLogic Server 8.1 uses the `DefaultAuthorizerInit.ldift` file to establish access controls for J2EE resources. By default, WebLogic Server allows access to all J2EE resources to users in the `Everyone` role. To protect these resources, WebLogic Server provides the Administration Console and other tools to define security policies.

When using WebLogic Enterprise Security, there is a need to supersede the WebLogic Server J2EE access controls. The `DefaultAuthorizerInit.ldift` file, provided in the WebLogic Enterprise Security 4.2 SP2 for the WebLogic Server 8.1 Security Service Module, is used for this purpose.

To enable the WebLogic Enterprise Security `DefaultAuthorizerInit.ldift` file to supersede WebLogic Server access controls for J2EE resources in the sample portal application, perform the following steps:

1. Copy `DefaultAuthorizer.ldift` from `BEA_HOME/wles42-ssm/wls-ssm/domain` to `BEA_HOME/user_projects/domains/mydomain`.

2. If the `/ldap` directory exists at the following location, delete it:
   `BEA_HOME/user_projects/domains/portalDomain/portalServer`

**Note:** Because these instructions assume that you are using the sample portal domain that ships with WebLogic Portal, this procedure instructs you to delete the `ldap` directory in the `portalDomain/portalServer` directory. Repeat the above steps for all WebLogic Enterprise Security enabled portal domains.

# Configuring Policy for the Portal Application

Developing a policy typically begins by determining which resources you need to protect. You then create the resources, roles and rules to define which privileges apply to each resource, and under what specific conditions. Next, you create policy rules that control which users and groups belong in these roles, and under what conditions. Later on in this section you are instructed to deploy this policy to the WebLogic Server 8.1 Security Service Module that you use to control access to your portal application resources.

WebLogic Enterprise Security provides two means for configuring portal application policy, the Administration Console and the Policy Import Tool. In this section you are directed to use the Administration Console to configure policy.

For more information on how to use the Administration Console to configure policy, see "Overview" in the *Policy Managers Guide* and "Policies" in the Console Help.

For instructions on how to use the Policy Import Tool to import policy files, see the Creating Policy Files section in the *Policy Managers Guide*.

This section covers the following topics:

- "Creating the Identity Directory and Users" on page 5-12

- "Configuring Resources and Privilege" on page 5-13

- "Configuring Resource Privileges" on page 5-17

- "Creating Roles and Role Policy" on page 5-17

- "Creating Policy Rules" on page 5-18

- "Policy for Visitor Entitlements to Portal Resources" on page 5-20

## Creating the Identity Directory and Users

This section describes how to use the Administration Console to create an identity directory, groups, and users for a portal application.

**Note:** This procedure uses `myusers` as the name of the Identity directory; however, you can use a different name.

To create the Identity directory and users:

1. In the left pane, click Identity. The Identity page displays the name of each directory available.

2. Click New. The Create Directory dialog box appears.

3. In the Name text box, type `myusers` and click OK. The `myusers` directory appears in the list of Identity directories.

4. Open the Identity folder, and click Identity.

5. The Identity page displays the name of each directory available.

6. From the Identity page, select `myusers`.

7. In the left pane, click Users. The Users page displays.

8. Click New. The Create User dialog box appears.

9. Create visitor users for your portal application.

## Configuring Resources and Privilege

This section describes how to use the Administration Console to define the portal application resources that you will protect using WebLogic Enterprise Security.

To configure resources, perform the following tasks:

- "Creating the Realm Resource" on page 5-13

- "Creating the Shared Resources" on page 5-14

- "Creating the Console Resources" on page 5-15

- "Creating the PortalApp Resources" on page 5-16

### Creating the Realm Resource

**Note:** myrealm is used is this procedure as the realm name because the WebLogic Portal sample portal application exists in the myrealm realm. You can choose any realm name for your portal application.

To create a realm resources, perform the following steps:

1. Open the Resources folder, and click Resources. The Resource page displays.

2. Select Policy and click New. The Create Resource dialog box appears.

3. In the Name text box, enter myrealm, select Binding from the Type drop-down list box, and click Ok. The myrealm resource appears under the Policy node.

4. Select the myrealm resource and click Configure. The Configure Resource dialog box appears.

5. From the Type drop-down list box, select Binding Application, check the Distribution Point and Allow Virtual Resources check boxes, and click Ok.

6. Refer to Table 5-2 and modify the configuration of the ASI Authorization provider as described there.

**Table 5-2  Portal Security Configuration Modifications**

| Security Provider | Configuration Settings |
| --- | --- |
| ASI Authorization Provider | On the Details tab, set the Application Deployment Parent to `//app/policy/myrealm` and click Apply. On the Bindings tab, from the Bind drop-down menu, select `//app/policy/myrealm`, and click Bind. |

## Creating the Shared Resources

Figure 5-2 shows the shared resources that you must create.

**Figure 5-2  shared Resources**
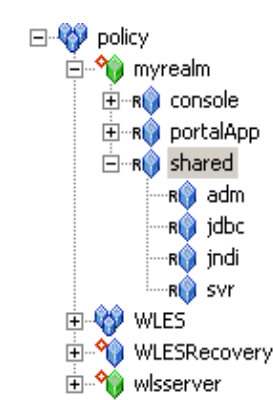


To create the `shared` resources, perform the following steps:

1. Select the `myrealm` resource and click New. The Create Resource dialog box appears.

2. In the Name text box, enter `shared`, and click Ok. The `shared` resource appears under `myrealm`.

3. Select the shared resource and click Configure. The Configure Resource dialog box appears.

4. Check the Allow Virtual Resources and click Ok.

5. Click the `shared` resource and click New. The Create Resource dialog box appears.

6. In the Name text box, enter `adm` and click Ok. The `adm` resource appears under the `shared` resource.

7. To configure the `jdbc`, `jndi`, and `svr` resources as shown in Figure 5-2, repeat steps 5 and 6 for each resource.

### Creating the Console Resources

Figure 5-3 shows the console resources that you must create.

**Figure 5-3  console Resources**



To create the `console` resources, perform the following steps:

1. Select the `myrealm` resource and click New. The Create Resource dialog box appears.

2. In the Name text box, enter `console`, and click Ok. The `console` resource appears under `myrealm`.

3. To create the `url`, `console`, `login`, and `bea_logo.gif` resources as shown in Figure 5-3, repeat steps 1 and 2 for each resource.

4. Select the `console` resource directly under `myrealm` and click Configure. The Configure Resource dialog box appears.

5. Check the Allow Virtual Resources and click Ok.

## Creating the PortalApp Resources

**Note:** This procedure uses `portalApp` as the name of the portal application resource because it is the name of the WebLogic Portal sample portal application. However, you should use the name of your portal application when creating the portal application resource.

Figure 5-4 shows the `portalApp` resources that you must create.

**Figure 5-4**  PortalApp Resources



To create the `portalApp` resources, perform the following steps:

1. Select the `myrealm` resource and click New. The Create Resource dialog box appears.

2. In the Name text box, enter `portalApp`, and click Ok. The `portalApp` resource appears under `myrealm`.

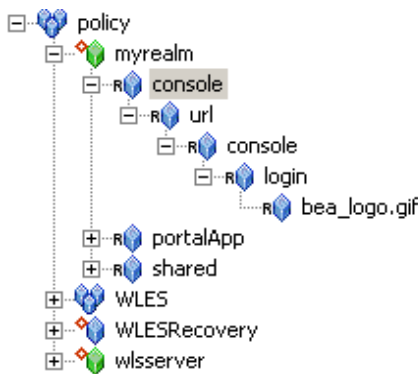3. Select the `portalApp` resource and click New. The Create Resource dialog box appears.

4. In the Name text box, enter `ejb` and click Ok. The `ejb` resource appears under the `portalApp` resource.

5. Select the `ejb` resource and click Configure. The Configure Resource dialog box appears.

6. Check the Allow Virtual Resources and click Ok.

7. To configure the `url` resource, repeat steps 5 and 6.

8. Select the `portalApp` resource and click New. The Create Resource dialog box appears.

9. In the Name text box, enter `wlp` and click Ok. The `wlp` resource appears under the `portalApp` resource. Do **not** configure the `wlp` resource to allow virtual resources.
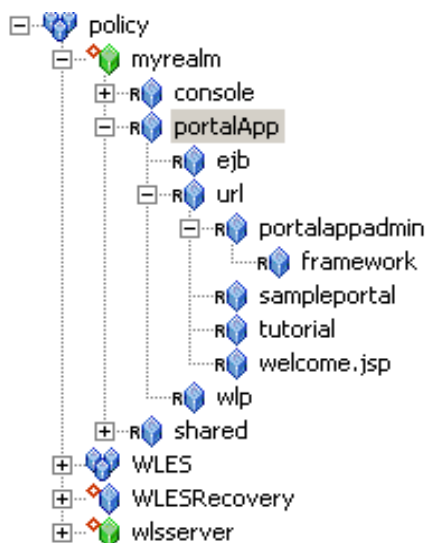
10. Select the `url` resource and click New. The Create Resource dialog box appears.

11. In the Name text box, enter `portalappadmin` and click Ok. The `portalappadmin` resource appears under the `url`.

12. Repeat steps 10 and 11 to create the remaining resources shown under the portalApp resource in Figure 5-4. Do **not** configure any of the remaining resources to allow virtual resources.

## Configuring Resource Privileges

This section describes how to use the Administration Console to define portal application privileges.

To create privileges, perform the following steps:

1. Open the Resources folder, and click Privileges. The existing privileges appear in the right pane.

2. Click New. The Create Privilege dialog box appears.

3. In the Privilege Name text box, enter `lookup` and click Ok. The `lookup` privilege appears in the list of privileges.

4. Click New. The Create Privilege dialog box appears.

5. In the Privilege Name text box, enter `reserve` and click Ok. The `reserve` privilege appears in the list of privileges.

6. Click OK.

## Creating Roles and Role Policy

This section describes how to use the Administration Console to create roles that will be used to control access to portal application resources.

Table 5-3 lists and describes the role policy that you have to create for the WebLogic Portal domain.

**Table 5-3 Roles and Role Policy**

| Policy | Description |
|---|---|
| `grant(//role/Everyone,`<br>`//app/policy/myrealm,`<br>`//sgrp/myusers/allusers/) if true;` | Creates the role policy necessary for the Everyone role to be used in the `myrealm` Identity directory.<br><br>**Note:** If you do not create the `Everyone` role policy correctly, none of the policy rules defined in Table 5-4 that use the `Everyone` role will work properly. |

To create roles and role policies, refer to Table 5-3 and perform the following steps:

**Caution:** If you do not create the `Everyone` role policy correctly, none of the policy rules defined in Table 5-4 that use the `Everyone` role will work properly.

1. Open the Roles folder, and click Roles. The Roles page displays.

2. Click New. The Create Role dialog box appears.

3. In the Role Name text box, enter `Everyone`, and click Ok. The role appears in the list of roles.

4. Select the `Everyone` role and click Assign. The Role Policy page appears.

5. Click New. The Create Role Policy dialog box appears.

6. Select the Roles tab and add the `Everyone` role to the Selected Roles column.

7. Select the Resources tab and add `myrealm` to the Selected Resources column.

8. Select the Identities tab, add the `allusers` user to the Selected Identities column, and click Ok. The role appears in the Role policy page.

## Creating Policy Rules

This section describes how to use the Administration Console to create policy rules to protect portal application resources.

Table 5-4 lists and describes the policy rules that you have to create for the WebLogic Portal domain to protect the sample portal application resources.

**Table 5-4  Policy Rules**

| Policy | Description |
|---|---|
| `grant(any, //app/policy/myrealm/shared/svr, //role/Admin) if true;`<br><br>`grant(any, //app/policy/myrealm/shared/adm, //role/Admin) if true;` | Administrative roles for booting the WebLogic Portal server and performing administrative tasks. |
| `grant(any, //app/policy/myrealm/portalApp/url/ sampleportal, //role/Everyone) if true;`<br><br>`grant(any, //app/policy/myrealm/portalApp/url/tutorial, //role/Everyone) if true;`<br><br>`grant(any, //app/policy/myrealm/portalApp/url/welcome.jsp, //role/Everyone) if true;` | Grants those in the role Everyone (includes the anonymous user) access to all of the tutorial and sample portal url resources. This policy creates Portal open by default orientation for these two sample portals. |
| `grant(GET, //app/policy/myrealm/portalApp/url/ portalappadmin/framework, //role/Everyone) if true;` | Allows unauthenticated users to access images used on the Administration Portal login page. |
| `grant(any, //app/policy/myrealm/portalApp/url/ portalappadmin, //role/ PortalSystemAdministrator) if true;` | Only those in the role PortalSystemAdministrator have access to the WebLogic Portal Administration url Portal resources |
| `grant(lookup, //app/policy/myrealm/shared/jndi, //role/Everyone) if true;` | Grants permission for those is the Everyone role to lookup JNDI resources. |
| `grant(reserve, //app/policy/myrealm/shared/jdbc, //role/Everyone) if true;` | Grants permission for those is the Everyone role to reserve JDBC resources. |
| `grant(any, //app/policy/myrealm/console, //role/Admin) if true;` | Grants permission for those in the Admin role to access the url resources of the WebLogic Server console. |

**Table 5-4  Policy Rules (Continued)**

| Policy | Description |
| --- | --- |
| `grant(GET, //app/policy/myrealm/console/url/console/login/ bea_logo.gif, //role/Everyone) if true;` | Grants permission for those in the Everyone role to get access to the bea logo gif image resource in the WebLogic Server console |
| `grant(any, //app/policy/myrealm/portalApp/ejb, //role/Everyone)` | Initially allows access to all EJB methods. |

Perform the following steps create the policy rules listed in Table 5-4.

1. Open the Policy folder, and click Policy. The Policy page displays.

2. Click New. The Create Rule dialog box appears.

3. Select the Grant radio button.

4. To add one or more privileges to a rule:

    a. Click the Privileges tab, select the `any` privilege from the Select Privileges from Group list box and add it to the Selected Privileges box.

    b. Click the Resources tab, select the `svr` resource from the Child Resource box and add it to the Selected Resources box.

        **Note:** If Table 5-4 lists multiple resources for a single privilege and role, you may add all of the resources at once.

    c. Click the Policy Subjects, select the `Admin` role from the Roles List box, add it to the Selected Policy Subjects box, and click Ok.

5. Repeat step 4 for each of the remaining policy rules listed in Table 5-4.

## Policy for Visitor Entitlements to Portal Resources

Visitor entitlements is a mechanism used by WebLogic Portal for determining who may access the resources in a portal application and what they may do with those resources. WebLogic Enterprise Security provides a means of defining robust role-based policy for portal resources. The resources that can be entitled within a portal application include:

- desktops

- books

- pages

- portlets

- look and feels

Table 5-5 shows the capabilities of each of these resources:

**Table 5-5** Capabilities According to Resource Type

| Resource Type | View | Minimize | Maximize | Edit | Remove |
|---|---|---|---|---|---|
| Desktop | x | | | | |
| Book | x | x | x | | |
| Page | x | | | | |
| Portlet | x | x | x | x | x |
| Look & Feel | x | | | | |

The capabilities listed in Table 5-5 are defined as follows:

- View—Determines whether or not the user can see the resource.

- Minimize/Maximize—Determines whether or not the user is able to minimize or maximize the portlet or book. This applies to books within a page, not to the primary book.

- Edit—Determines whether or not the user can edit the resource properties.

- Remove —Determines whether or not the user can remove the portlet from a page.

The following topics provide information on how to use WebLogic Enterprise Security to configure portal resources:

- "Configuring Policy for Desktops" on page 5-22

- "Configuring Policy for Books" on page 5-22

- "Configuring Policy for Pages" on page 5-23

- "Configuring Policy for Portlets" on page 5-23

- "Configuring Policy for Look and Feels" on page 5-24

## Configuring Policy for Desktops

A desktop is a view of the portal that the visitor accesses. There can be one or more desktops per portal, so the portal is effectively a container for the desktops. A Desktop is referenced as a resource in WebLogic Enterprise Security in the following manner:

```
//app/policy/myrealm/portalapp/wlp/sampleportal/com_bea_p13n/Desktop/samplePortal
```

where:

*myrealm* is the realm in which the portal application is installed

*portalapp* is the portal application directory

*sampleportal* is the name of the sample portal application

*samplePortal* is the label definition of the desktop.

If you define policy at the `samplePortal` level, you can control access at the `samplePortal` desktop level.

Table 5-6 shows a policy that would grant the `view` privilege to the `samplePortal` desktop for visitors in the SampleVisitor role.

**Table 5-6  SamplePortal Policy**

| Role Type | Privilege | Resource | Policy Subject |
|-----------|-----------|----------|----------------|
| Grant | view | /myrealm/portalapp/wlp/sampleportal/com_bea_p13n /Desktop/samplePortal | SampleVisitor role |

## Configuring Policy for Books

A book is a collection of pages. A book is referenced as a resource in WebLogic Enterprise Security in the following manner:

```
//app/policy/myrealm/portalapp/wlp/sampleportal/com_bea_p13n/Book/book_1
```

where `book_1` is the label definition of the book.

If you define policy at the `book_1` level, you can control access at the book_1 book level.

Table 5-7 shows a policy that would grant the `view` privilege to the book_1 book for visitors in the SampleVisitor role.

**Table 5-7  Book_1 Policy**

| Role Type | Privilege | Resource | Policy Subject |
|---|---|---|---|
| Grant | view | `/myrealm/portalapp/wlp/sampleportal/com_bea_p13n` `/Book/book_1` | SampleVisitor role |

### Configuring Policy for Pages

A page is the primary holder of individual portal elements such as portlets. A page is referenced as a resource in WebLogic Enterprise Security in the following manner:

`//app/policy/myrealm/portalapp/wlp/sampleportal/com_bea_p13n/Page/page_2`

where `page_2` is the label definition of the page.

If you define policy at the `page_2` level, you can control access at the page_2 page level.

Table 5-8 shows a policy that would grant the `view` privilege to the page_2 page for visitors in the SampleVisitor role.

**Table 5-8  Page_2 Policy**

| Role Type | Privilege | Resource | Policy Subject |
|---|---|---|---|
| Grant | view | `/myrealm/portalapp/wlp/sampleportal/` `com_bea_p13n/Page/page_2` | SampleVisitor role |

### Configuring Policy for Portlets

Portlets are the visible components that act as the interface to applications and content. A portlet is referenced as a resource in WebLogic Enterprise Security in the following manner:

`//app/policy/myrealm/portalapp/wlp/sampleportal/com_bea_p13n/Portlet/portlet_login1`

where `portlet_login` is the label definition of the portlet.

If you define policy at the `portlet_login1`level, you can control access at the `portlet_login1` Portlet level.

Table 5-9 shows a policy that would grant the `view` privilege to the `portlet_login1` Portlet for visitors in the SampleVisitor role.

**Table 5-9  Portlet_login1 Policy**

| Role Type | Privilege | Resource | Policy Subject |
|---|---|---|---|
| Grant | view | /myrealm/portalapp/wlp/sampleportal/com_bea_p13n /Portlet/*portlet_login1* | SampleVisitor role |

### Configuring Policy for Look and Feels

A Look and Feel is a selectable combination of skins and skeletons that determine the physical appearance of a portal desktop. A Look and Feel is referenced as a resource in WebLogic Enterprise Security in the following manner:

```
//app/policy/myrealm/portalapp/wlp/sampleportal/com_bea_p13n/LookAndFeel/textL
ookAndFeel
```

where `textLookAndFeel` is the label definition of the Look and Feel.

If you define policy at the `textLookAndFeel`level, you can control access at the `textLookAndFeel` level.

Table 5-10 shows a policy that would grant the `view` privilege to the `textLookAndFeel` Look and Feel visitors in the SampleVisitor role.

**Table 5-10  Portlet_login1 Policy**

| Role Type | Privilege | Resource | Policy Subject |
|---|---|---|---|
| Grant | view | /myrealm/portalapp/wlp/sampleportal/com_bea_p13n /LookAndFeel/*textLookAndFeel* | SampleVisitor role |

### Defining Policy for Portlets using Instance ID

Portlets have a unique instance ID that allows for granular policy definition outside the standard hierarchy of the Desktop->Book->Page->Portlet. To use this in WebLogic Enterprise Security, add a condition statement in the portlet rule that adds the portlet instance ID. For example:

```
grant( [//priv/maximized,//priv/minimized,//priv/view],
//app/policy/myrealm/portalapp/wlp/sampleportal/com_bea_p13n/Portlet
/ortlet_login1, //role/Operator) if instanceid = "portlet_login1";
```

Table 5-11 shows a policy that would grant the `view` privilege to the *portlet_login1* Portlet for visitors in the Operator role.

**Table 5-11  Portlet_login1 Policy Using Instance ID**

| Role Type | Privilege | Resource | Policy Subject | Condition |
|---|---|---|---|---|
| Grant | view | /myrealm/portalapp/wlp/sampleportal/ com_bea_p13n/Portlet/*portlet_login1* | Operator role | if instanceid = "*portlet_login1*"; |

# Discovering Portal Application Resources

When developing policy for use with a Security Service Module, you can use the discovery mode feature of to WebLogic Enterprise Security Administration Application to help build your policy. Instructions for using discovery mode are provided in the "Resource Discovery" section in the *Policy Managers Guide*.

# Distributing Policy and Security Configuration

Distribute the policy and security configuration to the WebLogic Server 8.1 Security Service Module.

For information on how to distribute policy and security configuration, see "Deployment" in the *Administration Application Guide* and the Console Help. Be sure to verify the results of your distribution.

# Starting the WebLogic Portal Server

To start a WebLogic Portal server, perform the following steps:

1. Open a shell (command prompt) on the machine on which you created the portal domain.

2. Change to the portal domain directory:
   `BEA_HOME\user_projects\domains\portalDomain`.

3. Run the following script:

   On Windows: `startWebLogic.cmd`

   On UNIX: `startWeblogic.sh`

# Configuring Portal Administration to Use the WebLogic Authenticator

To use the WebLogic Authentication provider to manage administrative users for portal administration, perform the following steps:

1. Within the Portal Administration console, go to `Service Administration`.

2. Select `Authentication Hierarchy Service`.

3. Add `WebLogicAuthenticator` to the Authentication Providers to Build list.

# Using Portal Administration Tools to Create a Portal Desktop

Before you can use WebLogic Enterprise Security to control access to a portal desktop, you must use WebLogic Portal Administration Tools to create a portal desktop.

To create a portal desktops, perform the following steps:

1. To have full control of the definition and instance labels, use WebLogic Workshop 8.1 to create the initial portal application.

2. Using the initial portal application as a template, use the Portal Administration Tools to create the portal desktops.

For instructions on using Portal Administration Tools to create portal desktops, see "Create a Desktop" in the WebLogic Administration Portal Online Help.

To create a portal desktop for the sample portal application using the Portal Administration Tools, perform the following steps:

1. Open a browser and point it to: `http://localhost:7001/testportalappAdmin`. The Sign In page appears.

2. Enter Username: `portaladmin`, Password: `portaladmin`, and click Sign In. The Portal Resources navigation tree appears.

3. Right click on Portals and select New Portal. The Portal Resource page appears.

4. Enter a Name for this Portal, for example: `myportal`, enter a partial URL for this Portal, for example: `myportal`, and click Save. The Available Portals list appears.

5. Select the `myportal` link from the list. The Editing Portal: `myportal` page appears.

6. Click Create New Desktop. The New Desktop Properties dialog appears.

7. Enter Title, for example: `mydesktop` and a Partial URL, for example: `mydesktop`.

8. Choose a Template:Choose: `testportalweb/test.portal` and click Create New Desktop. Desktops contained in Portal `myportal` list appear.

9. Select `mydesktop` from the list. The Editing Desktop: `mydesktop` page appears.

10. Select View Desktop. `mydesktop` is displayed in a new browser window. Verify that the desktop contains the sample portal application.

11. Close the mydesktop browser and Logout. The Sign In page appears.

12. Exit the Portal Administration Tool by closing its browser.

## Accessing the Portal Application

To access a portal application running on a portal server, open a browser and point it to the desktop URL. For example, if you set up the desktop for the sample portal application as described in "Using Portal Administration Tools to Create a Portal Desktop" on page 5-26, you can access the sample portal application using the following URL:

`http://<host_name>:7001/sampleportal/appmanager/myportal/mydesktop`

where `<host_name>` is the machine on which portal application is running.

# Uninstalling

The following sections describe how to uninstall the WebLogic Server 8.1 Server Security Service Module (SSM) and the Service Control Manager (SCM):

- "Uninstalling the WebLogic Server 8.1 SSM on Windows" on page 6-2

- "Uninstalling the WebLogic Server 8.1 SSM on Solaris or Linux" on page 6-3

- "Uninstalling the SCM on Windows" on page 6-4

- "Uninstalling the SCM on Solaris or Linux" on page 6-6

# Uninstalling the WebLogic Server 8.1 SSM on Windows

To uninstall the Security Service Module from a Windows platform, do the following:

1. Log in to the machine as Administrator.

2. Stop the Authorization and Role Mapping Engine (ARME) for the Security Service Module that you are uninstalling: ARME.admin.*hostname.*

3. Stop the Service Control Manager (SCM).

4. Click Start, select Programs>BEA WebLogic Enterprise Security>Security Service Module>Uninstall Combo Security Service Manager.

   The Uninstall Welcome window appears.

5. Click Next.

   The Choose Components window appears.

6. Select the WLES SSMs to uninstall and click Next.

   The BEA Uninstaller window appears and the uninstall process begins.

**Note:** When you uninstall a Security Service Module, if it is the only remaining WebLogic Enterprise Security product on the machine, you are given the option of uninstalling the SCM. If you want to uninstall the Service Control Manager, check the Uninstall SCM box and click Next.

As the uninstall process runs, a checklist is displayed, listing the uninstallation tasks as they complete. After the product is removed, the "uninstall complete" message appears.

7. Click Done.

8. Open Windows Explorer and delete the directory for WebLogic Server 8.1 SSM. If you uninstalled all SSMs, delete the top-level SSM directory. The default top-level directory is *BEA_HOME*\wles42-ssm.

   **Note:** If you checked the Uninstall SCM box during the uninstall, you also uninstalled the Service Control Manager, so delete the SCM directory as well. The default SCM directory is *BEA_HOME*\wles42-scm.

9. To delete shortcuts for each SSM that you uninstalled from the Start>Programs>WebLogic Enterprise Security menu, right click on each shortcut you want to delete, and then choose delete.

**Note:** If you do not delete the menu shortcuts and you reinstall the WebLogic Server 8.1 Server Security Service Module, duplicate product names appear in the Programs menu.

10. If you have uninstalled all of the WebLogic Enterprise Security product software from your computer, and you do not know the passwords for the related asi users and groups, you must delete those users and groups. To delete the asi users and groups, perform the following steps:

  a. Select Start>Settings>Control Panel, click on Administrative Tools, click on Computer Management, and expand Local Users and Groups.

  b. Select Users and delete the Administration Application user (asiadmin by default) and the Service Control Manager user (scmuser by default)

  c. Select Groups and delete the WLES administrators group (asiadgrp) and the WLES users group (asiusers)

11. You have successfully removed the SSM product software from your computer.

# Uninstalling the WebLogic Server 8.1 SSM on Solaris or Linux

To run the graphical mode uninstallation program, your console must support a Java-based GUI. If the uninstallation program determines that your system cannot support a Java-based GUI, the uninstallation program automatically starts in console mode.

To uninstall the Security Service Module from a Solaris platform:

1. Log in to the machine as root (or su root).

2. Stop the Authorization and Role Mapping Engine (ARME) for the Security Service Module that you are uninstalling: `ARME.admin.`*`hostname.`*

3. Stop the Service Control Manager (SCM).

4. Open a command shell and go to the directory where you installed the product, for example:

   *`BEA_HOME`*`/wles42-ssm/uninstall`

   where:

   *`BEA_HOME`*`/wles42-ssm` represents the directory in which you installed the product.

5. At the command prompt, type `uninstall.sh`.

   The BEA Uninstaller window appears and the uninstall process begins.

> **Note:** If your system supports a graphical user interface, the uninstallation program starts in graphical mode. If your system does not support a graphical user interface, the uninstallation program starts in console mode.

6. Respond to the prompts to uninstall the product.

**Note:** When you uninstall a SSM, if it is the only remaining WebLogic Enterprise Security product on the machine, you are given the option of uninstalling the SCM. If you want to uninstall the Service Control Manager, check the Uninstall SCM box and click Next.

As the uninstall process runs, a checklist is displayed, listing the uninstallation tasks as they complete. After the product is removed, the "uninstall complete" message appears.

7. If you system supports a graphical user interface, click Done.

You have successfully removed Security Service Module from your computer.

**Note:** If you elected to uninstall the SCM, it is also uninstalled.

# Uninstalling the SCM on Windows

**Note:** If you elected to uninstall the Service Control Manager (SCM) when you uninstalled the Security Service Module (SSM), this task is not necessary.

To uninstall the Service Control Manager from a Windows platform, do the following:

**Note:** Before uninstalling the Service Control Manager, you must remove all Security Service Modules from your machine. If a Security Service Module is still installed on your machine, the uninstall program does not allow you to uninstall the Service Control Manager.

1. Log in to the machine as Administrator.

2. Stop the Service Control Manager (SCM): `WLES Service Control Manager`.

3. Click Start, select Programs>BEA WebLogic Enterprise Security>Service Control Manager>Uninstall Service Control Manager.

   The Uninstall Welcome window appears.

4. Click Next.

   The BEA Uninstaller window appears and the uninstall process begins.

   As the uninstall process runs, a checklist is displayed, listing the uninstallation tasks as they complete. After the product is removed, the "uninstall complete" message appears.

5. Click Done.

You have successfully removed the Service Control Manager from your computer.

**Note:** The Uninstall program does not completely remove the WebLogic Enterprise Security software from your machine. Remnants of the software installation remain, such as product directories, log files, asi users and groups, and program menu shortcuts from your system. You can remove these manually. For removal instructions, see "Additional Steps for Uninstalling the SCM on Windows" on page 6-5

## Additional Steps for Uninstalling the SCM on Windows

After the uninstall completes, you may notice that the product directory and various log files remain in the product directory, as well as program menu shortcuts and asi users and groups. You may remove these manually or you may want to keep them.

To remove these items, perform the following steps:

1. Delete the product directories for each product that you uninstalled. Doing this removes all instance files that remain, unless you created them outside of the product directory. In this case, you need to go to the directory where you installed the instance and delete the instance directory.

2. To delete shortcuts for each product that you uninstalled from the Start>Programs>WebLogic Enterprise Security menu, right click on each shortcut you want to delete, and then choose delete.

   **Note:** If you do not delete the menu shortcuts and you reinstall the product, duplicate product names appear in the Programs menu.

3. To delete asi users and groups, open the Control Panel>Administrative Tools>Computer Management>Local Users and Groups window and delete the following users and groups:

   – The Administration Application user (asiadmin by default)

   – The Service Control Manager user (scmuser by default)

   – The WLES administrators group (asiadgrp)

   – The WLES users group (asiusers)

   If you know the passwords of asiadmin and scmuser (passwords typed in during a previous install, rather than accepting the defaults), then you may leave these users in place and enter those passwords when you reinstall the product.

# Uninstalling the SCM on Solaris or Linux

**Note:** If you elected to uninstall the Service Control Manager when you uninstalled the Security Service Module, this task is not necessary.

To run the graphical mode uninstallation program, your console must support a Java-based GUI. If the uninstallation program determines that your system cannot support a Java-based GUI, the uninstallation program automatically starts in console mode.

To uninstall the Service Control Manager from a Unix platform:

**Note:** Before uninstalling the Service Control Manager, you must remove all Security Service Modules from your machine. If a Security Service Module is still installed on your machine, the uninstall program does not permit you to uninstall the Service Control Manager.

1. Log in to the machine as root (or su root).

2. Stop the Service Control Manager (SCM): `WLES Service Control Manager`.

3. Open a command shell and go to the directory where you installed the Service Control Manager, then go to the uninstall directory. For example:

   *BEA_HOME*/wles42-scm/uninstall

   where:

   *BEA_HOME*/wles42-scm/ represents the directory in which you installed Service Control Manager component.

4. At the command prompt, type `uninstall.sh`.

   The BEA Uninstaller window appears and the uninstall process begins.

**Note:** If your system supports a graphical user interface, the uninstallation program starts in graphical mode. If your system does not support a graphical user interface, the uninstallation program starts in console mode.

5. Respond to the prompts to uninstall the product.

6. If you system supports a graphical user interface, click Done.

You have successfully removed the Security Service Module from your computer.

Uninstalling