Deployment Example 1: Access Manager 7.1 Load Balancing, Distributed Authentication UI, and Session Failover

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

Part No: 820–2278–11
November 2007

Products covered by and information contained in this publication are controlled by U.S. Export Control laws and may be subject to the export or import laws in other countries. Nuclear, missile, chemical or biological weapons or nuclear maritime end uses or end users, whether direct or indirect, are strictly prohibited. Export or reexport to countries subject to U.S. embargo or to entities identified on U.S. export exclusion lists, including, but not limited to, the denied persons and specially designated nationals lists is strictly prohibited.

DOCUMENTATION IS PROVIDED “AS IS” AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

Les produits qui font l'objet de cette publication et les informations qu'il contient sont régis par la legislation américaine en matière de contrôle des exportations et peuvent être soumis au droit d'autres pays dans le domaine des exportations et importations. Les utilisations finales, ou utilisateurs finaux, pour des armes nucléaires, des missiles, des armes chimiques ou biologiques ou pour le nucléaire maritime, directement ou indirectement, sont strictement interdites. Les exportations ou réexportations vers des pays sous embargo des Etats-Unis, ou vers des entités figurant sur les listes d'exclusion d'exportation américaines, y compris, mais de manière non exclusive, la liste de personnes qui font objet d'un ordre de ne pas participer, d'une façon directe ou indirecte, aux exportations des produits ou des services qui sont régis par la legislation américaine en matière de contrôle des exportations et la liste de ressortissants spécifiquement designés, sont rigoureusement interdites.

LA DOCUMENTATION EST FOURNIE "EN L'ETAT" ET TOUTES AUTRES CONDITIONS, DECLARATIONS ET GARANTIES EXPRESSES OU TACITES SONT FORMELLEMENT EXCLUES, DANS LA MESURE AUTORISEE PAR LA LOI APPLICABLE, Y COMPRIS NOTAMMENT TOUTE GARANTIE IMPLICITE RELATIVE A LA QUALITE MARCHANDE, A L'APTITUDE A UNE UTILISATION PARTICULIERE OU A L'ABSENCE DE CONTREFACON.

Contents

Preface

Sun Java™ System Access Manager provides a comprehensive solution for protecting network resources that integrates authentication and authorization services, policy agents, and identity federation. This Preface to the *Deployment Example 1: Access Manager 7.1 Load Balancing, Distributed Authentication UI, and Session Failover* contains the following sections:

- "About This Guide" on page 9
- "Access Manager Core Documentation" on page 9
- "Sun Java System Product Documentation" on page 11
- "Typographical Conventions" on page 11

About This Guide

Deployment Example 1: Access Manager 7.1 Load Balancing, Distributed Authentication UI, and Session Failover provides instructions for building a Sun Java System Access Manager 7.1 solution for authentication, authorization and access control. The instructions in this guide were used to build, deploy and test this Deployment Example in a lab facility. You'll obtain the best results if you perform the tasks in the exact sequence in which they are presented. Use the Table of Contents as a master task list. Tasks are numbered for your convenience.

Caution – If you do plan to deviate from the task sequence or details described in this guide, you should refer to the relevant product documentation for information on differences in platforms, software versions or other requirement constraints.

Access Manager Core Documentation

The Access Manager core documentation set also contains the following titles:

- The *Sun Java System Access Manager 7.1 Release Notes* will be available online after the product is released. It gathers an assortment of last-minute information, including a description of what is new in this current release, known problems and limitations, installation notes, and how to report issues with the software or the documentation.
- The *Sun Java System Access Manager 7.1 Technical Overview* provides an overview of how Access Manager components work together to protect enterprise assets and web-based applications. It also explains basic Access Manager concepts and terminology.

- The *Sun Java System Access Manager 7.1 Deployment Planning Guide* provides planning and deployment solutions for Sun Java System Access Manager based on the solution life cycle
- The *Sun Java System Access Manager 7.1 Postinstallation Guide* provides information for configuring Access Manager after running the Java ES installer.
- The *Sun Java System Access Manager 7.1 Performance Tuning and Troubleshooting Guide* provides information on how to tune Access Manager and its related components for optimal performance.
- The *Sun Java System Access Manager 7.1 Administration Guide* describes various administrative tasks such as Realms Management, Policy Management, Authentication and Directory Management. Most of the tasks described in this book are performed through the Access Manager console as well as through the command line utilities.
- The *Sun Java System Access Manager 7.1 Administration Reference* is a look-up guide containing information about the command line interfaces, configuration attributes, Access Manager files, and error codes.
- The *Sun Java System Access Manager 7.1 Federation and SAML Administration Guide* provides information about the Federation module based on the Liberty Alliance Project specifications and the use of the Security Assertion Markup Language (SAML). It includes information on the integrated services based on these specifications, instructions for enabling a web services environment, and summaries of the application programming interface (API) for extending the framework.
- The *Sun Java System Access Manager 7.1 Developer's Guide* offers information on how to customize Access Manager and integrate its functionality into an organization's current technical infrastructure. It also contains details about the programmatic aspects of the product and its API.
- The *Sun Java System Access Manager 7.1 C API Reference* provides summaries of data types, structures, and functions that make up the public Access Manager C APIs.
- The *Sun Java System Access Manager 7.1 Java API Reference* provides information about the implementation of Java packages in Access Manager.
- The *Sun Java System Access Manager Policy Agent 2.2 User's Guide* provides an overview of the policy functionality and the policy agents available for Access Manager.

Updates to the *Release Notes* and links to modifications of the core documentation can be found on the Access Manager page at the Sun Java Enterprise System documentation web site. Updated documents will be marked with a revision date.

Sun Java System Product Documentation

Useful information might also be found in the documentation for the following products:

- Sun Java System Directory Server Enterprise Edition 6.0
- Sun Java System Web Server 7.0
- Sun Java System Access Manager Policy Agent 2.2

Typographical Conventions

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

TABLE P–1 Typographic Conventions

Typeface	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`

`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`**
		`Password:`

aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. A *cache* is a copy that is stored locally. Do *not* save the file. **Note:** Some emphasized items appear bold online.

PART I

About This Deployment Example

This first part of the *Deployment Example 1: Access Manager 7.1 Load Balancing, Distributed Authentication UI, and Session Failover* provides introductory material and an overview of the Sun Java™ System Access Manager 7.1 deployment solution that incorporates load-balancing, distributed authentication, policy agents, and protected resources. It contains the following chapters:

- Chapter 1, "Components and Features"
- Chapter 2, "Technical Overview"

CHAPTER 1

Components and Features

This chapter contains introductory information on the deployment example. It includes the following sections:

- "1.1 System Components and Architecture" on page 15
- "1.2 Key Features of Deployment" on page 19
- "1.3 Sequence of Interactions" on page 20

1.1 System Components and Architecture

The following components comprise the system environment of the deployment example. Figure 1–1 follows the list and illustrates the components as they will be situated when the deployment is complete.

Sun Java System Access Manager

Two Access Manager servers provide core Access Manager functionality. Both servers share the same configuration data, accessed through a single load balancer deployed in front of two instances of Directory Server.

Sun Java System Directory Server

Two instances of Directory Server provide storage for the Access Manager configuration data and user entries. Configuration data includes information about Access Manager services, realms, policies, and more. User entries will be created and used for testing the deployment. Both Directory Server instances are master replicas that engage in multi-master replication (MMR). MMR allows data to be synchronized in real time between two directories, providing high availability to the Access Manager layer.

Note – No Directory Server Administration Console is installed with the bits downloaded for this deployment example. The command line is used for all Directory Server configurations.

Protected Resources

The protected resources are host machines that contain content for which you want to restrict access. Towards this end, web servers, application servers, and policy agents will be installed. For example, a Human Resources Department might use Sun Java System Web Server to host content and applications. Some of the hosted information must be made available to benefits administration vendors (such as health care providers or stock administrators) that need to access employee information in order to coordinate benefits. These external vendors access the protected resources through an external-facing load balancer. Other information must be restricted to internal Human Resources administrators who access the protected resources through an internal-facing load balancer.

Sun Java System Access Manager Policy Agents 2.2

Both Web Policy Agents and J2EE Policy Agents are used to restrict access to content or applications hosted on the protected resources. The policy agents intercept HTTP requests from external users and redirect the request to Access Manager for authentication. The agent communicates with the Access Manager servers through an internal-facing load balancer.

Distributed Authentication User Interface

The Distributed Authentication User Interface is a component of Access Manager that provides a thin presentation layer for user authentication. During user authentication, a Distributed Authentication Module retrieves the user's credentials and passes them to Access Manager for verification. Thus, the user does not have direct network access to any Access Manager servers.

Sun Java System Message Queue Broker Cluster

Access Manager uses two Message Queue broker instances to form a cluster for distributing client connections and message delivery. The Berkeley Database by Sleepycat Software, Inc. is the session store database. When an Access Manager server goes down and session failover is enabled, the user's session token can be retrieved from one of the Message Queues by the available Access Manager server. This ensures that the user remains continuously authenticated, allowing access to the protected resources without having to reauthenticate.

Load Balancers

The load balancer hardware and software used for this deployment is BIG-IP® manufactured by F5 Networks. They are deployed as follows:

Distributed Authentication User Interface Load Balancer. This external-facing load balancer exposes the remote, web-based Distributed Authentication User Interface for user authentication and self-registration.

Access Manager Load Balancer. This internal-facing load balancer exposes the web-based Access Manager administration console to internal administrators. Alternatively, internal administrators can bypass this load balancer and log in directly to an Access Manager administration console.

Directory Server Load Balancer. The load balancers in front of the Directory Server instances provide round-robin load balancing for Directory Server access, and detects individual Directory Server failures and recovery. Failed servers are taken out of the load balancer list. The load balancer also provides a single virtual Directory Server host name for the Access Manager servers.

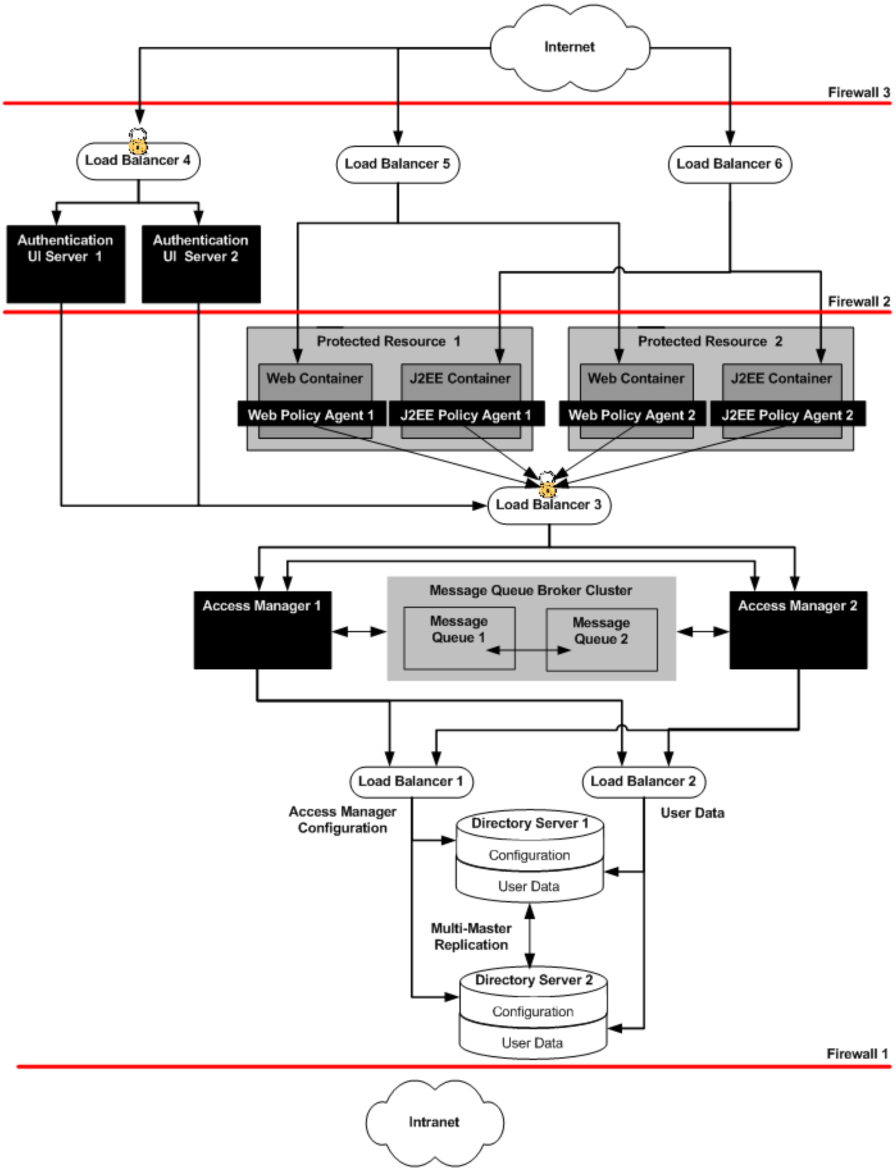


FIGURE 1–1 System Architecture

Note – Actual firewalls are not set up in this example deployment although they are referred to in this illustration. For information on specific firewall rules, see "2.5 Firewall Rules" on page 30.

1.2 Key Features of Deployment

- All components (including installations of Access Manager and Directory Server, the Distributed Authentication User Interface, and policy agents) are redundant to achieve high availability.
- All components use ZIP-based installation.
- All components use load-balancing for session failover and high performance.
- Each Directory Server contains two instances:
  1. `am-config` stores Access Manager configuration data.
  2. `am-users` serves as the LDAP v3 data store for user entries.
- The environment includes one service access interface for external users and agents, and a separate service access interface for internal administrators.
- Access Manager servers are configured to run as non-root users.
- The environment is configured for system failover capability, ensuring that when one Access Manager server goes down, requests are redirected to the second Access Manager server.

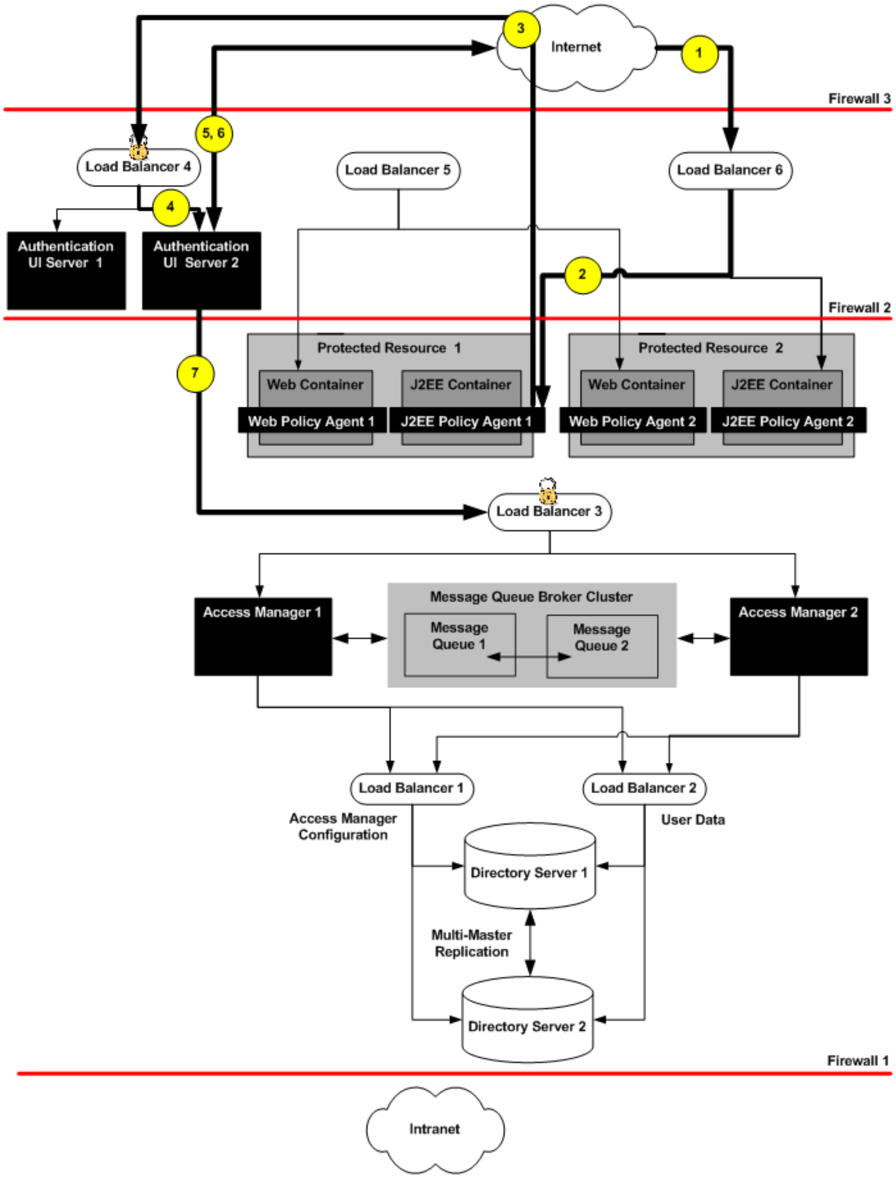
⚠ **Caution –** It is important to note that system failover, by itself, does not ensure Access Manager session failover. Session failover is configured separately.

- The environment is configured for session failover capability. Session failover ensures that when the Access Manager server *where the user's session was created* goes down, the user's session token can still be retrieved from a backend session database. Thus, the user is continuously authenticated, and does not have to log into the system again unless the session is invalidated as a result of logout or session expiration.
- Communications to the load balancer for the Access Manager servers and to the load balancer for the Distributed Authentication User Interface are in Secure Sockets Layer (SSL). SSL is then terminated and communications between the load balancers and their respective components is non-SSL.
- Policy agents are configured with a unique agent profile to authenticate to Access Manager.
- The Distributed Authentication User Interface uses a custom user profile to authenticate to Access Manager instead of the default `amadmin` or `UrlAccessAgent`.

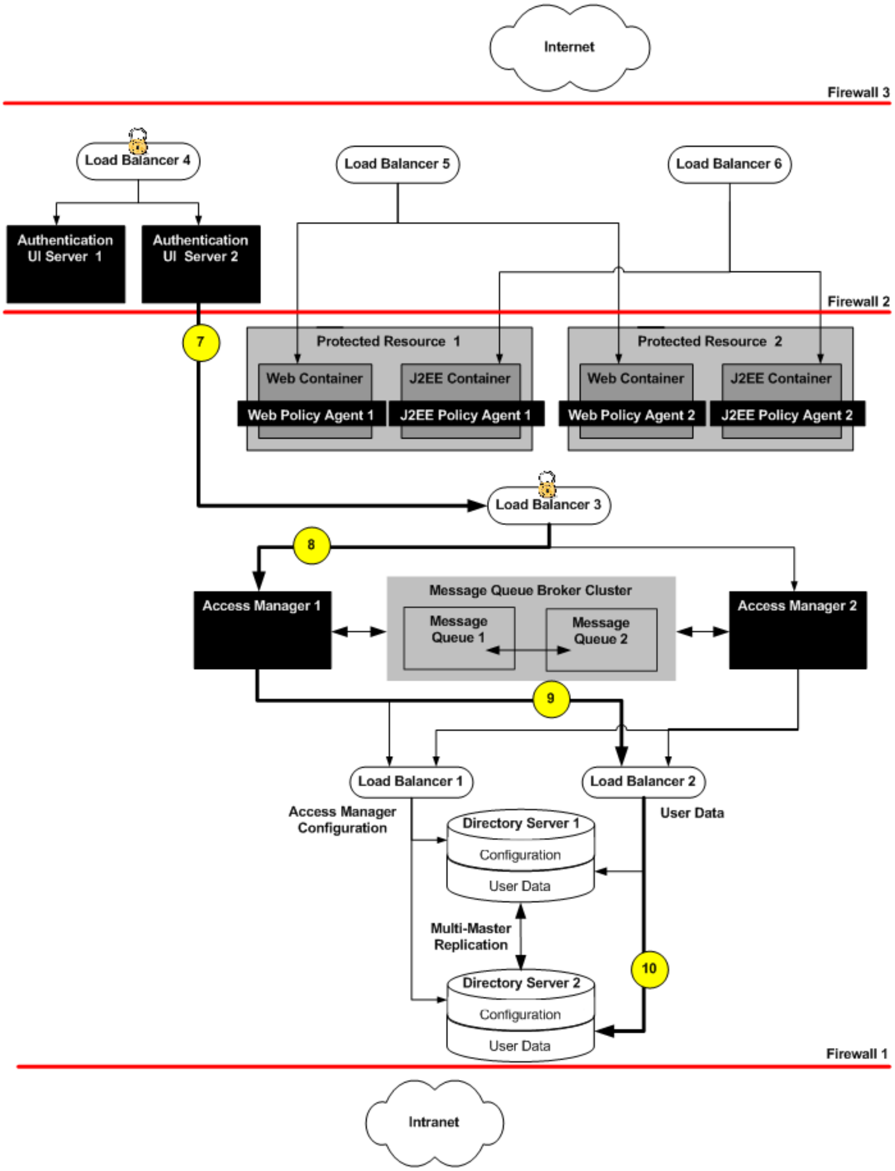
1.3 Sequence of Interactions

The following sequence describes the interactions between the various components in this deployment example. The interactions are illustrated and the numbered steps correspond to the numbers in the diagrams.

1. A user attempts to access a J2EE application hosted by Protected Resource 1 and Protected Resource 2.
2. Load Balancer 6 directs the user to Protected Resource 1.
3. The J2EE Policy Agent intercepts the request and checks for an Access Manager cookie. In this scenario, no cookie is found and the request is returned to the browser which then redirects it to Load Balancer 4, the load balancer for the Distributed Authentication User Interface.
4. Load Balancer 4 routes the user request to Distributed Authentication User Interface 2.
5. Distributed Authentication User Interface 2 displays a login page to the user.
6. The user enters credentials on the login page and they are returned to Distributed Authentication User Interface 2.

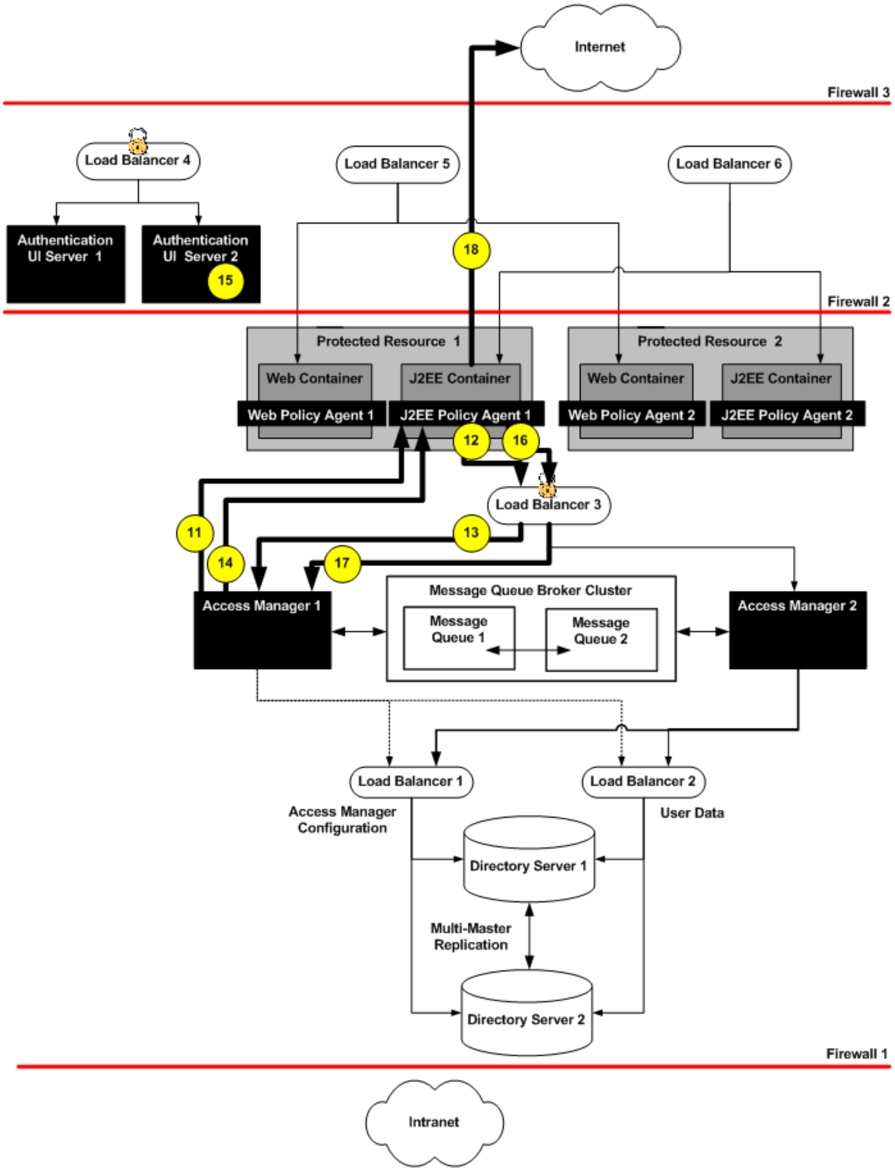


7. Distributed Authentication User Interface 2 passes the credentials to Load Balancer 3.
8. Load Balancer 3 routes the credentials to Access Manager 1 for validation.
9. Access Manager 1 sends a request for validation to Load Balancer 2 which specifically handles Directory Server requests for user data.
10. Load Balancer 2 routes the request to Directory Server 2 where validation takes place.



11. After successful authentication, Access Manager 1 sends the response back to the J2EE Policy Agent. The J2EE Policy Agent receives the request and checks for the Access Manger cookie.

12. When a cookie is found, the J2EE Policy Agent sends a session validation request to the Access Manager Load Balancer 3.
13. The Access Manager load balancer forwards the request to the Access Manager 1 where the session originated. Cookie-based persistency enables proper routing.
14. Access Manager 1 sends a response back to the J2EE Policy Agent.
15. If the session is not valid, the J2EE Policy Agent would redirect the user to the Distributed Authentication User Interface.
16. If the session is valid, the J2EE Policy Agent receives the response back and sends a policy request to the Access Manager Load Balancer 3.
17. The policy request is directed to Access Manager 1 which conducts the policy evaluation.
18. Based on the policy evaluation, the J2EE Policy Agent either allows access to the resource or denies access to the resource. In this scenario, the user is allowed access to the Application Server.


Internet
Firewall 3
Load Balancer 4
Load Balancer 5
Load Balancer 6
Authentication UI Server 1
Authentication UI Server 2
15
18
Firewall 2
Protected Resource 1
Web Container
J2EE Container
Web Policy Agent 1
J2EE Policy Agent 1
Protected Resource 2
Web Container
J2EE Container
Web Policy Agent 2
J2EE Policy Agent 2
12
16
Load Balancer 3
11
13
14
17
Access Manager 1
Message Queue Broker Cluster
Message Queue 1
Message Queue 2
Access Manager 2
Load Balancer 1
Load Balancer 2
Access Manager Configuration
User Data
Directory Server 1
Multi-Master Replication
Directory Server 2
Firewall 1
Intranet

CHAPTER 2

Technical Overview

This chapter contains technical information regarding the machines, software, and other components used in this deployment example. It contains the following sections:

2.1 Host Machines

The following table lists the attributes of the physical host machines used for this deployment example.

TABLE 2–1 Physical Machines and Operating Systems

Host Machine	Architecture	Operating System
DirectoryServer–1	x86	Solaris 10
DirectoryServer–2	x86	Solaris 10
AccessManager–1	SPARC	Solaris 10
AccessManager–2	SPARC	Solaris 10
MessageQueue–1	SPARC	Solaris 10
MessageQueue–2	SPARC	Solaris 10
AuthenticationUI–1	x86	Solaris 10

TABLE 2–1 Physical Machines and Operating Systems *(Continued)*

Host Machine	Architecture	Operating System
AuthenticationUI–2	x86	Solaris 10
ProtectedResource–1	SPARC	Solaris 10
ProtectedResource–2	SPARC	Solaris 10

2.2 Software

The following table lists the software used in this deployment example.

TABLE 2–2 Software Versions and Download Locations

Product	Version	Download Location
Sun Java^{TM} System Access Manager	7.1	`http://www.sun.com/download/`
Sun Java System Web Server	7.0	`http://www.sun.com/download/`
Sun Java System Directory Server	6.0	`http://www.sun.com/download/`
BEA Weblogic Server	9.2	`http://www.bea.com`
Web Policy Agent (for Sun Java System Web Server)	2.2	`http://www.sun.com/download/`
J2EE Policy Agent (for BEA Weblogic Server)	2.2	`http://www.sun.com/download/`
Java (for Access Manager and policy agents)	1.5.0_09	`http://www.java.com/en/`
BIG-IP Load Balancer		`http://www.f5.com`

2.3 Main Service URLs for Deployment Components

The following table summarizes the main service URLs for the components used in this deployment example. For detailed configuration information, see Part III.

TABLE 2–3 Components and Main Service URLs

Components	Main Service URL
Directory Server Instances and Load Balancers	

TABLE 2–3 Components and Main Service URLs *(Continued)*

Components	Main Service URL
Directory Server 1	`ldap://DirectoryServer-1.example.com:1389` (for Access Manager configuration data) `ldap://DirectoryServer-1.example.com:1489` (for user data)
Directory Server 2	`ldap://DirectoryServer-2.example.com:1389` (for Access Manager configuration data) `ldap://DirectoryServer-2.example.com:1489` (for user data)
Load Balancer 1	`http://LoadBalancer-1.example.com:389` (for Access Manager configuration data)
Load Balancer 2	`http://LoadBalancer-2.example.com:489` (for user data)
Access Manager Servers and Load Balancer	
Access Manager 1	`http://AccessManager-1.example.com:1080/amserver/console`
Access Manager 2	`http://AccessManager-2.example.com:1080/amserver/console`
Load Balancer 3	`http://LoadBalancer-3.example.com:7070` (for Intranet users) `https://LoadBalancer-3.example.com:9443` (for Internet users)
Message Queue Broker Clusters	
Message Queue 1	`http://MessageQueue-1.example.com:7777`
Message Queue 2	`http://MessageQueue-2.example.com:7777`
Distributed Authentication User Interfaces and Load Balancer	
Distributed Authentication User Interface 1	`http://AuthenticationUI-1.example.com:1080/distAuth/UI/Login`
Distributed Authentication User Interface 2	`http://AuthenticationUI-2.example.com:1080/distAuth/UI/Login`
Load Balancer 4	`http://LoadBalancer-4.example.com:90` (non-secure for internal users) `https://LoadBalancer-4.example.com:9443` (secure for external users)
Protected Resources 1 and 2, Policy Agents, and Load Balancers	

TABLE 2–3 Components and Main Service URLs *(Continued)*

Components	Main Service URL
Web Container 1	`https://ProtectedResource-1.example.com:8989` (for Sun Java System Web Server administration console)
Web Policy Agent 1	`http://ProtectedResource-1.example.com:1080`
J2EE Container 1	`http://ProtectedResource-1.example.com:7001/console` (for BEA Weblogic administration server)
J2EE Policy Agent 1	`http://ProtectedResource-1.example.com:1081`
Web Container 2	`https://ProtectedResource-2.example.com:8989` (for Sun Java System Web Server administration console)
Web Policy Agent 2	`http://ProtectedResource-2.example.com:1080`
J2EE Container 2	`http://ProtectedResource-2.example.com:7001/console` (for BEA WebLogic administration server)
J2EE Policy Agent 2	`http://ProtectedResource-2.example.com:1081`
Load Balancer 5	`http://LoadBalancer-5.example.com:90` (for web policy agents)
Load Balancer 6	`http://LoadBalancer-6.example.com:91` (for J2EE policy agents)

2.4 Intercomponent Communication

The following table provides an overview of the types of communication that take place between servers, load balancers, and other components in the deployment example.

TABLE 2–4 Summary of Intercomponent Communication

Entity A	Entity B	Bi-Directional	Port	Protocol	Traffic Type
Internet Users	LoadBalancer-5		90	HTTP	Application Traffic
Intranet Users	LoadBalancer-3		7070	HTTP	Intranet User Authentication
Internet Users	LoadBalancer-6		91	HTTP	Application Traffic
Internet Users	LoadBalancer-4		9443	HTTPS	Internet User Authentication
LoadBalancer-4	AuthenticationUI-1		1080	HTTP	Internet User Authentication
LoadBalancer-4	AuthenticationUI-2		1080	HTTP	Internet User Authentication

TABLE 2–4 Summary of Intercomponent Communication *(Continued)*

Entity A	Entity B	Bi-Directional	Port	Protocol	Traffic Type
LoadBalancer-5	ProtectedResource-1		1080	HTTP	Application Traffic
LoadBalancer-5	ProtectedResource-2		1080	HTTP	Application Traffic
LoadBalancer-6	ProtectedResource-1		1081	HTTP	Application Traffic
LoadBalancer-6	ProtectedResource-2		1081	HTTP	Application Traffic
AuthenticationUI-1	LoadBalancer-3		9443	HTTPS	Internet User Authentication
AuthenticationUI-2	LoadBalancer-3		9443	HTTPS	Internet User Authentication
ProtectedResource-1	LoadBalancer-3		9443	HTTPS	Agent-AM communication
ProtectedResource-2	LoadBalancer-3		9443	HTTPS	Agent-AM communication
LoadBalancer-3	AccessManager-1		1080	HTTP	User Authentication Agent-AM communication
LoadBalancer-3	AccessManager-2		1080	HTTP	User Authentication Agent-AM communication
AccessManager-1	AccessManager-2	Yes	1080	HTTP	AM Back-channel communication
AccessManager-1	MessageQueue-1		7777	HTTP	Session communication
AccessManager-1	LoadBalancer-1		389	LDAP	AM Configuration communication
AccessManager-1	LoadBalancer-2		489	LDAP	User profile communication User Authentication
AccessManager-2	MessageQueue-2		7777	HTTP	Session communication
AccessManager-2	LoadBalancer-1		389	LDAP	AM Configuration communication
AccessManager-2	LoadBalancer-2		489	LDAP	User profile communication User Authentication
MessageQueue-1	MessageQueue-2	Yes	7777	HTTP	Session communication
MessageQueue-2	MessageQueue-1	Yes	7777	HTTP	Session communication
LoadBalancer-1	DirectoryServer-1		1389	LDAP	AM Configuration communication
LoadBalancer-1	DirectoryServer-2		1389	LDAP	AM Configuration communication
LoadBalancer-2	DirectoryServer-1		1489	LDAP	User profile communication User Authentication
LoadBalancer-2	DirectoryServer-2		1489	LDAP	User profile communication User Authentication
DirectoryServer-1	DirectoryServer-2	Yes	1389	LDAP	Data replication communication

TABLE 2–4 Summary of Intercomponent Communication *(Continued)*

Entity A	Entity B	Bi-Directional	Port	Protocol	Traffic Type
DirectoryServer-1	DirectoryServer-2	Yes	1489	LDAP	Data replication communication

2.5 Firewall Rules

Actual firewalls are not set up in this deployment example. The intended deployment if firewalls were configured would be to protect critical components using three distinct security zones as illustrated in Figure 1–1. One zone is completely secure, protected by all three firewalls, and used for internal traffic only. The second, less secure zone is protected by only two firewalls and is also for internal traffic only. The third, minimally-secured *demilitarized zone* (DMZ) leaves only simple components and interfaces exposed to the Internet and is used for external traffic. Thus, direct access to individual Access Manager servers and Directory Server instances is allowed only if permitted by firewall rules. Based on the illustration cited:

- The Access Manager servers are isolated between an internal firewall and the DMZ. Access Manager services are exposed through both an external-facing load balancer and an internal-facing load balancer. The load balancer and Access Manager servers together provide high data availability within the infrastructure.
- The policy agents themselves are deployed behind a load balancer configured in the DMZ.
- The Distributed Authentication User Interface would be deployed in the DMZ for communication with Access Manager behind a firewall, additionally protecting the Access Manager servers from exposure in the minimally-secured DMZ.

You may set up firewalls to allow traffic to flow as described in the following table.

TABLE 2–5 Summary of Firewall Rules

From	To	Port #	Protocol	Traffic Type
Internet users	LoadBalancer-4	9443	HTTPS	User authentication
Internet users	LoadBalancer-5	90	HTTP	Application access by internet user
Internet users	LoadBalancer-6	91	HTTP	Application access by internet user
AuthenticationUI-1	LoadBalancer-3	9443	HTTPS	User authentication
AuthenticationUI-2	LoadBalancer-3	9443	HTTPS	User authentication
LoadBalancer-5	ProtectedResource-1	1080	HTTP	Application access by user
LoadBalancer-6	ProtectedResource-2	1081	HTTP	Application access by user

TABLE 2–5 Summary of Firewall Rules *(Continued)*

From	To	Port #	Protocol	Traffic Type
Intranet User	LoadBalancer-3	7070	HTTP	User authentication and various Access Manager services

2.6 Replicated Entries

Throughout this deployment example, we use `ldapsearch` to view replicated entries. An alternative would be to enable the Directory Server audit log and run `tail -f`. Enabling the audit log will also help to track changes and updates made during Access Manager configuration.

PART II

Building the Environment

This second part of the *Deployment Example 1: Access Manager 7.1 Load Balancing, Distributed Authentication UI, and Session Failover* provides the instructions for installing and configuring the deployment and its components. It contains the following chapters:

- Chapter 3, "Before You Begin"
- Chapter 4, "Installing Sun Java System Directory Server and Creating Instances for Sun Java System Access Manager Configuration Data"
- Chapter 5, "Configuring Instances of Sun Java System Directory Server for User Data"
- Chapter 6, "Installing and Configuring Access Manager"
- Chapter 7, "Configuring an Access Manager Realm for User Authentication"
- Chapter 8, "Installing and Configuring the Distributed Authentication User Interface"
- Chapter 9, "Configuring the Protected Resource Host Machines"
- Chapter 10, "Setting Up Load Balancers for the Policy Agents"
- Chapter 11, "Implementing Session Failover"

CHAPTER 3

Before You Begin

This chapter contains information you need to know before beginning the documented installation and configuration procedures. It contains the following sections:

- "3.1 Technical Conventions" on page 35
- "3.2 Setting Up the Load Balancers" on page 35
- "3.3 Obtaining Secure Socket Layer Certificates" on page 36
- "3.4 Resolving Host Names" on page 36
- "3.5 Known Issues and Limitations" on page 37

3.1 Technical Conventions

See Chapter 2, "Technical Overview," for a quick reference of host machines, port numbers, operating systems, naming conventions, and component names used in this deployment example. See Part III for more detailed information.

3.2 Setting Up the Load Balancers

The load balancer hardware and software used in this deployment environment is BIG-IP® manufactured by F5 Networks. If you are using different load balancer software, see the documentation that comes with that product for detailed settings information. This document assumes that you have already installed the required load balancers.

The following sections require load-balancing hardware and software.

- "4.3 Configuring a Load Balancer for the Directory Server Configuration Data Instances" on page 55
- "5.3 Configuring the Load Balancer for the User Data Instances" on page 72
- "6.3 Configuring the Access Manager Load Balancer" on page 100
- "8.4 Configuring the Distributed Authentication User Interface Load Balancer" on page 154
- "10.1 Configuring the Web Policy Agents Load Balancer" on page 249

- “10.2 Configuring the J2EE Policy Agents Load Balancer” on page 258

3.3 Obtaining Secure Socket Layer Certificates

You will need to obtain certificate authority (CA) root certificates and server Secure Socket Layer (SSL) certificates to enable SSL in this deployment environment. The certificate issuer used in this deployment is a test CA certificate from OpenSSL. You can obtain a root CA certificate from a commercial certificate issuer such as Verisign. For more information, see the documentation provided by your certificate authority.

The following tasks are related to requesting, installing, and importing SSL certificates:

- “To Request an Secure Sockets Layer Certificate for the Access Manager Load Balancer” on page 107
- “To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer” on page 108
- “To Install an SSL Certificate on the Access Manager Load Balancer” on page 109
- “To Import the Access Manager Load Balancer Certificate Authority Root Certificate into Distributed Authentication User Interface 1” on page 140
- “To Import the Access Manager Load Balancer Certificate Authority Root Certificate into the Distributed Authentication User Interface 2” on page 152
- “To Request a Secure Sockets Layer Certificate for the Distributed Authentication User Interface Load Balancer” on page 158
- “To Import a CA Root Certificate on the Distributed Authentication User Interface Load Balancer” on page 159
- “To Install an SSL Certificate on the Distributed Authentication User Interface Load Balancer” on page 160
- “To Import the Certificate Authority Root Certificate into the Web Server 1 Keystore” on page 176
- “To Import the Certificate Authority Root Certificate into the Application Server 1 Keystore” on page 203
- “To Import the Certificate Authority Root Certificate into the Web Server 2 Keystore” on page 218
- “To Import the CA Root Certificate into the Application Server 2 Keystore” on page 243

3.4 Resolving Host Names

There are many ways to resolve the host names used in this deployment example. You may use a DNS naming service, or you can map IP addresses to host names in the local `host` file on all UNIX® hosts. The same entries must also be added to equivalent files on Windows hosts, and on client machines where browsers are used. For example:

```
1xx.xx.xx.x1       DirectoryServer-1    DirectoryServer-1.example.com
1xx.xx.xx.x2       DirectoryServer-2    DirectoryServer-2.example.com
```

```
1xx.xx.xx.x3       AccessManager-1       AccessManager-1.example.com
1xx.xx.xx.x4       AccessManager-2       AccessManager-2.example.com
```

3.5 Known Issues and Limitations

See Appendix G, "Known Issues and Limitations," for descriptions of problems you may encounter when implementing the deployment example. This list will be updated as new information becomes available.

⚠ **Caution –** Although these instructions incorporate many recommended or *best practices*, and may be suitable in many different scenarios, this is not the only way to achieve the same results. If you plan to deviate from the task sequence or details described, you should refer to the relevant product documentation for information on differences in platforms, software versions or other requirement constraints.

◆ ◆ ◆ CHAPTER 4

Installing Sun Java System Directory Server and Creating Instances for Sun Java System Access Manager Configuration Data

This chapter contains instructions for installing Sun Java™ System Directory Server and creating the instances in which Sun Java System Access Manager configuration data will be stored. It also includes the procedure for enabling multi-master replication between the two instances and configuring the configuration data load balancer. It contains the following sections:

- "4.1 Installing and Configuring Directory Server 1 and Directory Server 2" on page 39
- "4.2 Enabling Multi-Master Replication Between the Access Manager Configuration Data Instances" on page 48
- "4.3 Configuring a Load Balancer for the Directory Server Configuration Data Instances" on page 55

4.1 Installing and Configuring Directory Server 1 and Directory Server 2

This section contains the instructions for installing Directory Server on two different host machines and creating the instances in which Access Manager configuration data will be stored. Use the following list of procedures as a checklist for completing the tasks.

1. "To Download Sun Java System Directory Server Enterprise Edition 6.0 and Required Patches" on page 40
2. "To Patch the Directory Server Host Machines" on page 41
3. "To Install Directory Server 1" on page 42
4. "To Create an Access Manager Configuration Data Instance for Directory Server 1" on page 43
5. "To Create a Base Suffix for the Directory Server 1 Access Manager Configuration Data Instance" on page 44
6. "To Install Directory Server 2" on page 45
7. "To Create the Access Manager Configuration Data Instance for Directory Server 2" on page 46

8. "To Create a Base Suffix for the Directory Server 2 Access Manager Configuration Data Instance" on page 47

▼ To Download Sun Java System Directory Server Enterprise Edition 6.0 and Required Patches

Perform this procedure to download the Sun Java System Directory Server 6.0 bits and the required system patches to both the DirectoryServer–1 host machine and the DirectoryServer–2 host machine.

1 **Go to** `http://www.sun.com/software/products/directory_srvr_ee/get.jsp`**.**

2 **Provide the following information in the** ***Select product configuration*** **section and click View Downloads.**

Step 1: Select Component	**`Directory Server Enterprise Edition`**
Step 2: Select Version	**`6.0`**
Step 3: Select Delivery Type	**`Compress Archive (ZIP)`**
Step 4: Select Platform	Choose the platform you are using.

The Selection Results page will be displayed with links to the download sites for the Directory Server and required patches.

Note – The patch numbers generated for download on the Selection Results page are based on your input. Check the most recent Directory Server Enterprise Edition 6.0 Release Notes to determine if you need to install other patches based on your machine's architecture and operating system. In this deployment, the Release Notes indicate that based on the hardware and operating system being used, patch 118855–36, patch 119964–08, and patch 122033–05 are required.

---

3 **Log into the DirectoryServer–1 host machine as a root user.**

4 **Run the** `patchadd` **command to see if the patches are already installed.**

```
# patchadd -p | grep 118855–36
```

No results are returned which indicates that the patch is not yet installed on the system.

```
# patchadd -p | grep 119964–08
```

No results are returned which indicates that the patch is not yet installed on the system.

```
# patchadd -p | grep 122033–05
```

No results are returned which indicates that the patch is not yet installed on the system.

Note – If these patches are already installed on your machine, proceed to step 7.

---

5 **Make a directory for the patch downloads and change into it.**

```
# mkdir /export/patches
# cd /export/patches
```

6 **Download the patches.**

You can click on the patch links from the Selection Results page or search for patches directly at `http://sunsolve.sun.com`. If searching directly, navigate to the PatchFinder page and enter the patch number. For each patch you are downloading, click the HTTP link beside the heading *Download Signed Patch (xxx bytes)*.

Note – Signed patches are downloaded as JAR files. Unsigned patches are downloaded as ZIP files. In this step, ZIP files are downloaded.

---

7 **Make a directory for the Directory Server download and change into it.**

```
# mkdir /export/DS6
# cd /export/DS6
```

8 **Download the** `Directory Server EE 6.0 - Zip Distribution, Multi Language, (DS/DPS/DE/ISW/DSRK) - No Console)` **bits.**

Note – No Directory Server Administration Console is installed with these bits. This deployment example uses the command line to configure the software.

---

9 **Log out of the DirectoryServer–1 host machine.**

10 **Repeat this same procedure on the DirectoryServer–2 host machine.**

▼ To Patch the Directory Server Host Machines

If necessary, perform this procedure to patch both the Directory Server 1 host machine and the Directory Server 2 host machine.

1 **Log in to the DirectoryServer–1 host machine as a root user.**

2 **Change into the directory that contains the downloaded patch files.**

```
# cd /export/patches
```

3 **Unzip the patch files.**

```
# unzip 118855–36.zip
# unzip 119964-08.zip
# unzip 122033-05.zip
```

4 **Install the patches.**

```
# patchadd /export/patches/118855-36
# patchadd /export/patches/119964-08
# patchadd /export/patches/122033-05
```

Tip – You can use the `-M` option to install all patches at once. See the `patchadd` man page for more information.

---

5 **Reboot your machine, if requested.**

6 **After installation is complete, verify that each patch was added successfully.**

```
# patchadd -p | grep 118855–36
```

A series of patch numbers are displayed, and the patch 118855–36 is present.

```
# patchadd -p | grep 119964-08
```

A series of patch numbers are displayed, and the patch 119964-08 is present.

```
# patchadd -p | grep 122033-05
```

A series of patch numbers are displayed, and the patch 122033-05 is present.

7 **Log out of the DirectoryServer–1 host machine.**

8 **Repeat this same procedure on the DirectoryServer–2 host machine.**

▼ To Install Directory Server 1

Before You Begin Patch your machine accordingly and download the Directory Server bits to the host machine.

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **Resolve the following issues, if necessary.**

- The `LD_LIBRARY_PATH` environment variable should *not* be set to the default setting. Change the value to empty as in the following example:

  ```
  # setenv LD_LIBRARY_PATH
  ```

- The JAVA_HOME environment variable should be set appropriately for your system architecture. For example:

  ```
  # setenv JAVA_HOME /usr/jdk/jdk1.5.0_07
  ```

3 **Unzip the Directory Server ZIP file.**

```
# cd /export/DS6
# ls

DSEE.6.0Solaris10-X86_AMD64-full.tar.gz

# gunzip DSEE.6.0Solaris10-X86_AMD64-full.tar.gz
```

4 **Untar the resulting Directory Server tar file.**

```
# tar xvf DSEE.6.0Solaris10-X86_AMD64-full.tar
```

5 **From the resulting directory, run `dsee_deploy install` to install Directory Server.**

```
# cd DSEE_ZIP_Distribution
# ./dsee_deploy install -c DS -i /var/opt/mps/serverroot
```

The Licensing Agreement is displayed. At each `Type return to continue` prompt, press Return to continue.

6 **When `Do you accept the license terms?` is displayed, enter yes to continue.**

Once you accept the license terms, the Directory Server binaries will be installed in the `/var/opt/mps/serverroot/ds6` directory.

▼ To Create an Access Manager Configuration Data Instance for Directory Server 1

After installing the binaries, create an instance of Directory Server 1 named `am-config` on the DirectoryServer–1 host machine. The instance uses the default ports for non-root users: 1389 for LDAP and 1636 for LDAPS. It will be populated with Access Manager configuration data in "To Configure Access Manager 1" on page 94.

Note – By default, Directory Server always creates a secure LDAP port when creating an instance. We do not use this port.

Before You Begin This procedure assumes you have just completed "To Install Directory Server 1" on page 42.

1 **As a root user on the DirectoryServer–1 host machine, run `dsadm create` to create the instance.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm create -p 1389 -P 1636 /var/opt/mps/am-config
Choose the Directory Manager password: d1rm4n4ger
Confirm the Directory Manager password: d1rm4n4ger

use 'dsadm start /var/opt/mps/am-config' to start the instance
```

2 **Run `dsadm start` to start the instance.**

```
# ./dsadm start /var/opt/mps/am-config

Server started: pid=10381
```

3 **Run `netstat` to verify that the new instance is up and running.**

```
# netstat -an | grep 1389

.1389        *.*        0        0  49152        0 LISTEN
```

4 **Run `ldapsearch` to verify that you can read the root Directory Server entry (DSE) of the new instance.**

```
# ldapsearch -h DirectoryServer-1.example.com
-p 1389 -b "" -s base "(objectclass=*)"

version: 1
dn:
objectClass: top
...
supportedLDAPVersion: 3
vendorname: Sun Microsystems, Inc.
vendorVersion: Sun-Java(tm)-System-Directory/6.0
...
```

▼ To Create a Base Suffix for the Directory Server 1 Access Manager Configuration Data Instance

After creating the configuration data instance of DirectoryServer–1, create a base suffix in which the entries will be stored.

Before You Begin This procedure assumes you have just completed "To Create an Access Manager Configuration Data Instance for Directory Server 1" on page 43.

1 **As a root user on the Directory Server 1 host machine, run `dsconf create-suffix` to create a new base suffix.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf create-suffix -p 1389 -B dbExample
-L /var/opt/mps/am-config/db/exampleDS dc=example,dc=com
```

2 **Provide the appropriate information when prompted.**

```
Certificate "CN=DirectoryServer-1, CN=1636, CN=directory Server, O=Sun Microsystems"
presented by the server is not trusted.
Type "Y" to accept, "y" to accept just one, "n" to refuse, "d" for more details: Y
Enter "cn=Directory Manager" password: d1rm4n4ger
```

Tip – When you enter an uppercase `Y`, you are not asked for the certificate again in the next steps.

3 **Run `dsconf list-suffixes` to verify that the base suffix was successfully created.**

```
# ./dsconf list-suffixes -p 1389
Enter "cn=Directory Manager" password: d1rm4n4ger

dc=example,dc=com
```

4 **Log out of the Directory Server 1 host machine.**

To Install Directory Server 2

Before You Begin Patch your machine accordingly and download the Directory Server bits to the host machine.

1 **As a root user, log in to the Directory Server 2 host machine.**

2 **Resolve the following issues, if necessary.**

- The `LD_LIBRARY_PATH` environment variable should *not* be set to the default setting. Change the value to empty as in the following example:

  ```
  # setenv LD_LIBRARY_PATH
  ```

- The `JAVA_HOME` environment variable should be set appropriately for your system architecture. For example:

  ```
  # setenv JAVA_HOME /usr/jdk/jdk1.5.0_07
  ```

3 **Unzip the Directory Server ZIP file.**

```
# cd /export/DS6
# ls
```

```
DSEE.6.0Solaris10-X86_AMD64-full.tar.gz

# gunzip DSEE.6.0Solaris10-X86_AMD64-full.tar.gz
```

4. **Untar the resulting Directory Server tar file.**

```
# tar xvf DSEE.6.0Solaris10-X86_AMD64-full.tar
```

5. **In the resulting directory, run `dsee_deploy install` to install Directory Server.**

```
# cd DSEE_ZIP_Distribution
# ./dsee_deploy install -c DS -i /var/opt/mps/serverroot
```

The Licensing Agreement is displayed. At each `Type return to continue` prompt, press Return to continue.

6. **When `Do you accept the license terms?` is displayed, enter `yes` to continue.**

Once you accept the license terms, the Directory Server binaries will be installed in the `/var/opt/mps/serverroot/ds6` directory.

▼ To Create the Access Manager Configuration Data Instance for Directory Server 2

After installing the binaries, create an instance of Directory Server 2 named `am-config` on the DirectoryServer–2 host machine. The instance uses the default ports for non-root users: 1389 for LDAP and 1636 for LDAPS. It will be populated with Access Manager configuration data in "To Configure Access Manager 2" on page 96.

Note – By default, Directory Server always creates a secure LDAP port when creating an instance. We do not use this port.

Before You Begin This procedure assumes you have just completed "To Install Directory Server 2" on page 45.

1. **As a root user on the DirectoryServer–2 host machine, run `dsadm create` to create the instance.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm create -p 1389 -P 1636 /var/opt/mps/am-config
Choose the Directory Manager password: d1rm4n4ger
Confirm the Directory Manager password: d1rm4n4ger

use 'dsadm start /var/opt/mps/am-config' to start the instance
```

2 **Run `dsadm start` to start the instance.**

```
# ./dsadm start /var/opt/mps/am-config

Server started: pid=10381
```

3 **Run `netstat` to verify that the new instance is up and running.**

```
# netstat -an | grep 1389

.1389        *.*        0        0  49152        0 LISTEN
```

4 **Run `ldapsearch` to verify that you can read the root DSE of the new instance.**

```
# ldapsearch -h DirectoryServer-2.example.com
-p 1389 -b "" -s base "(objectclass=*)"

version: 1
dn:
objectClass: top
...
supportedLDAPVersion: 3
vendorname: Sun Microsystems, Inc.
vendorVersion: Sun-Java(tm)-System-Directory/6.0
...
```

▼ To Create a Base Suffix for the Directory Server 2 Access Manager Configuration Data Instance

After creating the configuration data instance of DirectoryServer–2, create a base suffix in which the entries will be stored.

Before You Begin This procedure assumes you have completed "To Create the Access Manager Configuration Data Instance for Directory Server 2" on page 46.

1 **As a root user on the DirectoryServer–2 host machine, run `dsconf create-suffix` to create a new base suffix.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf create-suffix -p 1389 -B dbExample
-L /var/opt/mps/am-config/db/exampleDS dc=example,dc=com
```

2 **Provide the appropriate information when prompted.**

```
Certificate "CN=DirectoryServer-2, CN=1636, CN=directory Server, O=Sun Microsystems"
presented by the server is not trusted.
Type "Y" to accept, "y" to accept just one, "n" to refuese, "d" for more details: Y
```

```
Enter "cn=Directory Manager" password: d1rm4n4ger
```

Tip – When you enter an uppercase **Y**, you are not asked for the certificate again in the next steps.

3 **Run `dsconf list-suffixes` to verify that the base suffix was successfully created.**

```
# ./dsconf list-suffixes -p 1389
Enter "cn=Directory Manager" password: d1rm4n4ger

dc=example,dc=com
```

4 **Log out of the DirectoryServer–2 host machine.**

4.2 Enabling Multi-Master Replication Between the Access Manager Configuration Data Instances

This section contains the instructions to enable multi-master replication (MMR) between two directory masters. This includes creating replication agreements between the masters and initializing the second directory master with the data and schema from the first directory master. The previously created `am-config` instances will serve as the two masters. An illustration of the architecture can be seen in Figure 4–1.

Use the following list of procedures as a checklist for completing the tasks.

1. "To Enable Multi-Master Replication for the Directory Server 1 Configuration Data Instance" on page 48
2. "To Enable Multi-Master Replication for the Directory Server 2 Configuration Data Instance" on page 49
3. "To Change the Default Replication Manager Passwords for Each Configuration Data Instance" on page 50
4. "To Create Replication Agreements for Each Configuration Data Instance" on page 52
5. "To Initialize the Configuration Data Instance Replication Agreements" on page 53
6. "To Verify that Configuration Data Replication Works Properly" on page 54

▼ To Enable Multi-Master Replication for the Directory Server 1 Configuration Data Instance

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **(Optional) Run `dsconf list-suffixes` to verify that the instance is not already enabled for replication.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf list-suffixes -p 1389 -v
Enter "cn=Directory Manager" password: d1rm4n4ger
...
dc=example,dc=com     1        not-replicated        N/A        N/A        29
The "list-suffixes" operation succeeded on "localhost:1389"
```

The base suffix of the instance is `not-replicated` as displayed in the resulting list.

3 **Run `dsconf enable-repl` to enable replication.**

```
# ./dsconf enable-repl -h DirectoryServer-1.example.com
-p 1389 -d 11 master dc=example,dc=com
Enter "cn=Directory Manager" password: d1rm4n4ger

Use "dsconf create-repl-agmt" to create replication agreements on
"dc=example,dc=com".
```

The `-d` option takes as input a randomly chosen identifier to represent the Directory Server 1 configuration data instance; in this case, `11`. `master` indicates that the instance is a master and not a replica. The base suffix is specified as `dc=example,dc=com`.

4 **Run `dsconf list-suffixes` again to verify that the instance is now enabled for replication.**

```
# ./dsconf list-suffixes -p 1389 -v
Enter "cn=Directory Manager" password: d1rm4n4ger
...
dc=example,dc=com     1        master(11)        N/A        N/A        29
The "list-suffixes" operation succeeded on "localhost:1389"
```

The base suffix of the instance is `master(11)` as displayed in the resulting list, indicating that the master was successfully enabled.

5 **Log out of the DirectoryServer–1 host machine.**

▼ To Enable Multi-Master Replication for the Directory Server 2 Configuration Data Instance

1 **As a root user, log in to the DirectoryServer–2 host machine.**

2 **(Optional) Run the `dsconf list-suffixes` command to verify that the instance is not already enabled for replication.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf list-suffixes -p 1389 -v
Enter "cn=Directory Manager" password: d1rm4n4ger
...
dc=example,dc=com     1        not-replicated        N/A        N/A        29
The "list-suffixes" operation succeeded on "localhost:1389"
```

The base suffix of the instance is `not-replicated` as displayed in the resulting list.

3 **Run `dsconf enable-repl` to enable replication.**

```
# ./dsconf enable-repl -h DirectoryServer-2.example.com
-p 1389 -d 22 master dc=example,dc=com
Enter "cn=Directory Manager" password: d1rm4n4ger

Use "dsconf create-repl-agmt" to create replication agreements on
"dc=example,dc=com".
```

The `-d` option takes as input a randomly chosen identifier to represent the Directory Server 2 configuration data instance; in this case, 22. `master` indicates that the instance is a master and not a replica. The base suffix is specified as `dc=example,dc=com`.

4 **Run `dsconf list-suffixes` again to verify that the instance is now enabled for replication.**

```
# ./dsconf list-suffixes -p 1389 -v
Enter "cn=Directory Manager" password: d1rm4n4ger
...
dc=example,dc=com     1        master(22)        N/A        N/A        29
The "list-suffixes" operation succeeded on "localhost:1389"
```

The base suffix of the instance is `master(22)` as displayed in the resulting list, indicating that the master was successfully enabled.

5 **Log out of the DirectoryServer–2 host machine.**

To Change the Default Replication Manager Passwords for Each Configuration Data Instance

The *replication manager* is the user that suppliers use to bind to the consumer server when sending replication updates. (In MMR the consumer server refers to whichever master happens to be the consumer for that particular operation.) It is recommended by the Directory Server documentation to change the default password created during the process of enabling replication.

1. **As a root user, log in to the DirectoryServer–1 host machine.**

2. **Create a temporary file that contains the new replication manager password.**

   This file will be read once, and the password stored for future use.

   ```
   # cd /var/opt/mps/serverroot/ds6/bin
   # echo replm4n4ger > pwd.txt
   ```

3. **Verify that the file was successfully created.**

   ```
   # cat pwd.txt

   replm4n4ger
   ```

4. **Run `dsconf set-server-prop` to set the new replication manager password using `pwd.txt` as input.**

   ```
   # ./dsconf set-server-prop -h DirectoryServer-1.example.com
     -p 1389 def-repl-manager-pwd-file:pwd.txt
   Enter "cn=Directory Manager" password: d1rm4n4ger
   ```

5. **Remove the `pwd.txt` file.**

6. **Log out of the DirectoryServer–1 host machine.**

7. **As a root user, log in to the DirectoryServer–2 host machine.**

8. **Create a temporary file that contains the new replication manager password.**

   This file will be read once, and the password stored for future use.

   ```
   # cd /var/opt/mps/serverroot/ds6/bin
   # echo replm4n4ger > pwd.txt
   ```

9. **Verify that the file was successfully created.**

   ```
   # cat pwd.txt

   replm4n4ger
   ```

10. **Run `dsconf set-server-prop` to set the new replication manager password using `pwd.txt` as input.**

    ```
    # ./dsconf set-server-prop -h DirectoryServer-2.example.com
      -p 1389 def-repl-manager-pwd-file:pwd.txt
    Enter "cn=Directory Manager" password: d1rm4n4ger
    ```

11. **Remove the `pwd.txt` file.**

12. **Log out of the DirectoryServer–2 host machine.**

▼ To Create Replication Agreements for Each Configuration Data Instance

A *replication agreement* is a set of parameters on a supplier that controls how updates are sent to a given consumer. In this case, we are making the configuration data instances aware of each other.

1. **As a root user, log in to the DirectoryServer–1 host machine.**

2. **Run `dsconf create-repl-agmt` to create the replication agreement.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf create-repl-agmt -h DirectoryServer-1.example.com
  -p 1389 dc=example,dc=com DirectoryServer-2.example.com:1389
Enter "cn=Directory Manager" password: d1rm4n4ger

Use "dsconf init-repl-dest dc=example,dc=com DirectoryServer-2.example.com:1389"
to start replication of "dc=example,dc=com" data.
```

3. **Run `dsconf list-repl-agmts` to verify that the replication agreement was successfully created.**

```
# ./dsconf list-repl-agmts -p 1389
Enter "cn=Directory Manager" password: d1rm4n4ger

dc=example,dc=com DirectoryServer-2.example.com:1389
```

The response indicates that the Directory Server 1 configuration data base suffix will be replicated to Directory Server 2.

4. **Log out of the DirectoryServer–1 host machine.**

5. **As a root user, log in to the DirectoryServer–2 host machine.**

6. **Run `dsconf create-repl-agmt` to create the replication agreement.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf create-repl-agmt -h DirectoryServer-2.example.com
  -p 1389 dc=example,dc=com DirectoryServer-1.example.com:1389
Enter "cn=Directory Manager" password: d1rm4n4ger

Use "dsconf init-repl-dest dc=example,dc=com DirectoryServer-1.example.com:1389"
to start replication of "dc=example,dc=com" data.
```

7 **Run** `dsconf list-repl-agmts` **to verify that the replication agreement was successfully created.**

```
# ./dsconf list-repl-agmts -p 1389
Enter "cn=Directory Manager" password: d1rm4n4ger

dc=example,dc=com DirectoryServer-1.example.com:1389
```

The response indicates that the Directory Server 2 configuration data base suffix will be replicated to Directory Server 1.

8 **Log out of the DirectoryServer–2 host machine.**

▼ To Initialize the Configuration Data Instance Replication Agreements

In this procedure, initialize the configuration data instance on Directory Server 1. The previously created replication agreement will replicate the data to Directory Server 2.

Note – Initialization is **not** required on both instances when configuring for MMR.

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **Run** `dsconf show-repl-agmt-status` **to verify that the replication agreements have not yet been initialized.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf show-repl-agmt-status -h DirectoryServer-1.example.com
  -p 1389 dc=example,dc=com DirectoryServer-2.example.com:1389
Enter "cn=Directory Manager" password: d1rm4n4ger

Configuration Status         : OK
Authentication Status        : OK
Initialization Status        : NOT OK

Status:                                       : Dest. Not Initialized
```

3 **Run** `dsconf init-repl-dest` **to initialize the replication agreements.**

```
# ./dsconf init-repl-dest -h DirectoryServer-1.example.com
  -p 1389 dc=example,dc=com DirectoryServer-2.example.com:1389
Enter "cn=Directory Manager" password: d1rm4n4ger

Sent 1 entries...
Sent 2 entries...
```

```
Completed initialization of "DirectoryServer-2.example.com:1389";
May 15, 2007 1:53:32 PM
```

4 **Run** `dsconf show-repl-agmt-status` **again to verify that the replication agreements are now initialized.**

```
# ./dsconf show-repl-agmt-status -h DirectoryServer-1.example.com
  -p 1389 dc=example,dc=com DirectoryServer-2.example.com:1389
Enter "cn=Directory Manager" password: d1rm4n4ger

Configuration Status         : OK
Authentication Status        : OK
Initialization Status        : OK

Status:                                        : Enabled
Last Update Date                        : Jul 12, 2007 8:47 PM
```

5 **Log out of the DirectoryServer–1 host machine.**

▼ To Verify that Configuration Data Replication Works Properly

1 **As a root user, log in to the Directory Server 1 host machine.**

2 **Run** `ldapmodify` **to create a new directory entry.**

```
# ldapmodify -a -h DirectoryServer-1.example.com -p 1389
  -D cn=admin,cn=Administrators,cn=config -w d1rm4n4ger

dn: ou=People,dc=example,dc=com
objectclass: top
objectclass: organizationalUnit
ou: People
description: Container for user entries
```

Hit ENTER to indicate end of input.

```
adding new entry ou=People,dc=example,dc=com
```

Hit Control C to terminate the command.

```
^C
```

This step creates a new organization unit on Directory Server 1.

3 **As a root user, log in to the Director Server–2 host machine.**

4 **Run `ldapsearch` on Directory Server 2 to verify that the entry was successfully replicated.**

```
# ldapsearch -b "dc=example,dc=com" -p 1389 -D "cn=Directory Manager"
  -w d1rm4n4ger "objectclass=organizationalUnit"

version: 1
dn: ou=People,dc=example,dc=com
objectClass: top
objectClass: organizationalUnit
ou: People
description Container for user entries
```

5 **Run `ldapdelete` on Directory Server 2 to delete the entry.**

```
# ldapdelete -h DirectoryServer-2.example.com -p 1389
  -D "cn=Directory Manager" -w d1rm4n4ger "ou=People,dc=example,dc=com"
```

6 **Run `ldapsearch` on Directory Server 1 to verify that the entry was deleted.**

```
# ldapsearch -b "dc=example,dc=com" -p 1389 -D "cn=Directory Manager"
  -w d1rm4n4ger "objectclass=organizationalUnit"
```

If the delete was successfully replicated to Directory Server 1, the search will return no results.

7 **Log out of the Directory Server host machines.**

4.3 Configuring a Load Balancer for the Directory Server Configuration Data Instances

Two load balancers are configured for the Directory Server installations. Load Balancer 1 fronts the configuration data instances and Load Balancer 2 fronts the user data instances. In the following procedure, you configure a load balancer in front of the configuration data instances. The following figure illustrates this architecture.

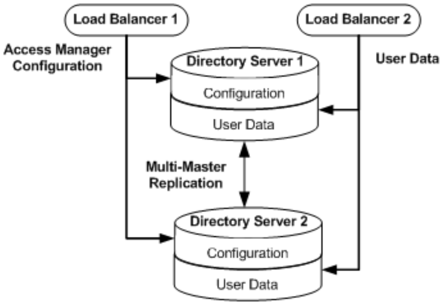


FIGURE 4–1 Directory Server Instances Configured for Multi-Master Replication and Load Balancing

▼ To Configure the Access Manager Configuration Data Load Balancer 1

Before You Begin

- This procedure assumes that you have already installed a load balancer.
- The load balancer hardware and software used in this deployment is BIG-IP® manufactured by F5 Networks. If you are using different load balancer software, see the documentation that comes with that product for detailed settings information.
- Contact your network administrator to obtain an available virtual IP address for the load balancer you want to configure.
- Know the IP address of the load balancer hardware, the URL for the load balancer login page, and a username and password for logging in to the load balancer application.
- Get the IP addresses for Directory Server 1 and Directory Server 2 by running the following command on each host machine:

```
# ifconfig -a
```

1 **Access `https://is-f5.example.com`, the BIG-IP load balancer login page, in a web browser.**

2 **Log in using the following information:**

User name: **`username`**

Password: **`password`**

3 **Click** ***Configure your BIG-IP (R) using the Configuration Utility*****.**

4 **Create a Pool.**

A pool contains all the backend server instances.

a. **In the left pane, click Pools.**

b. On the Pools tab, click Add.

c. In the Add Pool dialog, provide the following information:

Pool Name	**`DirectoryServer-ConfigData-Pool`**
Load Balancing Method	Round Robin
Resources	Add the IP address and port number of both Directory Server hosts: `DirectoryServer-1:1389` and `DirectoryServer-2:1389`.

d. Click Done.

5 Add a Virtual Server.

This step defines instances of the load balancer.

Tip – If you encounter JavaScript™ errors or otherwise cannot proceed to create a virtual server, try using Internet Explorer.

---

a. In the left frame, Click Virtual Servers.

b. On the Virtual Servers tab, click Add.

c. In the Add a Virtual Server dialog box, provide the following information:

Address	Enter the IP address for the `LoadBalancer-1.example.com`
Service	**`389`**
Pool	**`DirectoryServer-ConfigData-Pool`**

d. Continue to click Next until you reach the Pool Selection dialog box.

e. In the Pool Selection dialog box, assign `DirectoryServer-ConfigData-Pool` to the virtual server.

f. Click Done.

6 Add Monitors

Monitors are required by the load balancer to detect backend server failures.

a. In the left frame, click Monitors.

b. Click the Basic Associations tab.

c. **Add an LDAP monitor for the Directory Server 1 node.**

In the Node column, locate the IP address and port number, `DirectoryServer–1:1389`, and select the Add checkbox.

d. **Add an LDAP monitor for the Directory Server 2 node.**

In the Node column, locate the IP address and port number, `DirectoryServer–2:1389`, and select the Add checkbox.

e. **At the top of the Node column, in the drop-down list, choose `ldap-tcp`.**

f. **Click Apply.**

7 **Configure the load balancer for simple persistence.**

The configuration data load balancer is configured for *simple persistence*. With simple persistence, all requests sent *within a specified interval* are processed by the same Directory Server instance, ensuring complete replication of entries. For example, when a request requires information to be written to Directory Server 1, that information must also be replicated to Directory Server 2. As the replication takes time to complete, if a related request is directed by the load balancer to Directory Server 2 during the replication process itself, the request may fail as the entry might only be partially created. When properly configured, simple persistence ensures that both requests are routed to Directory Server 1 and processed in consecutive order; the first request is finished before the second request begins processing. Simple persistence ensures that within the specified interval, no errors or delays occur due to replication time or redirects when retrieving data. Simple persistence tracks connections based only on the client IP address.

a. **In the left frame, click Pools.**

b. **Click the name of the pool you want to configure.**

In this example, `DirectoryServer-ConfigData-Pool`.

c. **Click the Persistence tab.**

d. **Under Persistence Type, select Simple.**

e. **Enter 300 seconds for the Timeout interval.**

f. **Click Apply.**

8 **Verify the Directory Server load balancer configuration.**

a. **Log in as a root user to the host machine of each Directory Server instance.**

b. **On each Directory Server host machine, use the `tail` command to monitor the Directory Server access log.**

```
# cd /var/opt/mps/am-config/logs
# tail -f access
```

You should see connections to the load balancer IP address opening and closing. For example:

```
[12/Oct/2006:13:10:20-0700] conn=54 op=-1 msgId=-1 —
fd=22 slot=22 LDAP connection from IP_address to IP_address
[12/Oct/2006:13:10:20-0700] conn=54 op=-1 msgId=-1 — closing — B1
[12/Oct/2006:13:10:20-0700] conn=54 op=-1 msgId=-1 — closed.
```

c. **Execute the following LDAP search against the Directory Server load balancer.**

```
# ldapsearch -h LoadBalancer-1.example.com -p 389 -b "dc=example,dc=com"
  -D "cn=directory manager" -w d1rm4n4ger "(objectclass=*)"
```

The `ldapsearch` operation should return entries. Make sure they display in the access log on only one Directory Server.

d. **Run `dsadm stop` to stop Directory Server 1.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm stop /var/opt/mps/am-config
```

e. **Again perform the LDAP search against the Directory Server load balancer to confirm that the request is forwarded to the running Directory Server 2.**

```
# ldapsearch -h LoadBalancer-1.example.com -p 389 -b "dc=example,dc=com"
  -D "cn=directory manager" -w d1rm4n4ger "(objectclass=*)"
```

The `ldapsearch` operation should return entries. Verify that the entries display in the access log only on Directory Server 2.

Note – You may encounter the following error message:

```
ldap_simple_bind: Cant’ connect to the LDAP
server — Connection refused
```

This means that the load balancer may not fully detect that Directory Server 1 is stopped. In this case, you may have started the search too soon based on the polling interval setting. For example, if the polling interval is set to 10 seconds, you should wait ten seconds to start the search. You can reset the timeout properties to a lower value using the following procedure.

a. **Click the Monitors tab.**

b. **Click the ldap-tcp monitor name.**

c. **In the Interval field, set the value to 5.**

This tells the load balancer to poll the server every 5 seconds.

d. **In the Timeout field, set the value to 16.**

e. **Click Apply and repeat the LDAP search.**

See your load balancer documentation for more information on the timeout property.

f. **Start Directory Server 1.**

```
# ./dsadm start /var/opt/mps/am-config
```

g. **Stop Directory Server 2.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm stop /var/opt/mps/am-config
```

h. **Perform the following LDAP search against the Directory Server load balancer to confirm that the request is forwarded to the running Directory Server 1.**

```
# ldapsearch -h LoadBalancer-1.example.com -p 389 -b "dc=example,dc=com"
  -D "cn=Directory Manager" -w d1rm4n4ger "(objectclass=*)"
```

The `ldapsearch` operation should return entries. Verify that the entries display in the access log only on Directory Server 1.

i. **Start Directory Server 2.**

```
# ./dsadm start /var/opt/mps/am-config
```

j. **Log out of both Directory Server host machines and the load balancer console.**

CHAPTER 5

Configuring Instances of Sun Java System Directory Server for User Data

This chapter contains instructions for creating instances of Directory Server to hold user data called `am-users`. If you have an existing user data store, you can go directly to the instructions in "7.2 Creating and Configuring a Realm for Test Users" on page 121 to configure Access Manager to recognize your data store and users. This chapter contains the following sections:

- "5.1 Creating Directory Server Instances for User Data" on page 61
- "5.2 Enabling Multi-Master Replication of the User Data Instances" on page 65
- "5.3 Configuring the Load Balancer for the User Data Instances" on page 72

5.1 Creating Directory Server Instances for User Data

This section contains information on creating user data instances on the Directory Server 1 and Directory Server 2 host machines. Use the following list of procedures as a checklist for these tasks.

1. "To Create a User Data Instance for Directory Server 1" on page 61
2. "To Create a Base Suffix for the User Data Instance on Directory Server 1" on page 62
3. "To Create a User Data Instance for Directory Server 2" on page 63
4. "To Create a Base Suffix for the User Data Instance on Directory Server 2" on page 64

▼ To Create a User Data Instance for Directory Server 1

In this procedure, you create a Directory Server instance named `am-users` for storing user data on Directory Server 1. The new instance uses the ports for non-root users: 1489 for LDAP and 1736 for LDAPS. This instance will be populated with user information in Chapter 7, "Configuring an Access Manager Realm for User Authentication."

Note – By default, Directory Server always creates a secure LDAP port when creating an instance. We do not use this port.

1. **As a root user, log in to the DirectoryServer–1 host machine.**

2. **Run `dsadm create` to create a user data instance called `am-users`.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm create -p 1489 -P 1736 /var/opt/mps/am-users
Choose the Directory Manager password: d1rm4n4ger
Confirm the Directory Manager password: d1rm4n4ger

Use 'dsadm start /var/opt/mps/am-users' to start the instance
```

3. **Run `dsadm start` to start the instance.**

```
# ./dsadm start /var/opt/mps/am-users

Server started: pid=10381
```

4. **Run `netstat` to verify that the new instance is up and running.**

```
# netstat -an | grep 1489

.1489        *.*        0        0  49152        0 LISTEN
```

5. **Run `ldapsearch` to verify that you can read the root Directory Server entry (DSE) of the new instance.**

```
# ldapsearch -h DirectoryServer-1.example.com
  -p 1489 -b "" -s base "(objectclass=*)"

version: 1
dn:
objectClass: top
...
supportedLDAPVersion: 3
vendorname: Sun Microsystems, Inc.
vendorVersion: Sun-Java(tm)-System-Directory/6.0
...
```

To Create a Base Suffix for the User Data Instance on Directory Server 1

After creating the user data instance, you create a base suffix in which the entries will be stored.

Before You Begin This procedure assumes you have just completed "To Create a User Data Instance for Directory Server 1" on page 61.

1 As a root user on the DirectoryServer–1 host machine, run `dsconf create-suffix` to create a base suffix.

```
# ./dsconf create-suffix -p 1489 -B dbExample
  -L /var/opt/mps/am-users/db/exampleDS dc=company,dc=com
```

2 Provide information when prompted.

```
Certificate "CN=DirectoryServer-1, CN=1736, CN=directory Server, O=Sun Microsystems"
presented by the server is not trusted.
Type "Y" to accept, "y" to accept just one, "n" to refuese, "d" for more details: Y
Enter "cn=Directory Manager" password: d1rm4n4ger
```

Note – When you enter an uppercase **Y**, you are not asked for the certificate again in the next steps.

---

3 Run `dsconf list-suffixes` to verify that the base suffix was successfully created.

```
# ./dsconf list-suffixes -p 1489
Enter "cn=Directory Manager" password: d1rm4n4ger

dc=company,dc=com
```

If the base suffix was successfully created, `dc=company, dc=com` is returned. You can also see `am-users` in the list of directory instances:

```
# cd /var/opt/mps
# ls

am-config        am-users        serverroot
```

4 Log out of the DirectoryServer–1 host machine.

▼ To Create a User Data Instance for Directory Server 2

In this procedure, you create a Directory Server instance named `am-users` for storing user data on Directory Server 2. The new instance uses the ports for non-root users: 1489 for LDAP and 1736 for LDAPS. This instance will be populated with user information in Chapter 7, "Configuring an Access Manager Realm for User Authentication."

Note – By default, Directory Server always creates a secure LDAP port when creating an instance. We do not use this port.

---

1. **As a root user, log in to the DirectoryServer–2 host machine.**

2. **Run `dsadm create` to create a user data instance called `am-users`.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm create -p 1489 -P 1736 /var/opt/mps/am-users
Choose the Directory Manager password: d1rm4n4ger
Confirm the Directory Manager password: d1rm4n4ger

Use 'dsadm start /var/opt/mps/am-users' to start the instance
```

3. **Run `dsadm start` to start the instance.**

```
# ./dsadm start /var/opt/mps/am-users

Server started: pid=10381
```

4. **Run `netstat` to verify that the new instance is up and running.**

```
# netstat -an | grep 1489

.1489        *.*        0        0  49152        0 LISTEN
```

5. **Run `ldapsearch` to verify that you can read the root DSE of the new instance.**

```
# ldapsearch -h DirectoryServer-2.example.com
  -p 1489 -b "" -s base "(objectclass=*)"

version: 1
dn:
objectClass: top
...
supportedLDAPVersion: 3
vendorname: Sun Microsystems, Inc.
vendorVersion: Sun-Java(tm)-System-Directory/6.0
...
```

To Create a Base Suffix for the User Data Instance on Directory Server 2

After creating an instance, you must create a base suffix in which the entries will be stored.

Before You Begin This procedure assumes you have just completed "To Create a User Data Instance for Directory Server 2" on page 63.

1. **As a root user on the DirectoryServer–2 host machine, run `dsconf create-suffix` to create a base suffix.**

```
# ./dsconf create-suffix -p 1489 -B dbExample
  -L /var/opt/mps/am-users/db/exampleDS dc=company,dc=com
```

2. **Provide information when prompted.**

```
Certificate "CN=DirectoryServer-2, CN=1736, CN=directory Server, O=Sun Microsystems"
presented by the server is not trusted.
Type "Y" to accept, "y" to accept just one, "n" to refuese, "d" for more details: Y
Enter "cn=Directory Manager" password: d1rm4n4ger
```

Note – When you enter an uppercase **Y**, you are not asked for the certificate again in the next steps.

3. **Run `dsconf list-suffixes` to verify that the base suffix was successfully created.**

```
# ./dsconf list-suffixes -p 1489
Enter "cn=Directory Manager" password: d1rm4n4ger
dc=company,dc=com
```

If the base suffix was successfully created, `dc=company, dc=com` is returned. You can also see `am-users` in the list of directory instances as follows:

```
# cd /var/opt/mps
# ls

am-config        am-users        serverroot
```

4. **Log out of the DirectoryServer–2 host machine.**

5.2 Enabling Multi-Master Replication of the User Data Instances

This section contains the instructions to enable multi-master replication (MMR) between two directory masters. This includes creating replication agreements between the masters and initializing the second directory master with the data and schema from the first directory master. The previously created `am-users` instances will serve as the two masters. An illustration of the architecture can be seen in Figure 4–1.

Use the following list of procedures as a checklist for completing the tasks.

1. "To Enable Multi-Master Replication for User Data Instance on Directory Server 1" on page 66
2. "To Enable Multi-Master Replication for User Data Instance on Directory Server 2" on page 67
3. "To Change the Default Replication Manager Passwords for Each User Data Instance" on page 68
4. "To Create Replication Agreements for Each User Data Instance" on page 69
5. "To Initialize the User Data Instance Replication Agreements" on page 70
6. "To Verify that User Data Replication Works Properly" on page 71

▼ To Enable Multi-Master Replication for User Data Instance on Directory Server 1

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **(Optional) Run** `dsconf list-suffixes` **to verify that the instance is not already enabled for replication.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf list-suffixes -p 1489 -v
Enter "cn=Directory Manager" password: d1rm4n4ger
...
dc=company,dc=com     1        not-replicated        N/A        N/A        29
The "list-suffixes" operation succeeded on "DirectoryServer-1.example.com:1489"
```

The base suffix of the instance is `not-replicated` as displayed in the resulting list.

3 **Run** `dsconf enable-repl` **to enable replication.**

```
# ./dsconf enable-repl -h DirectoryServer-1.example.com
  -p 1489 -d 11 master dc=company,dc=com
Enter "cn=Directory Manager" password: d1rm4n4ger
Use "dsconf create-repl-agmt" to create replication agreements on
"dc=company,dc=com".
```

The `-d` option takes as input a randomly chosen identifier to represent the Directory Server 1 configuration data instance; in this case, 11. `master` indicates that the instance is a master and not a replica. The base suffix is specified as `dc=company,dc=com`.

4 **Run** `dsconf list-suffixes` **again to verify that the instance is now enabled for replication.**

```
# ./dsconf list-suffixes -p 1489 -v
Enter "cn=Directory Manager" password: d1rm4n4ger
...
dc=company,dc=com     1        master(11)        N/A        N/A        29
The "list-suffixes" operation succeeded on
"DirectoryServer-1.example.com:1489"
```

The base suffix of the instance is master(11) as displayed in the resulting list, indicating that the master was successfully enabled.

5 **Log out of the DirectoryServer–1 host machine.**

▼ To Enable Multi-Master Replication for User Data Instance on Directory Server 2

1 **As a root user, log in to the DirectoryServer–2 host machine.**

2 **(Optional) Run** dsconf list-suffixes **to verify that the instance is not already enabled for replication.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf list-suffixes -p 1489 -v
Enter "cn=Directory Manager" password: d1rm4n4ger
...
dc=company,dc=com     1        not-replicated        N/A        N/A        29
The "list-suffixes" operation succeeded on "DirectoryServer-2.example.com:1489"
```

The base suffix of the instance is not-replicated as displayed in the resulting list.

3 **Run** dsconf enable-repl **to enable replication.**

```
# ./dsconf enable-repl -h DirectoryServer-2.example.com
  -p 1489 -d 22 master dc=company,dc=com
Enter "cn=Directory Manager" password: d1rm4n4ger
Use "dsconf create-repl-agmt" to create replication agreements on
"dc=company,dc=com".
```

The -d option takes as input a randomly chosen identifier to represent the Directory Server 1 configuration data instance; in this case, 22. master indicates that the instance is a master and not a replica. The base suffix is specified as dc=company,dc=com.

4 **Run** dsconf list-suffixes **again to verify that the instance is now enabled for replication.**

```
# ./dsconf list-suffixes -p 1489 -v
Enter "cn=Directory Manager" password: d1rm4n4ger
...
dc=company,dc=com     1        master(22)        N/A        N/A        29
The "list-suffixes" operation succeeded on "DirectoryServer-2.example.com:1489"
```

The base suffix of the instance is master(22) as displayed in the resulting list, indicating that the master was successfully enabled.

5 **Log out of the DirectoryServer–2 host machine.**

▼ To Change the Default Replication Manager Passwords for Each User Data Instance

The *replication manager* is the user that suppliers use to bind to the consumer server when sending replication updates. (In MMR the consumer server refers to whichever master happens to be the consumer for that particular operation.) It is recommended by the Directory Server documentation to change the default password created during the process of enabling replication.

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **Create a temporary file that contains the new replication manager password.**

This file is read once, and the password is stored for future use.

```
# cd /var/opt/mps/serverroot/ds6/bin
# echo replm4n4ger > pwd.txt
```

3 **Verify that the file was successfully created.**

```
# cat pwd.txt

replm4n4ger
```

4 **Run `dsconf set-server-prop` to set the replication manager password using `pwd.txt` as input.**

```
# ./dsconf set-server-prop -h DirectoryServer-1.example.com
  -p 1489 def-repl-manager-pwd-file:pwd.txt
Enter "cn=Directory Manager" password: d1rm4n4ger
```

5 **Remove the `pwd.txt` file.**

6 **Log out of the DirectoryServer–1 host machine.**

7 **As a root user, log in to the DirectoryServer–2 host machine.**

8 **Create a temporary file that contains the new replication manager password.**

This file is read once, and the password is stored for future use.

```
# cd /var/opt/mps/serverroot/ds6/bin
# echo replm4n4ger > pwd.txt
```

9 **Verify that the file was successfully created.**

```
# cat pwd.txt

replm4n4ger
```

10 **Run** `dsconf set-server-prop` **to set the replication manager password using** `pwd.txt` **as input.**

```
# ./dsconf set-server-prop -h DirectoryServer-2.example.com
  -p 1489 def-repl-manager-pwd-file:pwd.txt
Enter "cn=Directory Manager" password: d1rm4n4ger
```

11 **Remove the** `pwd.txt` **file.**

12 **Log out of the DirectoryServer–2 host machine.**

▼ To Create Replication Agreements for Each User Data Instance

A *replication agreement* is a set of parameters on a supplier that controls how updates are sent to a given consumer. In this case, we are making the user data instances aware of each other.

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **Run** `dsconf create-repl-agmt` **to create the replication agreement.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf create-repl-agmt -h DirectoryServer-1.example.com
  -p 1489 dc=company,dc=com DirectoryServer-2.example.com:1489
Enter "cn=Directory Manager" password: d1rm4n4ger

Use "dsconf init-repl-dest dc=company,dc=com DirectoryServer-2.example.com:1489"
to start replication of "dc=company,dc=com" data.
```

3 **Run** `dsconf list-repl-agmts` **to verify that the replication agreement was successfully created.**

```
# ./dsconf list-repl-agmts -p 1489
Enter "cn=Directory Manager" password: d1rm4n4ger

dc=company,dc=com DirectoryServer-2.example.com:1489
```

This response indicates that the Directory Server 1 base suffix will be replicated to Directory Server 2.

4 **Log out of the DirectoryServer–1 host machine.**

5 **As a root user, log in to the DirectoryServer–2 host machine.**

6 **Run** `dsconf create-repl-agmt` **to create the replication agreement.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf create-repl-agmt -h DirectoryServer-2.example.com
  -p 1489 dc=company,dc=com DirectoryServer-1.example.com:1489
```

```
Enter "cn=Directory Manager" password: d1rm4n4ger

Use "dsconf init-repl-dest dc=company,dc=com DirectoryServer-1.example.com:1489"
to start replication of "dc=company,dc=com" data.
```

7 **Run `dsconf list-repl-agmts` to verify that the replication agreement was successfully created.**

```
# ./dsconf list-repl-agmts -p 1489
Enter "cn=Directory Manager" password: d1rm4n4ger

dc=company,dc=com DirectoryServer-1.example.com:1489
```

This response indicates that the Directory Server 2 base suffix will be replicated to Directory Server 1.

8 **Log out of the DirectoryServer–2 host machine.**

▼ To Initialize the User Data Instance Replication Agreements

In this procedure, initialize the user data instance on Directory Server 1. The previously created agreements will replicate the data to Directory Server 2.

Note – Initialization is **not** required on both instances when configuring for MMR.

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **Run `dsconf show-repl-agmt-status` to verify that the replication agreements are not yet initialized.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf show-repl-agmt-status -h DirectoryServer-1.example.com
  -p 1489 dc=company,dc=com DirectoryServer-2.example.com:1489
Enter "cn=Directory Manager" password: d1rm4n4ger

Configuration Status        : OK
Authentication Status       : OK
Initialization Status       : NOT OK

Status:                                         : Dest. Not Initialized
```

3 **Run `dsconf init-repl-dest` to initialize the replication agreements.**

```
# ./dsconf init-repl-dest -h DirectoryServer-1.example.com
  -p 1489 dc=company,dc=com DirectoryServer-2.example.com:1489
Enter "cn=Directory Manager" password: d1rm4n4ger
```

```
Sent 1 entries...
Sent 2 entries...
Completed initialization of "DirectoryServer-2.example.com:1489";
May 15, 2007 1:53:32 PM
```

4 **Run `dsconf show-repl-agmt-status` again to verify that the replication agreements are now initialized.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsconf show-repl-agmt-status -h DirectoryServer-1.example.com
  -p 1489 dc=company,dc=com DirectoryServer-2.example.com:1489
Enter "cn=Directory Manager" password: d1rm4n4ger
Configuration Status         : OK
Authentication Status        : OK
Initialization Status        : OK

Status:                                         : Enabled
Last Update Date                        : Jul 12, 2007 8:47:42 PM
```

5 **Log out of the DirectoryServer–1 host machine.**

▼ To Verify that User Data Replication Works Properly

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **Run `ldapmodify` to create a new directory entry.**

```
# ldapmodify -a -h DirectoryServer-1.example.com -p 1489
  -D cn=admin,cn=Administrators,cn=config -w d1rm4n4ger

dn: ou=People,dc=company,dc=com
objectclass: top
objectclass: organizationalUnit
ou: People
description: Container for user entries
```

Hit ENTER to indicate end of input.

```
adding new entry ou=People,dc=company,dc=com
```

Hit Control C to terminate the command.

```
^C
```

This step creates a new organizational unit on Directory Server 1.

3 **After the entry is created, as a root user, log in to the DirectoryServer–2 host machine.**

4 **Run `ldapsearch` on Directory Server 2 to verify that the directory entry was successfully replicated.**

```
# ldapsearch -b "dc=company,dc=com" -p 1489 -D "cn=Directory Manager"
  -w d1rm4n4ger "objectclass=organizationalUnit"

version: 1
dn: ou=People,dc=company,dc=com
objectClass: top
objectClass: organizationalUnit
ou: People
description Container for user entries
```

5 **Now run `ldapdelete` on Directory Server 2 to delete the entry just created.**

```
# ldapdelete -h DirectoryServer-2.example.com -p 1489
  -D "cn=Directory Manager" -w d1rm4n4ger "ou=People,dc=company,dc=com"
```

6 **As a root user on Directory Server 1, run `ldapsearch` to verify that the entry was deleted.**

```
# ldapsearch -b "dc=company,dc=com" -p 1489 -D "cn=Directory Manager"
  -w d1rm4n4ger "objectclass=organizationalUnit"
```

If the delete was successfully replicated to Directory Server 1, the search will return no results.

7 **Log out of the Directory Server host machines.**

5.3 Configuring the Load Balancer for the User Data Instances

Two load balancers are configured for the Directory Server installations. Load Balancer 1 fronts the configuration data instances and Load Balancer 2 fronts the user data instances. In the following procedure, you configure a load balancer in front of the configuration data instances. Figure 4–1 illustrates this architecture.

▼ To Configure User Data Load Balancer 2

- This procedure assumes that you have already installed a load balancer.
- The load balancer hardware and software used in the lab facility for this deployment is BIG-IP® manufactured by F5 Networks. If you are using different load balancer software, see the documentation that comes with that product for detailed settings information.
- Contact your network administrator to obtain an available virtual IP address for the load balancer you want to configure.
- Know the IP address of the load balancer hardware, the URL for the load balancer login page, and a username and password for logging in to the load balancer application.

- Get the IP addresses for Directory Server 1 and Directory Server 2 by running the following command on each host machine:

```
# ifconfig -a
```

1 **Access** `https://is-f5.example.com`**, the BIG-IP load balancer login page, in a web browser.**

2 **Log in using the following information:**

User name:	**`username`**
Password:	**`password`**

3 **Click** *Configure your BIG-IP (R) using the Configuration Utility***.**

4 **Create a Pool.**

A pool contains all the backend server instances.

a. **In the left pane, click Pools.**

b. **On the Pools tab, click Add.**

c. **In the Add Pool dialog, provide the following information:**

Pool Name	**`DirectoryServer-UserData-Pool`**
Load Balancing Method	Round Robin
Resources	Add the IP address and port number of both Directory Server hosts: `DirectoryServer-1:1489` and `DirectoryServer-2:1489`.

d. **Click Done.**

5 **Add a Virtual Server.**

This step defines instances of the load balancer.

Tip – If you encounter JavaScript™ errors or otherwise cannot proceed to create a virtual server, try using Internet Explorer.

---

a. **In the left frame, click Virtual Servers.**

b. **On the Virtual Servers tab, click Add.**

c. **In the Add a Virtual Server dialog box, provide the following information:**

Address	Enter the IP address for `LoadBalancer-2.example.com`
Service	**`489`**
Pool	**`DirectoryServer-UserData-Pool`**

d. **Continue to click Next until you reach the Pool Selection dialog box.**

e. **In the Pool Selection dialog box, assign `DirectoryServer-UserData-Pool` to the virtual server.**

f. **Click Done.**

6 **Add Monitors**

Monitors are required for the load balancer to detect the backend server failures.

a. **In the left frame, click Monitors.**

b. **Click the Basic Associations tab.**

c. **Add an LDAP monitor for the Directory Server 1 node.**

In the Node column, locate the IP address and port number, `DirectoryServer-1:1489`, and select the Add checkbox.

d. **Add an LDAP monitor for the Directory Server 2 node.**

In the Node column, locate the IP address and port number, `DirectoryServer–2:1489`, and select the Add checkbox.

e. **At the top of the Node column, in the drop-down list, choose `ldap-tcp`.**

f. **Click Apply.**

7 **Configure the load balancer for persistence.**

The user data load balancer is configured for *simple persistence*. With simple persistence, all requests sent *within a specified interval* are processed by the same Directory Server instance, ensuring complete replication of entries. For example, when a request requires information to be written to Directory Server 1, that information must also be replicated to Directory Server 2. As the replication takes time to complete, if a related request is directed by the load balancer to Directory Server 2 during the replication process itself, the request may fail as the entry might only be partially created. When properly configured, simple persistence ensures that both requests are routed to Directory Server 1 and processed in consecutive order; the first request is finished before the second request begins processing. Simple persistence ensures that within the specified interval, no errors or delays occur due to replication time or redirects when retrieving data. Simple persistence tracks connections based only on the client IP address.

a. **In the left frame, click Pools.**

b. **Click the name of the pool you want to configure.**

   In this example, `DirectoryServer-UserData-Pool`.

c. **Click the Persistence tab.**

d. **Under Persistence Type, select Simple.**

e. **Enter 300 seconds for the Timeout interval.**

f. **Click Apply.**

8 **Verify the Directory Server load balancer configuration.**

a. **Log in as a root user to the host machine of each Directory Server instance.**

b. **On each Directory Server host machine, use the `tail` command to monitor the Directory Server access log.**

```
# cd /var/opt/mps/am-users/logs
# tail -f access
```

   You should see connections to the load balancer IP address opening and closing. For example:

```
[12/Oct/2006:13:10:20-0700] conn=54 op=-1 msgId=-1 —
fd=22 slot=22 LDAP connection from IP_address to IP_address
[12/Oct/2006:13:10:20-0700] conn=54 op=-1 msgId=-1 — closing — B1
[12/Oct/2006:13:10:20-0700] conn=54 op=-1 msgId=-1 — closed.
```

c. **Execute the following LDAP search against the Directory Server load balancer.**

```
# ldapsearch -h LoadBalancer-2.example.com -p 489 -b "dc=company,dc=com"
  -D "cn=directory manager" -w d1rm4n4ger "(objectclass=*)"
```

   The `ldapsearch` operation should return entries. Make sure they display in the access log on only one Directory Server.

d. **Run `dsadm stop` to stop Directory Server 1.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm stop /var/opt/mps/am-users
```

e. **Again perform the following LDAP search against the Directory Server load balancer.**

```
# ldapsearch -h LoadBalancer-2.example.com -p 489 -b "dc=company,dc=com"
  -D "cn=directory manager" -w d1rm4n4ger "(objectclass=*)"
```

The `ldapsearch` operation should return entries. Verify that the entries display in the access log on only Directory Server 2.

Note – You may encounter the following error message:

```
ldap_simple_bind: Cant' connect to the LDAP
server — Connection refused
```

This means that the load balancer may not fully detect that Directory Server 1 is stopped. In this case, you may have started the search too soon based on the polling interval setting. For example, if the polling interval is set to 10 seconds, you should wait ten seconds to start the search. You can reset the timeout properties to a lower value using the following procedure.

a. **Click the Monitors tab.**

b. **Click the ldap-tcp monitor name.**

c. **In the Interval field, set the value to 5.**

This tells the load balancer to poll the server every 5 seconds.

d. **In the Timeout field, set the value to 16.**

e. **Click Apply and repeat the LDAP search.**

See your load balancer documentation for more information on the timeout property.

---

f. **Start Directory Server 1.**

```
# ./dsadm start /var/opt/mps/am-users
```

g. **Stop Directory Server 2.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm stop /var/opt/mps/am-users
```

h. **Perform the following LDAP search against the Directory Server load balancer to confirm that the request is forwarded to the running Directory Server 1.**

```
# ldapsearch -h LoadBalancer-2.example.com -p 489 -b "dc=company,dc=com"
  -D "cn=Directory Manager" - w d1rm4n4ger "(objectclass=*)"
```

The `ldapsearch` operation should return entries. Make sure the entries display in the access log on only Directory Server 1.

i. **Start Directory Server 2.**

```
# ./dsadm start /var/opt/mps/am-users
```

j. Log out of both Directory Server host machines and the load balancer console.

CHAPTER 6

Installing and Configuring Access Manager

This chapter contains instructions on how to install and configure Sun Java™ System Access Manager. It begins with installation of the two web containers into which the Access Manager WAR will be deployed. It also contains instructions for other post-installation procedures, including the configuration of the Access Manager load balancer. It contains the following sections:

- "6.1 Installing the Access Manager Web Containers" on page 79
- "6.2 Deploying and Configuring Access Manager 1 and Access Manager 2" on page 88
- "6.3 Configuring the Access Manager Load Balancer" on page 100
- "6.4 Configuring the Access Manager Platform Service" on page 111
- "6.5 Reconfiguring Access Manager to Communicate with Directory Server" on page 114

6.1 Installing the Access Manager Web Containers

In this section, we will create a non-root user with the `roleadd` command in the Solaris Operating Environment, and install Sun Java System Web Server using the non-root user on each Access Manager host machine. Use the following as your checklist for completing these tasks.

1. "To Create a Non-Root User on the Access Manager 1 Host Machine" on page 80
2. "To Install Sun Java System Web Server for Access Manager 1" on page 81
3. "To Create a Non-Root User on the Access Manager 2 Host Machine" on page 84
4. "To Install Sun Java System Web Server for Access Manager 2" on page 85

Note – Web Server can also be installed with a root user.

▼ To Create a Non-Root User on the Access Manager 1 Host Machine

1. **As a root user, log in to the AccessManager–1 host machine.**

2. **Use `roleadd` to create a new user.**

```
# roleadd -s /sbin/sh -m -g staff -d /export/am71adm am71adm
```

Note – We chose to use `roleadd` rather than `useradd` for security reasons as `roleadd` disables the ability of the user to log in.

---

3. **(Optional) Verify that the user was created.**

```
# cat /etc/passwd

root:x:0:0:Super-User:/:/sbin/sh
daemon:x:1:1::/:
...
nobody4:x:65534:SunOS 4.x NFS Anonymous Access User:/:
am71adm:x:215933:10::/export/am71adm:/sbin/sh
```

4. **(Optional) Verify that the user's directory was created.**

```
# cd /export/am71adm
# ls

local.cshrc    local.profile    local.login
```

5. **Create a password for the non-root user.**

```
# passwd am71adm
New Password: 4m71a6m
Re-ener new Pasword: 4m71a6m

passwd: password successfully changed for am71adm
```

⚠ **Caution –** If you do not perform this step, you will not be able to *switch user* (`su`) when logged in as the non-root user.

▼ To Install Sun Java System Web Server for Access Manager 1

Before You Begin This procedure assumes you have just completed "To Create a Non-Root User on the Access Manager 1 Host Machine" on page 80.

1. **On the AccessManager-1 host machine, install required patches if necessary.**

```
# patchadd -p | grep 117461-08
```

A list of patch numbers is displayed. On our lab machine, the required patch 117461-08 is present so there is no need to install patches at this time.

Note – Results for your machines might be different. Read the latest version of the Web Server 7.0 Release Notes to determine if you need to install patches and, if so, what they might be. You can search for patches directly at `http://sunsolve.sun.com` by navigating to the PatchFinder page, entering the patch number and clicking Find Patch.

---

2. **Create a directory into which the Web Server bits can be downloaded and change into it.**

```
# mkdir /export/WS7
# cd /export/WS7
```

3. **Download the Sun Java System Web Server 7.0 software from `http://www.sun.com/download/products.xml?id=45ad781d`.**

Follow the instructions on the Sun Microsystems Product Downloads web site for downloading the software.

4. **Unpack the software package.**

```
# gunzip sjsws-7_0-solaris-sparc.tar.gz
# tar xvf sjsws-7_0-solaris-sparc.tar
```

5. **Run `setup`.**

```
# ./setup --console
```

6. **When prompted, provide the following information.**

You are running the installation program for the Sun Java System Web Server 7.0. ... The installation program pauses as questions are presented so you can read the information and make your choice. When you are ready to continue, press Enter.	Press Enter. Continue to press Enter when prompted.
Have you read the Software License Agreement and do you accept all the terms?	Enter **yes**.
Sun Java System Web Server 7.0 Installation Directory [/sun/webserver7]	Enter **/opt/SUNWwbsvr**
Specified directory /opt/SUNWwbsvr does not exist. Create Directory? [Yes/No]	Enter **yes**.
Select Type of Installation 1. Express 2. Custom 3. Exit What would you like to do? [1]	Enter **2**.
Component Selection 1. Server Core 2. Server Core 64-biy Binaries 3. Administration Command Line Interface 4. Sample Applications 5. Language Pack Enter the comma-separated list [1,2,3,4,5]	Enter **1,3,5**.
Java Configuration 1. Install Java Standard Edition 1.5.0_09 2. Reuse existing Java SE 1.5.0_09 or greater 3. Exit What would you like to do? [1]	Enter **1**.
Administrative Options 1. Create an Administration Server and a Web Server Instance 2. Create an Administration Node Enter your option. [1]	Enter **1**.
Start servers during system startup. [yes/no]	Enter **no**.
Host Name [AccessManager-1.example.com]	Accept the default value.
SSL Port [8989]	Accept the default value.
Create a non-SSL Port? [yes/no]	Enter **no**.

`Runtime User ID [root]`	Enter **`am71adm`**.
`Administrator User Name [admin]`	Accept the default value.
`Administrator Password:`	Enter **`web4dmin`**.
`Retype Password:`	Enter **`web4dmin`**.
`Server Name [AccessManager-1.example.com]`	Accept the default value.
`Http Port [8080]`	Enter **`1080`**.

`Document Root Directory [/opt/SUNWwbsvr/`
`https-AccessManager-1.example.com/docs]`
	Accept the default value.

`Ready To Install`
`1. Install Now`
`2. Start Over`
`3. Exit Installation`
`What would you like to do?`
	Enter **`1`**.

When installation is complete, the following message is displayed:

```
Installation Successful.
```

7 **To verify that Web Server was installed with the non-root user, examine the permissions.**

```
# cd /opt/SUNWwbsvr/admin-server/
# ls -al

total 16
drwxr-xr-x   8 root     root         512 Jul 19 10:36 .
drwxr-xr-x  11 am71adm  staff        512 Jul 19 10:36 ..
drwxr-xr-x   2 root     root         512 Jul 19 10:36 bin
drwx------   2 am71adm  staff        512 Jul 19 10:36 config
drwx------   3 am71adm  staff        512 Jul 19 11:09 config-store
drwx------   3 am71adm  staff        512 Jul 19 10:40 generated
drwxr-xr-x   2 am71adm  staff        512 Jul 19 10:40 logs
drwx------   2 am71adm  staff        512 Jul 19 10:36 sessions
```

The appropriate files and directories are owned by `am71adm`.

8 **Start the Web Server 1 administration server.**

```
# su am71adm
# cd /opt/SUNWwbsvr/admin-server/bin
# ./startserv
```

9 **Verify that the non-root user was able to start Web Server with the following sub-procedure.**

a. **Access from** `https://AccessManager-1.example.com:8989` **a web browser.**

b. **Log in to the Web Server console as `admin`.**

User Name: **`admin`**

Password: **`web4dmin`**

The Web Server administration console opens, verifying that the non-root user was able to start Web Server.

c. **Exit the console and close the browser.**

10 **Log out of the AccessManager–1 host machine.**

▼ To Create a Non-Root User on the Access Manager 2 Host Machine

1 **As a root user, log in to the AccessManager–2 host machine.**

2 **Use `roleadd` to create a new user.**

```
# roleadd -s /sbin/sh -m -g staff -d /export/am71adm am71adm
```

3 **(Optional) Verify that the user was created.**

```
# cat /etc/passwd

root:x:0:0:Super-User:/:/sbin/sh
daemon:x:1:1::/:
...
nobody4:x:65534:SunOS 4.x NFS Anonymous Access User:/:
am71adm:x:215933:10::/export/am71adm:/sbin/sh
```

4 **(Optional) Verify that the user's directory was created.**

```
# cd /export/am71adm
# ls

local.cshrc    local.profile    local.login
```

5 **Create a password for the non-root user.**

```
# passwd am71adm
New Password: 4m71a6m
Re-ener new Pasword: 4m71a6m

passwd: password successfully changed for am71adm
```

Caution – If you do not perform this step, you will not be able to *switch user* (su) when logged in as the non-root user.

▼ To Install Sun Java System Web Server for Access Manager 2

Before You Begin This procedure assumes that you just completed "To Create a Non-Root User on the Access Manager 2 Host Machine" on page 84.

1 **On the AccessManager-2 host machine, install required patches if necessary.**

```
# patchadd -p | grep 117461-08
```

A list of patch numbers is displayed. On our lab machine, the required patch 117461-08 is present so there is no need to install patches at this time.

Note – Results for your machines might be different. Read the latest version of the Web Server 7.0 Release Notes to determine if you need to install patches and, if so, what they might be. You can search for patches directly at `http://sunsolve.sun.com` by navigating to the PatchFinder page, entering the patch number and clicking Find Patch.

2 **Create a directory into which the Web Server bits can be downloaded and change into it.**

```
# mkdir /export/WS7
# cd /export/WS7
```

3 **Download the Sun Java System Web Server 7.0 software from** `http://www.sun.com/download/products.xml?id=45ad781d`**.**

Follow the instructions on the Sun Microsystems Product Downloads web site for downloading the software.

4 **Unpack the software package.**

```
# gunzip sjsws-7_0-solaris-sparc.tar.gz
# tar xvf sjsws-7_0-solaris-sparc.tar
```

5 **Run** `setup`**.**

```
# ./setup --console
```

6 **When prompted, provide the following information.**

`You are running the installation program`
`for the Sun Java System Web Server 7.0.`
`...`
`The installation program pauses as questions`
`are presented so you can read the`
`information and make your choice.`
`When you are ready to continue, press Enter.`
	Press Enter. Continue to press Enter when prompted.

`Have you read the Software License`
`Agreement and do you accept all the terms?`
	Enter **yes**.

`Sun Java System Web Server 7.0`
`Installation Directory [/sun/webserver7]`
	Enter **/opt/SUNWwbsvr**

`Specified directory /opt/SUNWwbsvr`
`does not exist.  Create Directory? [Yes/No]`
	Enter **yes**.

`Select Type of Installation`
 	
`1. Express`
`2. Custom`
`3. Exit`
`What would you like to do? [1]`
	Enter **2**.

`Component Selection`
 	
`1. Server Core`
`2. Server Core 64-biy Binaries`
`3. Administration Command Line Interface`
`4. Sample Applications`
`5. Language Pack`
`Enter the comma-separated list [1,2,3,4,5]`
	Enter **1,3,5**.

`Java Configuration`
`1. Install Java Standard Edition 1.5.0_09`
`2. Reuse existing Java SE 1.5.0_09 or greater`
`3. Exit`
`What would you like to do? [1]`
	Enter **1**.

`Administrative Options`
`1. Create an Administration Server and a`
`   Web Server Instance`
`2. Create an Administration Node`
`Enter your option. [1]`
	Enter **1**.

`Start servers during system startup. [yes/no]`	Enter **no**.
`Host Name [AccessManager-2.example.com]`	Accept the default value.
`SSL Port [8989]`	Accept the default value.
`Create a non-SSL Port? [yes/no]`	Enter **no**.

`Runtime User ID [root]`	Enter **am71adm**.
`Administrator User Name [admin]`	Accept the default value.
`Administrator Password:`	Enter **web4dmin**.
`Retype Password:`	Enter **web4dmin**.
`Server Name [AccessManager-2.example.com]`	Accept the default value.
`Http Port [8080]`	Enter **1080**.

`Document Root Directory [/opt/SUNWwbsvr/`
`https-AccessManager-2.example.com/docs]`
	Accept the default value.

`Ready To Install`
`1. Install Now`
`2. Start Over`
`3. Exit Installation`
`What would you like to do?`
	Enter **1**.

When installation is complete, the following message is displayed:

```
Installation Successful.
```

7 To verify that Web Server was installed with the non-root user, examine the permissions.

```
# cd /opt/SUNWwbsvr/admin-server/
# ls -al

total 16
drwxr-xr-x   8 root     root         512 Jul 19 10:36 .
drwxr-xr-x  11 am71adm  staff        512 Jul 19 10:36 ..
drwxr-xr-x   2 root     root         512 Jul 19 10:36 bin
drwx------   2 am71adm  staff        512 Jul 19 10:36 config
drwx------   3 am71adm  staff        512 Jul 19 11:09 config-store
drwx------   3 am71adm  staff        512 Jul 19 10:40 generated
drwxr-xr-x   2 am71adm  staff        512 Jul 19 10:40 logs
drwx------   2 am71adm  staff        512 Jul 19 10:36 sessions
```

The appropriate files and directories are owned by `am71adm`.

8 Start the Web Server 2 administration server.

```
# su am71adm
# cd /opt/SUNWwbsvr/admin-server/bin
# ./startserv
```

9 Verify that the non-root user was able to start Web Server with the following sub-procedure.

a. Access `https://AccessManager-2.example.com:8989` from a web browser.

b. **Log in to the Web Server console as `admin`.**

User Name: **`admin`**

Password: **`web4dmin`**

The Web Server administration console opens, verifying that the non-root user was able to start Web Server.

c. **Exit the console and close the browser.**

10 **Log out of the AccessManager–2 host machine.**

6.2 Deploying and Configuring Access Manager 1 and Access Manager 2

An Access Manager WAR will be deployed in the installed Web Server containers on both the Access Manager host machines. Additionally, you will configure the installations and back up the Access Manager configuration data. Use the following list of procedures as a checklist for completing the tasks.

1. "To Generate an Access Manager WAR File on the Access Manager 1 Host Machine" on page 88
2. "To Deploy the Access Manager WAR File as Access Manager 1" on page 90
3. "To Copy the Access Manager WAR File to Access Manager 2" on page 92
4. "To Deploy the Access Manager WAR File as Access Manager 2" on page 92
5. "To Configure Access Manager 1" on page 94
6. "To Configure Access Manager 2" on page 96
7. "To Back Up the Access Manager Configuration Data from Directory Server 1" on page 99

▼ To Generate an Access Manager WAR File on the Access Manager 1 Host Machine

1 **As a root user, log in to the AccessManager–1 host machine.**

2 **Create a directory into which the Access Manager WAR file can be downloaded and change into it.**

```
# mkdir /export/AM71
# cd /export/AM71
```

3 **Download the Access Manager 7.1 WAR file from** http://www.sun.com/download/products.xml?id=460d5c8e**.**

4 **Unzip the Access Manager download.**

```
# unzip AccessManager7_1RTM.zip
# ls -al

total 228716
drwxr-xr-x   6 root     root          512 Jul 11 20:00 .
drwxr-xr-x   5 root     sys           512 Jul 19 10:30 ..
-rw-r--r--   1 root     root    117008919 Jul 10 15:00 AccessManager7_1RTM.zip
drwxr-xr-x   4 root     root          512 Jun 25 20:16 applications
drwxr-xr-x   2 root     root         1536 Jun 25 20:16 legal
-rw-r--r--   1 root     root         3018 Jun 25 20:16 README
drwxr-xr-x   2 root     root          512 Jun 25 20:16 samples
-r--r--r--   1 root     root        11934 Jun 25 20:16 Software_License_Agt_SLA.txt
drwxr-xr-x   2 root     root          512 Jun 25 20:16 tools
```

5 **Switch to the non-root user.**

```
# su am71adm
```

6 **Create a staging area in which the WAR will be exploded.**

```
# cd /export/am71adm
# mkdir am-staging
```

Tip – In the staging area, after exploding the WAR, you can modify the WAR contents to suit your needs, generate a new WAR, and deploy it on any number of remote host computers. Whenever you need to make changes to the WAR, you maintain the changes in this one staging area, and redeploy the modified WAR as many times as you want, on as many host machines as you need.

7 **Explode the WAR file.**

```
# cd am-staging
# jar xvf /export/AM71/applications/jdk15/amserver.war
```

8 **Add the following context parameter to the** `web.xml` **file.**

By default, during the WAR deployment, Access Manager creates a bootstrap file in the user's home directory. The bootstrap file points to the directory where all the Access Manager configurations will be created. By specifying this new context parameter, Access Manager will create the bootstrap file in the directory you specify; in this case, `/export/am71adm/bootstrap`. `web.xml` is located in `/export/am71adm/am-staging/WEB-INF/`.

```
<context-param>
<param-name>com.sun.identity.bootClassPath</param-name>
```

```
<param-value>/export/am71adm/bootstrap</param-value>
</context-param>
```

9 **Regenerate the Access Manager WAR file.**

```
# cd /export/am71adm/am-staging
# jar cvf ../amserver.war *
```

A new WAR file is created, including the modified `web.xml`.

10 **Verify that the new WAR file was created in the proper location and with the appropriate permissions.**

```
# cd /export/am71adm
# ls -al

total 62262
drwxr-xr-x   6 am71adm  staff        512 Jul 19 11:46 .
drwxr-xr-x   5 root     sys          512 Jul 19 10:30 ..
-rw-r--r--   1 am71adm  staff        144 Jul 19 10:30 .profile
drwx------   3 am71adm  staff        512 Jul 19 10:40 .sunw
-rw-r--r--   1 am71adm  staff        566 Jul 19 11:06 .wadmtruststore
drwxr-xr-x  16 am71adm  staff        512 Jul 19 10:47 am-staging
-rw-r--r--   1 am71adm  staff   31834862 Jul 19 10:56 amserver.war
-rw-r--r--   1 am71adm  staff        136 Jul 19 10:30 local.cshrc
-rw-r--r--   1 am71adm  staff        157 Jul 19 10:30 local.login
-rw-r--r--   1 am71adm  staff        174 Jul 19 10:30 local.profile
```

Note – The `amserver.war` file is owned by `am71adm`.

To Deploy the Access Manager WAR File as Access Manager 1

Before You Begin This procedure assumes you have just completed "To Generate an Access Manager WAR File on the Access Manager 1 Host Machine" on page 88.

1 **On the AccessManager-1 host machine, start the Web Server administration server.**

```
# cd /opt/SUNWwbsvr/admin-server/bin
# ./startserv
```

2 **Change to the non-root user `am71adm`.**

```
# cd /opt/SUNWwbsvr/bin
# su am71adm
```

3 **Start the Web Server `AccessManager-1` instance.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-1.example.com/bin
# ./startserv
```

4 **Run `wadm add-webapp` to add the Access Manager WAR file to the Web Server.**

```
# ./wadm add-webapp --user=admin --host=AccessManager-1.example.com
  --port=8989 --config=AccessManager-1.example.com
  --vs=AccessManager-1.example.com
  --uri=/amserver /export/am71adm/amserver.war

Please enter admin-user-password> web4dmin
...
Do you trust the above certificate? [yes/no] yes

CLI201 Command ’add-webapp’ ran successfully.
```

5 **Run `wadm deploy-config` to deploy the Access Manager WAR file.**

```
# ./wadm deploy-config --user=admin --host=AccessManager-1.example.com
  --port=8989 AccessManager-1.example.com

Please enter admin-user-password> web4dmin

CLI201 Command ’deploy-config’ ran successfully.
```

6 **To verify that the Access Manager WAR file was successfully deployed, list the contents of the Web Server instance directory.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-1.example.com/
  web-app/AccessManager-1.example.com
# ls -al

total 6
drwxr-xr-x   3 am71adm  staff        512 Jul 19 11:08 .
drwxr-xr-x   3 am71adm  staff        512 Jul 19 11:08 ..
drwxr-xr-x  16 am71adm  staff        512 Jul 19 11:09 amserver
```

`amserver` exists in the directory and is owned by the non-root user `am71adm`.

7 **Restart the Web Server instance.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-1.example.com/bin
# ./stopserv; ./startserv
```

8 **Log out of the AccessManager–1 host machine.**

▼ To Copy the Access Manager WAR File to Access Manager 2

Before You Begin This procedure assumes you have completed "To Generate an Access Manager WAR File on the Access Manager 1 Host Machine" on page 88.

1. **As a root user, log in to the AccessManager–2 host machine.**

2. **Change to the non-root user `am71adm`.**

   ```
   # su am71adm
   ```

3. **Change into the `am71adm` directory.**

   ```
   # cd /export/am71adm
   ```

4. **Copy `amserver.war` from the AccessManager–1 host machine to the `am71adm` directory.**

5. **Verify that the WAR file was copied into the proper location and with the appropriate permissions.**

   ```
   # ls -al

   total 62260
   drwxr-xr-x   5 am71adm  staff        512 Jul 19 12:10 .
   drwxr-xr-x   6 root     sys          512 Jul 19 11:53 ..
   -rw-r--r--   1 am71adm  staff        144 Jul 19 11:53 .profile
   drwx------   3 am71adm  staff        512 Jul 19 11:57 .sunw
   -rw-r--r--   1 am71adm  staff        566 Jul 19 12:05 .wadmtruststore
   -rw-r--r--   1 am71adm  staff   31834862 Jul 19 12:01 amserver.war
   -rw-r--r--   1 am71adm  staff        136 Jul 19 11:53 local.cshrc
   -rw-r--r--   1 am71adm  staff        157 Jul 19 11:53 local.login
   -rw-r--r--   1 am71adm  staff        174 Jul 19 11:53 local.profile
   ```

   The `amserver.war` files are owned by `am71adm`.

▼ To Deploy the Access Manager WAR File as Access Manager 2

Before You Begin This procedure assumes you have just completed "To Copy the Access Manager WAR File to Access Manager 2" on page 92.

1. **On the AccessManager-2 host machine, start the Web Server administration server.**

   ```
   # cd /opt/SUNWwbsvr/admin-server/bin
   # ./startserv
   ```

2 Change to the non-root user am71adm.

```
# cd /opt/SUNWwbsvr/bin
# su am71adm
```

3 Start the Web Server AccessManager-2 instance.

```
# cd /opt/SUNWwbsvr/https-AccessManager-2.example.com/bin
# ./startserv
```

4 Run wadm add-webapp to add the Access Manager WAR file to the Web Server container.

```
# ./wadm add-webapp --user=admin --host=AccessManager-2.example.com
  --port=8989 --config=AccessManager-2.example.com
  --vs=AccessManager-2.example.com
  --uri=/amserver /export/am71adm/amserver.war

Please enter admin-user-password> web4dmin
...
Do you trust the above certificate? [yes/no] yes

CLI201 Command ’add-webapp’ ran successfully.
```

5 Run wadm deploy-config to deploy the Access Manager WAR file.

```
# ./wadm deploy-config --user=admin --host=AccessManager-2.example.com
  --port=8989 AccessManager-2.example.com

Please enter admin-user-password> web4dmin

CLI201 Command ’deploy-config’ ran successfully.
```

6 To verify that the Access Manager WAR file was successfully deployed, list the contents of the Web Server instance directory.

```
# cd /opt/SUNWwbsvr/https-AccessManager-2.example.com/
  web-app/AccessManager-2.example.com
# ls -al

total 6
drwxr-xr-x   3 am71adm  staff        512 Jul 19 12:07 .
drwxr-xr-x   3 am71adm  staff        512 Jul 19 12:07 ..
drwxr-xr-x  16 am71adm  staff        512 Jul 19 12:07 amserver
```

amserver exists in the directory and is owned by the non-root user am71adm.

7 Restart the Web Server instance.

```
# cd /opt/SUNWwbsvr/https-AccessManager-2.example.com/bin
# ./stopserv; ./startserv
```

8 **Log out of the AccessManager–2 host machine.**

▼ To Configure Access Manager 1

Before You Begin The encryption key used in this procedure must be identical to the encryption key used in the procedure "To Configure Access Manager 2" on page 96. You should therefore save the encryption key from this procedure for easy access when you are configuring Access Manager 2.

Note – This constraint is particular to this deployment example only.

1 **Access** `http://AccessManager-1.example.com:1080/amserver` **from a web browser.**

The Access Manager Configurator page is displayed for first time access.

2 **Provide the following information on the Configurator page.**

Administrator: Password

`4m4dmin1`

Administrator: Retype Password:

`4m4dmin1`

General Settings: Configuration Directory:

`/export/am71adm/config`

General Settings: Encryption Key

The value is **`PXXdT8Sf+ubQwxUhB+/R37LVBrJFYNnhR`**.

Tip – Copy the value from this field, and save it for use in "To Configure Access Manager 2" on page 96.

Configuration Store Settings: Type:

Choose Directory Server.

Caution – It is a common mistake to accept the default value here. Be sure to choose Directory Server.

Server Settings: Name:

`LoadBalancer-1.example.com`

Server Settings: Port:

`389`

Server Settings: Suffix to store configuration data:
`dc=example,dc=com`

Directory Server Administrator: Directory Administrator DN:
`cn=Directory Manager`

Directory Server Administrator: Password:
`d1rm4n4ger`

Directory Server Administrator: Retype Password:
`d1rm4n4ger`

Load User Management Schema:
Click the box to mark it.

3 **Click Configure.**

When configuration is complete, you are redirected to the Access Manager login page.

4 **Log in to the Access Manager console as the administrator.**

User Name: **`amadmin`**

Password: **`4m4dmin1`**

If authentication succeeds, Access Manager has successfully accessed the Directory Server load balancer. You should see the `example` realm in the Realm page.

5 **Log out of the Access Manager console.**

6 **(Optional) To verify that the Access Manager schema was successfully loaded into the configuration data instance on the DirectoryServer–1 host machine do the following.**

a. **As a root user, log in to the DirectoryServer–1 host machine.**

b. **Run `ldapsearch`.**

```
# ldapsearch -p 1389 -b "dc=example,dc=com" -D "cn=Directory Manager"
  -w d1rm4n4ger "(objectclass=*)"
```

You should see a number of entries for Access Manager administrators and special users.

c. **Log out of the DirectoryServer–1 host machine.**

7 **(Optional) To verify that the `config` directory and the supporting `bootstrap` directory have been created with the proper permissions, do the following.**

a. **As a root user, log in to the AccessManager–1 host machine.**

b. **Examine the file system.**

```
# cd /export/am71adm
# ls -al

total 62262
drwxr-xr-x   6 am71adm  staff        512 Jul 19 11:46 .
drwxr-xr-x   5 root     sys          512 Jul 19 10:30 ..
-rw-r--r--   1 am71adm  staff        144 Jul 19 10:30 .profile
drwx------   3 am71adm  staff        512 Jul 19 10:40 .sunw
-rw-r--r--   1 am71adm  staff        566 Jul 19 11:06 .wadmtruststore
drwxr-xr-x  16 am71adm  staff        512 Jul 19 10:47 am-staging
-rw-r--r--   1 am71adm  staff   31834862 Jul 19 10:56 amserver.war
drwxr-xr-x   3 am71adm  staff        512 Jul 19 11:46 bootstrap
drwxr-xr-x   3 am71adm  staff        512 Jul 19 11:46 config
-rw-r--r--   1 am71adm  staff        136 Jul 19 10:30 local.cshrc
-rw-r--r--   1 am71adm  staff        157 Jul 19 10:30 local.login
-rw-r--r--   1 am71adm  staff        174 Jul 19 10:30 local.profile
```

The `config` directory and the `bootstrap` directory were created, and are owned by non-root user `am71adm`.

c. **Log out of the AccessManager–1 host machine.**

Troubleshooting If you cannot login successfully, try the fully qualified name for the user `amadmin`. If you can authenticate using the fully qualified name, you can focus on issues other than authentication and login. In the `/export/am71adm/config/AMConfig.properties` file, the value of `com.sun.identity.authentication.super.user` is the fully qualified name for `amadmin`; in this example, `uid=amAdmin,ou=People,dc=example,dc=com`.

▼ To Configure Access Manager 2

The encryption key used in this procedure must be identical to the encryption key used in the procedure "To Configure Access Manager 1" on page 94. If you did not save the encryption key, it can be found as the value of the `am.encryption.pwd` property in the `/export/am71adm/config/AMConfig.properties` file on the Access Manager 1 host machine.

Note – This constraint is particular to this deployment example only.

---

1 **Access `http://AccessManager-2.example.com:1080/amserver` from a web browser.**

The Access Manager Configurator page is displayed for first time access.

2 **Provide the following information on the Configurator page.**

Administrator: Password
`4m4dmin1`

Administrator: Retype Password:
`4m4dmin1`

General Settings: Configuration Directory:
`/export/am71adm/config`

General Settings: Encryption Key:
`PXXdT8Sf+ubQwxUhB+/R37LVBrJFYNnhR`

Caution – Be sure this value is copied from Access Manager 1. See "To Configure Access Manager 1" on page 94.

Configuration Store Settings: Type:
Choose Directory Server.

Caution – It is a common mistake to accept the default value here. Be sure to choose Directory Server.

Server Settings: Name:
`LoadBalancer-1.example.com`

Server Settings: Port:
`389`

Server Settings: Suffix to store configuration data:
`dc=example,dc=com`

Directory Server Administrator: Directory Administrator DN:
`cn=Directory Manager`

Directory Server Administrator: Password:
`d1rm4n4ger`

Directory Server Administrator: Retype Password:
`d1rm4n4ger`

Load User Management Schema:

Caution – Do not mark the box with a check. The user management schema was loaded into Directory Server when you configured Access Manager 1.

3 **Click Configure.**

When configuration is complete, you are redirected to the Access Manager login page.

4 **Log in to the Access Manager console as the administrator.**

User Name: **amadmin**

Password: **4m4dmin1**

If authentication succeeds, Access Manager has successfully accessed the Directory Server load balancer. You should see the example realm in the Realm page.

5 **Click the `example` realm name.**

You should see three values in the Realms/DNS Aliases List.

- `accessmanager-1.example.com`
- `accessmanager-2.example.com`
- `example`

6 **Log out of the Access Manager console.**

7 **(Optional) To verify that the `config` directory and the supporting `bootstrap` directory have been created with the proper permissions, do the following.**

a. **As a root user, log in to the AccessManager–2 host machine.**

b. **Examine the file system.**

```
# cd /export/am71adm
# ls -al

total 62262
drwxr-xr-x   6 am71adm  staff        512 Jul 19 11:46 .
drwxr-xr-x   5 root     sys          512 Jul 19 10:30 ..
-rw-r--r--   1 am71adm  staff        144 Jul 19 10:30 .profile
drwx------   3 am71adm  staff        512 Jul 19 10:40 .sunw
-rw-r--r--   1 am71adm  staff        566 Jul 19 11:06 .wadmtruststore
-rw-r--r--   1 am71adm  staff   31834862 Jul 19 10:56 amserver.war
drwxr-xr-x   3 am71adm  staff        512 Jul 19 11:46 bootstrap
drwxr-xr-x   3 am71adm  staff        512 Jul 19 11:46 config
-rw-r--r--   1 am71adm  staff        136 Jul 19 10:30 local.cshrc
-rw-r--r--   1 am71adm  staff        157 Jul 19 10:30 local.login
-rw-r--r--   1 am71adm  staff        174 Jul 19 10:30 local.profile
```

`amserver.war` and the `bootstrap` and `config` files are all in this directory, and owned by non-root user `am71adm`.

c. **Log out of the AccessManager–2 host machine.**

Troubleshooting If you cannot login successfully, try the fully qualified name for the user `amadmin`. If you can authenticate using the fully qualified name, you can focus on issues other than authentication and login. In the `/export/am71adm/config/AMConfig.properties` file, the value of `com.sun.identity.authentication.super.user` is the fully qualified name for `amadmin`; in this example, `uid=amAdmin,ou=People,dc=example,dc=com`.

To Back Up the Access Manager Configuration Data from Directory Server 1

Backing up your Access Manager configuration data ensures that if you run into problems later, you can revert to this configuration without having to reinstall Access Manager. In this procedure, we will back up the configuration data from Directory Server 1.

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **Stop the configuration data instance on Directory Server 1.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm stop /var/opt/mps/am-config

Server stopped
```

Note – The backup utility `db2ldif` can only be used if the `slapd` process has been shutdown.

3 **Change to the `am-config` directory.**

```
# cd /var/opt/mps/am-config
```

4 **Run `db2ldif` from within the `am-config` directory.**

```
# ./db2ldif -n dbExample

ldiffile: /var/opt/mps/am-config/ldif/2007_06_27_132405.ldif
[27/Jun/2007:13:24:06 -0700] - export dbExample:
Processed n entries (100%).
```

5 **(Optional) Create a `README` that describes the contents of the new LDIF file.**

```
# cd /var/opt/mps/am-config/ldif
# ls

2007_06_27_132405.ldif

# cat > README
```

Hit ENTER and type the following:

```
2007_06_27_132405.ldif: backup after post-am install, pre-patch application
```

Hit Control D to terminate the cat command

```
^D
```

```
# ls
```

```
2007_06_27_132405.ldif  README
```

6 **Start the configuration data instance on Directory Server 1.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm start /var/opt/mps/am-config
```

7 **Log out of the DirectoryServer–1 host machine.**

6.3 Configuring the Access Manager Load Balancer

The Access Manager servers are fronted by one load balancer (Load Balancer 3). Users internal to the company will access the servers through the non-secure port 7070. Users external to the company will access the servers through the secure port 9443. Additionally, configuring two load balancer instances enables you to customize internal-facing and external-facing login pages. Users external to your company first access the Distributed Authentication User Interface which, in turn, routes the request to the secure port 9443. The following figure illustrates this architecture.

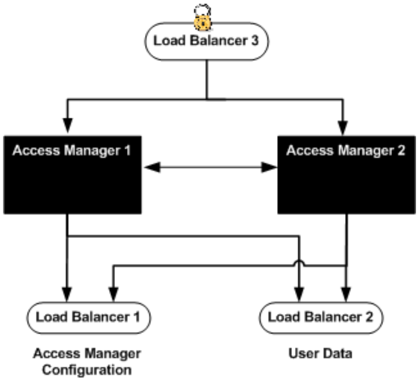


FIGURE 6–1 Load Balancer 3 Fronts Two Access Manager Servers

Load Balancer 3 sends the user and agent requests to the server where the session originated. Secure Sockets Layer (SSL) is terminated before a request is forwarded to the Access Manager servers to allow the load balancer to inspect the traffic for proper routing. Load Balancer 3 is capable of the following types of load balancing:

Cookie-based

The load balancer makes decisions based on client's cookies. The load balancer looks at the request and detects the presence of a cookie by a specific name. If the cookie is detected in the request, the load balancer routes the request to the specific server to which the cookie has been assigned. If the cookie is not detected in the request, the load balancer balances client requests among the available servers.

IP-based

This is similar to cookie-based load balancing, but the decision is based on the IP address of the client. The load balancer sends all requests from a specific IP address to the same server.

TCP

The load balancer mainstreams session affinity. This means that all requests related to a TCP session, are forwarded to the same server. In this deployment example, Load Balancer 3 forwards all requests from a single client to exactly the same server. When the session is started and maintained by one client, session affinity is guaranteed. This type of load-balancing is applicable to the TCP-based protocols.

Use the following list of procedures as a checklist for configuring the Access Manager load balancer. The first procedure tests the Directory Server load balancing and system failover configurations.

1. "To Verify Successful Directory Server Load Balancing and System Failover for Access Manager 1 and Access Manager 2" on page 101
2. "To Configure the Access Manager Load Balancer" on page 103
3. "To Request an Secure Sockets Layer Certificate for the Access Manager Load Balancer" on page 107
4. "To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer" on page 108
5. "To Install an SSL Certificate on the Access Manager Load Balancer" on page 109
6. "To Create an SSL Proxy for SSL Termination on the Access Manager Load Balancer" on page 110

▼ To Verify Successful Directory Server Load Balancing and System Failover for Access Manager 1 and Access Manager 2

Perform the following steps to confirm that Access Manager directory requests are directed to only one instance of Directory Server, and that system failover and recovery work properly. The steps in this procedure are specific to Access Manager 1. Substitute `http://AccessManager-2.example.com:1080/amserver/console` where appropriate to perform this procedure for Access Manager 2.

1 **Confirm that the load balancer is properly configured for simple persistence.**

a. **As a root user, log in to the DirectoryServer–1 and the DirectoryServer–2 host machines.**

b. **On each server, use the `tail` command to watch the Directory Server access log.**

```
# cd /var/opt/mps/am-config/logs
# tail-f logs/access
```

c. **Access `http://AccessManager-1.example.com:1080/amserver/console` from a web browser and log in to the Access Manager 1 console as the default administrator.**

Username	**amadmin**
Password	**4m4dmin1**

d. **Navigate inside the Access Manager 1 console while paying attention to the Directory Server access logs.**

You should see all directory accesses are directed to one Directory Server instance only, excluding the health check probing from the load balancer device. The navigation should not have any errors.

e. **Log out of the Access Manager 1 console and close the browser when successful.**

2 **Confirm that Directory Server failover is working properly.**

a. **Stop Directory Server 1 instance.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm stop /var/opt/mps/am-config

Server stopped
```

b. **Access `http://AccessManager-1.example.com:1080/amserver/console` from a web browser and log in to the Access Manager 1 console as the default administrator.**

Username	**amadmin**
Password	**4m4dmin1**

c. **Navigate inside the Access Manager 1 console while paying attention to the Directory Server access logs.**

You should see all directory accesses are directed to Directory Server 2. The navigation should not have any errors.

d. **Log out and close the browser when successful.**

e. **Start the Directory Server 1 instance.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm start /var/opt/mps/am-config

Server started
```

f. **Stop Directory Server 2 instance.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm stop /var/opt/mps/am-config

Server stopped
```

g. **Access** `http://AccessManager-1.example.com:1080/amserver/console` **from a web browser and log in as the administrator, if necessary.**

Username	**amadmin**
Password	**4m4dmin1**

h. **Navigate inside the Access Manager 1 console while paying attention to the Directory Server access logs.**

You should see all directory accesses are directed to Directory Server 1. The navigation should not have any errors.

i. **Log out and close the browser when successful.**

j. **Start the Directory Server 2 instance.**

```
# cd /var/opt/mps/serverroot/ds6/bin
# ./dsadm start /var/opt/mps/am-config

Server started
```

3 **Confirm that both Directory Servers are running and log out of both host machines.**

4 **Repeat this procedure for Access Manager 2.**

Substitute `http://AccessManager-2.example.com:1080/amserver/console` where applicable and perform these steps again.

▼ To Configure the Access Manager Load Balancer

Before You Begin

- This procedure assumes that you have already installed a load balancer.

- The load balancer hardware and software used for this deployment is BIG-IP® manufactured by F5 Networks. If you are using different load balancer software, see the documentation that comes with that product for detailed settings information.
- Contact your network administrator to obtain two available virtual IP addresses.
- Know the IP address of the load balancer hardware, the URL for the load balancer login page, and a username and password for logging in to the load balancer application.
- Get the IP addresses for Access Manager 1 and Access Manager 2 by running the following command on each host machine:

```
# ifconfig -a
```

1 **Access** `https://is-f5.example.com`**, the BIG-IP load balancer login page, in a web browser.**

2 **Log in using the following information:**

User name:	**`username`**
Password:	**`password`**

3 **Click** *Configure your BIG-IP (R) using the Configuration Utility***.**

4 **Create a Pool.**

A pool contains all the backend server instances.

a. **In the left pane, click Pools.**

b. **On the Pools tab, click Add.**

c. **In the Add Pool dialog, provide the following information.**

Pool Name	**`AccessManager-Pool`**
Load Balancing Method	Round Robin
Resources	Add the IP addresses and port numbers for the Access Manager servers: `AccessManager-1:1080` and `AccessManager-2:1080`.

d. **Click Done.**

5 **Add a Virtual Server for the non-secure port 7070 on the Access Manager Load Balancer 3.**

This step defines instances of the load balancer.

Note – If you encounter JavaScript™ errors or otherwise cannot proceed to create a virtual server, try using Internet Explorer.

a. **In the left frame, click Virtual Servers.**

b. **On the Virtual Servers tab, click Add.**

c. **In the Add a Virtual Server dialog box, provide the following information:**

Address	Enter the IP address for `LoadBalancer-3.example.com`
Service	**`7070`**
Pool	**`AccessManager-Pool`**

d. **Continue to click Next until you reach the Pool Selection dialog box.**

e. **In the Pool Selection dialog box, assign the `AccessManager-Pool` Pool.**

f. **Click Done.**

6 **Add Monitors.**

Access Manager comes with a JSP file named `isAlive.jsp` that can be contacted to determine if the server is down. In the following steps, you create a custom monitor that periodically accesses the JSP. If a success response can be obtained, it means not only that Access Manager is responding to TCP connection request, but also that free threads exist to process the request.

a. **Click the Monitors tab**

b. **Click Add and provide the following information.**

Name:	**`AccessManager-http`**
Inherits From:	Choose `http`.

c. **Click Next on the Configure Basic Properties page.**

d. **Enter the following value in the Send String field of the *Configure ECV HTTP Monitor* dialog.**

```
GET /amserver/isAlive.jsp
```

e. **On the Destination Address and Service (Alias) page, click Done.**

The monitor you entered is now added to the list of monitors.

f. **Click the Basic Associations tab.**

g. **Find the IP address for `AccessManager-1:1080` and `AccessManager-2:1080`.**

h. **Mark the Add checkbox for `AccessManager-1` and `AccessManager-2`.**

i. **At the top of the Node column, choose the monitor that you just added,** `AccessManager-http`**.**

j. **Click Apply.**

7 **Configure the load balancer for persistence.**

a. **In the left pane, click Pools.**

b. **Click the name of the pool you want to configure.**

c. **Click the Persistence tab.**

d. **Under Persistence Type, select Cookie Hash and set the following values.**

In this type of persistence, the load balancer uses a portion of the cookie as a hash ID.

Cookie Name:	**amlbcookie**
Offset:	**1**
Length:	**1**

e. **Click Apply.**

8 **Log out of the load balancer console.**

9 **Verify that the Access Manager load balancer is configured properly.**

a. **As a root user, log in to the AccessManager–1 host machine.**

b. **Run** `tail` **to view the access log.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-1.example.com/logs
# tail -f access
```

If you see frequent entries similar to the one below, the custom monitor is configured properly.

```
IP_address--[12/Oct/2006:13:10:20-0700]
"GET /amserver/isAlive.jsp" 200 118
```

If you do not see "GET `/amserver/isAlive.jsp`", you must troubleshoot the load balancer configuration.

c. **As a root user, log in to the AccessManager–2 host machine.**

d. **Run `tail` to view the access log.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-2.example.com/logs
# tail -f access
```

If you see frequent entries similar to the one below, the custom monitor is configured properly.

```
IP_address--[12/Oct/2006:13:10:20-0700]
"GET /amserver/isAlive.jsp" 200 118
```

If you do not see "`GET /amserver/isAlive.jsp`", you must troubleshoot the load balancer configuration.

e. **Access `http://LoadBalancer-3.example.com:7070/`, the internal-facing load balancer, in a web browser.**

Caution – Do not supply the `amserver` prefix.

If the browser displays the default Sun Java System Web Server document root page, it is configured properly.

f. **Log out of both Access Manager host machines.**

▼ To Request an Secure Sockets Layer Certificate for the Access Manager Load Balancer

Generate a request for a Secure Sockets Layer (SSL) certificate to send to a certificate authority.

1 **Access `https://is-f5.example.com`, the BIG-IP load balancer login page, in a web browser.**

2 **Log in to the BIG-IP console using the following information.**

Username	**`username`**
Password	**`password`**

3 **Click** *Configure your BIG-IP (R) using the Configuration Utility*.

4 **In the left pane, click Proxies.**

5 **Click the Cert-Admin tab.**

6 **On the SSL Certificate Administration page, click** *Generate New Key Pair/Certificate Request*.

7 **In the Create Certificate Request page, provide the following information.**

Key Identifier:	`LoadBalancer-3.example.com`
Organizational Unit Name:	`Deployment`
Domain Name:	`LoadBalancer-3.example.com`
Challenge Password:	`password`
Retype Password:	`password`

8 **Click** ***Generate Key Pair/Certificate Request*****.**

On the SSL Certificate Request page, the request is generated in the Certificate Request field.

9 **Save the text contained in the Certificate Request field to a text file.**

10 **Log out of the console and close the browser.**

11 **Send the certificate request text you saved to the Certificate Authority of your choice.**

A Certificate Authority (CA) is an entity that issues certified digital certificates; VeriSign, Thawte, Entrust, and GoDaddy are just a few. In this deployment, CA certificates were obtained from OpenSSL. Follow the instructions provided by your Certificate Authority to submit a certificate request.

▼ To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer

The CA root certificate proves that the particular CA (such as VeriSign or Entrust) did, in fact, issue a particular SSL certificate. You install the root certificate on Load Balancer 3 to ensure that a link between the Load Balancer 3 SSL certificate can be maintained with the issuing company. CA root certificates are publicly available.

Before You Begin You should have a CA root certificate.

1 **Access** `https://is-f5.example.com`**, the BIG-IP load balancer login page, in a web browser.**

2 **Log in with the following information.**

User name:	`username`
Password:	`password`

3 **In the BIG-IP load balancer console, click Proxies.**

4 **Click the Cert-Admin tab.**

5 **Click Import.**

6 **In the Import Type field, choose Certificate, and click Continue.**

7 **Click Browse in the Certificate File field on the Install SSL Certificate page.**

8 **In the Choose File dialog, choose Browser.**

9 **Navigate to the file that includes the root CA certificate and click Open.**

10 **In the Certificate Identifier field, enter `OpenSSL_CA_cert`.**

11 **Click Install Certificate.**

12 **On the Certificate OpenSSL_CA_Cert page, click Return to Certificate Administration.**

The root certificate OpenSSL_CA_Cert is now included in the Certificate ID list.

▼ To Install an SSL Certificate on the Access Manager Load Balancer

Before You Begin This procedure assumes you have received an SSL certificate from a CA and just completed "To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer" on page 108.

1 **In the BIG-IP load balancer console, click Proxies.**

2 **Click the Cert-Admin tab.**

The key `LoadBalancer-3.example.com` is in the Key List. This was generated in "To Request an Secure Sockets Layer Certificate for the Access Manager Load Balancer" on page 107.

3 **In the Certificate ID column, click Install for `LoadBalancer-3.example.com`.**

4 **In the Certificate File field, click Browse.**

5 **In the Choose File dialog, navigate to the file that contains the certificate text sent to you by the CA and click Open.**

6 **Click Install Certificate.**

7 **On the Certificate LoadBalancer-3.example.com page, click Return to Certificate Administration Information.**

Verify that the Certificate ID indicates `LoadBalancer-3.example.com` on the SSL Certificate Administration page.

8 **Log out of the load balancer console.**

▼ To Create an SSL Proxy for SSL Termination on the Access Manager Load Balancer

Secure Socket Layer (SSL) termination at Load Balancer 3 increases performance on the Access Manager level, and simplifies SSL certificate management. Because Load Balancer 3 sends unencrypted data to the Access Manager server, it does not have to perform decryption, and the burden on its processor is relieved. Clients send SSL-encrypted data to Load Balancer 3 which, in turn, decrypts the data and sends the unencrypted data to the appropriate Access Manager server. Load Balancer 3 also encrypts responses from the Access Manager server, and sends these encrypted responses back to the client. Towards this end, you create an *SSL proxy*, the gateway for decrypting HTTP requests and encrypting the reply.

Note – SSL communication is terminated at Load Balancer 3 before a request is forwarded to the Access Manager servers.

Before You Begin Before creating the SSL proxy, you should have a certificate issued by a recognized CA.

1 **Access** `https://is-f5.example.com`**, the BIG-IP load balancer login page, in a web browser.**

2 **Log in with the following information.**

User name: **`username`**

Password: **`password`**

3 **Click** *Configure your BIG-IP (R) using the Configuration Utility***.**

4 **In the left pane, click Proxies.**

5 **Under the Proxies tab, click Add.**

6 **In the Add Proxy dialog, provide the following information.**

Proxy Type:	Check the SSL checkbox.
Proxy Address:	The IP address of Load Balancer 3.

Proxy Service:	**`9443`** The secure port number
Destination Address:	The IP address of Load Balancer 3.
Destination Service:	**`7070`** The non-secure port number
Destination Target:	Choose **`Local Virtual Server`**.
SSL Certificate:	Choose **`LoadBalancer-3.example.com`**.
SSL Key:	Choose **`LoadBalancer-3.example.com`**.
Enable ARP:	Check this checkbox.

7 **Click Next.**

8 **In the Rewrite Redirects field, choose `Matching`.**

9 **Click Done.**

The new proxy server is added to the Proxy Server list.

10 **Log out of the load balancer console.**

11 **Access `https://LoadBalancer-3.example.com:9443/index.html` from a web browser.**

If the Web Server index page is displayed, you can access the Web Server using the new proxy server port number and the load balancer is configured properly.

Tip – A message may be displayed indicating that the browser doesn't recognize the certificate issuer. If this happens, install the CA root certificate in the browser so that the browser recognizes the certificate issuer. See your browser's online help system for information on installing a root CA certificate.

---

12 **Close the browser.**

6.4 Configuring the Access Manager Platform Service

Access Manager 7.1 features the Platform Service which provides centralized configuration management for an Access Manager deployment. In this procedure, you configure the two Access Manager servers to work as a single unit. Once configured as a *site*, all client requests go through either the internal or external load balancer. Use the following list of procedures as a checklist for completing this task.

1. "To Create an Access Manager Site on Access Manager 1" on page 112
2. "To Verify that the Access Manager Site was Configured Properly" on page 114

▼ To Create an Access Manager Site on Access Manager 1

It is **not** necessary to repeat this procedure on Access Manager 2.

1 Access `http://AccessManager-1.example.com:1080/amserver/console` in a web browser.

2 Log in to the Access Manager console as the administrator.

Username	**`amadmin`**
Password	**`4m4dmin1`**

3 Under the Access Control tab, click `example`, the top-level Realm Name.

4 Enter `LoadBalancer-3.example.com`, the name of the internal load balancer, in the Realm/DNS Aliases field and click Add.

Caution – Do not remove the host names `AccessManager-1` and `AccessManager-2` from the alias list. These allow administrators to log in to the console directly in the event of a load balancer failure.

5 Enter `loadbalancer-3.example.com`, a second entry for the same host name in all lowercase, and click Add.

Caution – The Access Manager site will not be configured properly unless you use all lowercase when entering this second host name. This is a known issue.

6 Click Save.

7 Click Back to Realms.

8 Click the Configuration tab.

9 Under System Properties, click Platform.

10 Under Site Name, click New, and enter the following values for the external load balancer.

Server:	**`https://loadbalancer-3.example.com:9443`**
Site Name:	**`11`**

11 **Click OK.**

12 **Click Save**

13 **Under Site Name, click New again, and enter the following values for the internal load balancer.**

Server:	`http://loadbalancer-3.example.com:7070`
Site Name:	`12`

14 **Click OK.**

15 **Click Save**

16 **On the same Platform page, under Instance Name, click** `AccessManager-1.example.com:1080`**.**

Change the site ID to `01|11|12`

17 **Click OK.**

18 **Click Save**

19 **On the Platform page again, under Instance Name, click** `AccessManager-2.example.com:1080`**.**

Change the site ID to `02|11|12`

20 **Click OK.**

21 **Click Save**

22 **Log out of the Access Manager console.**

23 **Log in to the AccessManager–1 host machine and restart Access Manager for the changes to take effect.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-1.example.com/bin
# ./stopserv; ./startserv
```

24 **Log in to the AccessManager–2 host machine and restart Access Manager for the changes to take effect.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-2.example.com/bin
# ./stopserv; ./startserv
```

25 **Log out of both Access Manager host machines.**

▼ To Verify that the Access Manager Site was Configured Properly

1. **Access the internal load balancer at** `http://LoadBalancer-3.example.com:7070/amserver/UI/Login`.

   If an error message is displayed indicating that the browser cannot connect to either `AccessManager- 1.example.com` or `AccessManager-2.example.com`, the site configuration is not correct. If the site configuration is correct, all browser interactions will occur as expected.

   Note – If you have an issue accessing the Access Manager load balancer, read about reference number 6472662 in Appendix G, "Known Issues and Limitations."

   ---

2. **When the Access Manager login page is displayed, verify that the browser URL still contains the Site URL for the internal load balancer.**

   If it does not contain the Site URL, the site configuration is incorrect. If the site configuration is correct, all browser interactions will occur through the Site URL.

3. **Log in to the Access Manager console as the administrator.**

   User Name: **`amadmin`**

   Password: **`4m4dmin1`**

   A successful login occurs when the site configuration is correct.

4. **Log out of the Access Manager console.**

6.5 Reconfiguring Access Manager to Communicate with Directory Server

After Access Manager is deployed, any agent profiles and users created are stored in a flat file, by default. In "To Configure Access Manager 1" on page 94 and "To Configure Access Manager 2" on page 96, we used a Directory Server instance previously created that we can now use to store these agent profiles and users for the root realm. In this procedure, we reconfigure the Access Manager root realm to communicate with this configuration directory instance, `am-config`, allowing the agent profiles to authenticate successfully through the load balancer against either Access Manager server.

Caution – In an environment with more than one Access Manager server configured behind a load balancer, this procedure is required to use a centralized data store rather than the default flat file.

▼ To Reconfigure an Access Manager Realm to Retrieve Data from the Directory Server Configuration Data Instance

1. **Log in to the Access Manager console as the administrator.**

   User Name: **`amadmin`**

   Password: **`4m4dmin1`**

2. **Under the Access Control tab, click `example`, the top-level Realm Name.**

3. **Click the Data Stores tab to configure the Directory Server installation as the Access Manager Repository.**

   a. **Click New.**

   b. **Type `amConfigDS` in the Name field.**

   c. **Select the Access Manager Repository radio button and click Next.**

      This selection points to the Directory Server chosen during Access Manager configuration.

   d. **Keep the default values and click Finish.**

4. **Under the Data Stores tab, select the default Flat Files Repository and click Delete.**

5. **Click Back to Realms.**

6. **Log out of the Access Manager console.**

CHAPTER 7

Configuring an Access Manager Realm for User Authentication

This chapter contains instructions to create user entries to be used for testing and to import them into the replicated Directory Server user data instances. Additionally, we configure a sub-realm for the users and create an authentication chain for the sub-realm using Access Manager. It contains the following sections.

- "7.1 Importing Test Users into User Data Instance" on page 117
- "7.2 Creating and Configuring a Realm for Test Users" on page 121

7.1 Importing Test Users into User Data Instance

You create user entries in the Directory Server user data instance for the following users:

- `testuser1`
- `testuser2`

They will be used to verify that the policy agent is configured and working properly. Additionally, the `Groups` container will be used for the same purpose. This user data is imported into one Directory Server as it will be replicated to the other instance.

Note – If you are using an existing user data store, create the appropriate users in it and move on to "7.2 Creating and Configuring a Realm for Test Users" on page 121.

▼ To Import the Test Users Data into Directory Server 1

Create an LDIF file with user entries that is imported into Directory Server 1.

1 **As a root user, log in to the DirectoryServer–1 host machine.**

2 **Create an LDIF file with the following entries.**

```
dn: ou=users,dc=company,dc=com
objectclass: top
objectclass: organizationalUnit
ou: users
description: Container for user entries

dn: ou=Groups,dc=company,dc=com
objectClass: top
objectClass: organizationalUnit
ou: Groups
description: Container for group entries

dn: uid=testuser1,ou=users,dc=company,dc=com
uid: testuser1
givenName: Test
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetadmin
objectClass: inetorgperson
objectClass: inetUser
sn: User1
cn: Test User1
userPassword: password
inetUserStatus: Active

dn: uid=testuser2,ou=users,dc=company,dc=com
uid: testuser2
givenName: Test
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
objectClass: inetUser
sn: User2
cn: Test User2
userPassword: password
inetUserStatus: Active
```

3 **Save the file as `am-users.ldif` in the `/tmp` directory.**

4 **Import the LDIF file into Directory Server 1 using `ldapmodify`.**

```
# ldapmodify -h DirectoryServer-1.example.com -p 1489
  -D "cn=Directory Manager" -w d1rm4n4ger -a -f /tmp/am-users.ldif

adding new entry ou=users,dc=company,dc=com
adding new entry ou=Groups,dc=company,dc=com

adding new entry uid=testuser1,ou=users,dc=company,dc=com

adding new entry uid=testuser2,ou=users,dc=company,dc=com
```

5 **Verify that the new users were imported using `ldapsearch`.**

```
# ldapsearch -h DirectoryServer-1.example.com
  -b "dc=company,dc=com" -p 1489 -D "cn=Directory Manager"
  -w d1rm4n4ger "uid=test*"

version: 1
dn: uid=testuser1,ou=users,dc=company,dc=com
uid: testuser1
givenName: Test
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetadmin
objectClass: inetorgperson
objectClass: inetUser
sn: User1
cn: Test User1
userPassword: {SSHA}H5LpB+QLZMoL9SiXzY/DokHKXRclELVy7w25AA==
inetUserStatus: Active

dn: uid=testuser2,ou=users,dc=company,dc=com
uid: testuser2
givenName: Test
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
objectClass: inetUser
sn: User2
cn: Test User2
userPassword: {SSHA}aLNFCQ1qw78KpJeloVZJAAa5QSAPf/9c2mxCQQ==
inetUserStatus: Active
```

6 **Log out of the DirectoryServer–1 host machine.**

7 **(Optional) Verify that the entries were replicated to Directory Server 2 by logging in as a root user to the DirectoryServer–2 host machine and using `ldapsearch`.**

```
# ldapsearch -h DirectoryServer-2.example.com
  -b "dc=company,dc=com" -p 1489 -D "cn=Directory Manager"
  -w d1rm4n4ger ""
```

```
version: 1
dn: dc=company,dc=com
objectClass: top
objectClass: domain
dc: company

dn: ou=users,dc=company,dc=com
objectClass: top
objectClass: organizationalUnit
ou: users
description: Container for user entries

dn: ou=Groups,dc=company,dc=com
objectClass: top
objectClass: organizationalUnit
objectclass: iplanet-am-managed-group
ou: Groups
description: Container for group entries

dn: uid=testuser1,ou=users,dc=company,dc=com
uid: testuser1
givenName: Test
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetadmin
objectClass: inetorgperson
objectClass: inetUser
sn: User1
cn: Test User1
inetUserStatus: Active
userPassword: {SSHA}H5LpB+QLZMoL9SiXzY/DokHKXRclELVy7w25AA==

dn: uid=testuser2,ou=users,dc=company,dc=com
uid: testuser2
givenName: Test
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetorgperson
objectClass: inetUser
sn: User2
cn: Test User2
inetUserStatus: Active
userPassword: {SSHA}aLNFCQ1qw78KpJeloVZJAAa5QSAPf/9c2mxCQQ==
```

8. **Log out of the DirectoryServer–2 host machine.**

7.2 Creating and Configuring a Realm for Test Users

At this point, the root realm is configured to authenticate against the Directory Server configuration data instances. We use the root realm to authenticate special Access Manager accounts (like `amadmin` and agents) but, we now create a sub-realm to authenticate end users against the Directory Server user data instances. This creates a demarcation between Access Manager configuration and administrative account data, and the end-user profiles.

With the following procedures, we create a sub-realm to which we add our text subjects, and then configure the realm properties to suit the needs of this deployment example.

- "To Create a Realm" on page 121
- "To Change the Default User Data Store and Configure an Authentication Module for the Realm" on page 122
- "To Verify That Access Manager Recognizes the External User Data Store" on page 124
- "To Verify That a Realm Subject Can Successfully Authenticate" on page 125

▼ To Create a Realm

1 **Access `http://LoadBalancer-3.example.com:7070/` from a web browser.**

This is the Access Manager load balancer.

2 **Log in to the Access Manager console as the administrator.**

User Name: **`amadmin`**

Password: **`4m4dmin1`**

3 **Click the Access Control tab.**

4 **Click New to create a new realm.**

5 **On the New Realm page, enter `users` in the Name field.**

6 **Enter `users` in the New Value field of the Realm/DNS Aliases list and click Add.**

7 **Click OK.**

The `users` realm is listed as a sub-realm of the top-level realm, `example`.

▼ To Change the Default User Data Store and Configure an Authentication Module for the Realm

Now we instantiate an authentication module and reconfigure the default `ldapService` authentication chain to use the new authentication module. Additionally, we will change the realm's User Profile attribute and delete the default authentication module instances. During this procedure, we also change the default user data store to the user data instance previously created.

Before You Begin This procedure assumes you have just completed "To Create a Realm" on page 121 and are still logged in to the Access Manager console.

1 **Under the Access Control tab, click the `users` realm.**

2 **Click the Authentication tab.**

3 **Click Advanced Properties in the General section.**

4 **On the resulting page, change the value of the User Profile attribute to `Ignored`.**

This new value specifies that a user profile is not required by the Authentication Service to issue a token after successful authentication.

5 **Click Save.**

The profile is updated.

6 **Click Back to Authentication.**

You will return to the `users` realm Authentication page.

7 **Under Module Instances section, click New.**

These next steps instantiate the Data Store authentication module in the `users` sub-realm.

a. **On the New Module Instance page, set the following attribute values:**

Name:	**`usersDataStore`**
Type:	Choose Data Store

b. **Click OK.**

You will return to the `users` realm Authentication page and the **`usersDataStore`** module is displayed in the list of Module Instances.

8 **Under Authentication Chaining, click on the default `ldapService` chain.**

These next steps reconfigure the default `ldapService` chain to use the new authentication module.

a. **On the resulting page, select `usersDataStore` in the Instance column.**

b. **Set the Criteria attribute to Required.**

c. **Click Save.**

The `ldapService` chain is updated.

d. **Click Back to Authentication.**

You will return to the `users` realm Authentication page.

9 **Under Module Instances, mark the checkbox for LDAP and Data Store.**

These modules are inherited from the default top-level realm and used to authenticate to the Access Manager configuration data instance of Directory Server. They are no longer needed now that the `usersDataStore` authentication module instance will be used.

10 **Click Delete**

The modules are deleted and the `users` realm Authentication page is displayed.

11 **Click Save.**

12 **Click the Data Stores tab.**

a. **Mark the checkbox for `amConfigDS`.**

This is the data store inherited from the parent realm.

b. **Click Delete.**

c. **Click New.**

d. **On the resulting page, set the following attributes:**

Name:	**`usersLDAP`**
Type:	Choose Generic LDAPv3

e. **Click Next.**

f. **On the resulting page, set the following attributes:**

LDAP Server	■ Enter the hostname and port number for the existing directory in the form `LoadBalancer-2.example.com:489` and click Add. ■ Select the default `LoadBalancer-1.example.com:389` and click Remove.
LDAP Bind DN	Enter **`cn=Directory Manager`**
LDAP Bind Password	Enter **`d1rm4n4ger`**
LDAP Bind Password (confirm)	Enter **`d1rm4n4ger`**
LDAP Organization DN	Replace `dc=example,dc=com` with `dc=company,dc=com`
LDAP User Object Classes	Add `inetorgperson` as a new value.
LDAP People Container Value	Replace `people` with **`users`**. **Note –** If this field is empty, the search for user entries will start from the root suffix.
Persistent Search Base DN	Replace `dc=example,dc=com` with `dc=company,dc=com`

g. **Click Finish.**

13 **Log out of the Access Manager console.**

To Verify That Access Manager Recognizes the External User Data Store

1 **Access** `http://AccessManager-1.example.com:1080/amserver/console` **from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

User Name:	**`amadmin`**
Password:	**`4m4dmin1`**

3 **Click on the Access Control tab**

4 **Click on the** `users` **sub-realm.**

5 **Click on the Subjects tab.**

You should see Test User1 and Test User2.

6 **Log out of the Access Manager console.**

▼ To Verify That a Realm Subject Can Successfully Authenticate

1 **Access** `http://AccessManager-1.example.com:1080/amserver/UI/Login?realm=users` **from a web browser.**

The parameter `realm=users` specifies the realm to use for authentication. At this point, a user can log in against Directory Server only if the `realm` parameter is defined in the URL.

2 **Log in to Access Manager with a user name and password from the** `am-users` **directory.**

User Name	**`testuser1`**
Password	**`password`**

You should be able to log in successfully and see a page with a message that reads *You're logged in*. Since the User Profile attribute was set to `Ignored`, the user's profile is not displayed after a successful login. If the login is not successful, watch the Directory Server access log to troubleshoot the problem.

3 **Log out of the Access Manager console.**

CHAPTER 8

Installing and Configuring the Distributed Authentication User Interface

Access Manager provides a remote authentication interface component to enable secure authentication. Installing the Distributed Authentication User Interface on one or more web containers within a non-secure layer eliminates the exposure of service URLs to the end user. This chapter contains the following sections.

8.1 Creating an Agent Profile and Custom User for Distributed Authentication User Interface

Before installing and configuring the Distributed Authentication User Interface, you create an agent profile in Access Manager to be used by the Distributed Authentication User Interface to authenticate itself. An agent profile allows Access Manager to store authentication and configuration information regarding the Distributed Authentication User Interface. The agent profile created in this procedure will be stored in the Access Manager configuration data store.

Creating an agent profile also creates a custom user. This custom user will allow the Distributed Authentication User Interface to log into the Access Manager server and therefore must be defined as an Access Manager special user.

Note – Although the Distributed Authentication User Interface is not an agent, it acts on behalf of Access Manager and therefore must have its own agent profile.

Use the following list of procedures as a checklist for these tasks.

1. "To Create an Agent Profile for the Distributed Authentication User Interface" on page 128
2. "To Define Agent Profile User as an Access Manager Special User" on page 130
3. "To Verify that `authuiadmin` Was Created in Directory Server" on page 129

▼ To Create an Agent Profile for the Distributed Authentication User Interface

This agent profile will be used by the Distributed Authentication User Interface to authenticate itself to Access Manager. The process includes creation of a special user that will be defined as an Access Manager special user in the next procedure, "To Define Agent Profile User as an Access Manager Special User" on page 130.

1 **Access `http://LoadBalancer-3.example.com:7070/` from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

User Name: **`amadmin`**

Password: **`4m4dmin1`**

3 **Under the Access Control tab, click `example`, the top-level Realm Name.**

4 **Click the Subjects tab.**

5 **Click the Agent tab.**

6 **Click New to create a new agent profile.**

7 **Type `authuiadmin` in the ID field.**

8 **Type `4uthu14dmin` in the Password and Password (confirm) fields, respectively.**

9 **Click OK.**

10 **From the list of Agent names, click `authuiadmin`.**

11 **Copy the value of the `UniversalID` and save it to a temporary text file.**

You will need this value in "To Define Agent Profile User as an Access Manager Special User" on page 130.

12 **Log out of the console.**

13 **(Optional) Verify that the `agents` organizational unit was created successfully by logging into a Directory Server host machine and running `ldapsearch`.**

```
# ldapsearch -b "dc=example,dc=com" -h LoadBalancer-1.example.com
  -p 389 -D "cn=Directory Manager" -w d1rm4n4ger "ou=agents"

version: 1
dn: ou=agents,dc=example,dc=com
sunIdentityServerSupportedTypes: agent
ou: agents
objectClass: sunNameSpace
objectClass: iplanet-am-managed-org-unit
objectClass: top
objectClass: organizationalUnit
```

This organization unit will hold all agent profiles.

Note – The `agents` organizational unit is created only after the first agent profile is configured.

▼ To Verify that `authuiadmin` Was Created in Directory Server

This is an optional, verification step.

1 **Log in to either of the Directory Server host machines.**

2 **Run `ldapsearch` to verify that the `authuiadmin` entry was successfully created.**

```
# ldapsearch -b "dc=example,dc=com" -h LoadBalancer-1.example.com
  -p 389 -D "cn=Directory Manager" -w d1rm4n4ger "uid=authuiadmin"

version: 1
dn: uid=authuiadmin,ou=agents,dc=example,dc=com
sunIdentityServerDeviceStatus: Active
uid: authuiadmin
objectClass: sunIdentityServerDevice
objectClass: iplanet-am-user-service
objectClass: top
objectClass: iPlanetPreferences
sunIdentityServerDeviceType: Agent
cn: default
sunIdentityServerDeviceVersion: 2.2
userPassword: {SSHA}aeEi095TamPnJCOLinRNDzlLC8SDaOsdQ2Nqfw==
```

3 **Log out of the Directory Server host machine.**

▼ To Define Agent Profile User as an Access Manager Special User

The agent profile just created includes a user that will now be defined as an Access Manager special administrative user for both Access Manager 1 and Access Manager 2.

Before You Begin You should have the `UniversalID` value saved in "To Create an Agent Profile for the Distributed Authentication User Interface" on page 128.

1. **Define `authuiadmin` as a special user in Access Manager 1.**

   a. **As a root user, log in to the AccessManager–1 host machine.**

   b. **Locate `AMConfig.properties` in the `/export/am71adm/config` directory.**

   Tip – Backup `AMConfig.properties` before you modify it.

   c. **Add the `UniversalID` you saved to the end of the list of values for the `com.sun.identity.authentication.special.users` property in `AMConfig.properties`.**

   You saved `id=authuiadmin,ou=agent,dc=example, dc=com` in "To Create an Agent Profile for the Distributed Authentication User Interface" on page 128.

   Tip – Change `ou=agent` to `ou=agents` and `id` to `uid` before adding it to `AMConfig.properties`.

   d. **Restart the Web Server 1 web container to apply the change.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-1.example.com/bin
# ./stopserv; ./startserv
```

   e. **Log out of the AccessManager–1 host machine.**

2. **Define `authuiadmin` as a special user in Access Manager 2.**

   a. **As a root user, log in to the AccessManager–2 host machine.**

   b. **Locate `AMConfig.properties` in the `/export/am71adm/config` directory.**

   Tip – Backup `AMConfig.properties` before you modify it.

c. **Add the `UniversalID` you saved to the end of the list of values for the `com.sun.identity.authentication.special.users` property in `AMConfig.properties`.**

   You saved `id=authuiadmin,ou=agent,dc=example, dc=com` in "To Create an Agent Profile for the Distributed Authentication User Interface" on page 128.

   Tip – Change `ou=agent` to `ou=agents` and `id` to `uid` before adding it to `AMConfig.properties`.

d. **Restart the Web Server 2 web container to apply the change.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-2.example.com/bin
# ./stopserv; ./startserv
```

e. **Log out of the AccessManager–2 host machine.**

8.2 Installing and Configuring the Distributed Authentication User Interface 1

Use the following list of procedures as a checklist for installing and deploying the Distributed Authentication User Interface 1.

1. "To Create a Non-Root User on the Distributed Authentication User Interface 1 Host Machine" on page 131
2. "To Install Sun Java System Web Server for Distributed Authentication User Interface 1" on page 132
3. "To Configure the WAR for Distributed Authentication User Interface 1" on page 136
4. "To Deploy the Distributed Authentication User Interface 1 WAR" on page 139
5. "To Import the Access Manager Load Balancer Certificate Authority Root Certificate into Distributed Authentication User Interface 1" on page 140
6. "To Verify that Authentication Through the Distributed Authentication User Interface 1 is Successful" on page 142

▼ To Create a Non-Root User on the Distributed Authentication User Interface 1 Host Machine

Create a non-root user with the `roleadd` command in the Solaris Operating Environment on the Distributed Authentication User Interface 1 (AuthenticationUI-1) host machine

1 **As a root user, log in to the AuthenticationUI-1 host machine.**

2 **Use `roleadd` to create a new user.**

```
# roleadd -s /sbin/sh -m -g staff -d /export/da71adm da71adm
```

3 **(Optional) Verify that the user was created.**

```
# cat /etc/passwd

root:x:0:0:Super-User:/:/sbin/sh
daemon:x:1:1::/:
...
nobody4:x:65534:SunOS 4.x NFS Anonymous Access User:/:
da71adm:x:215933:10::/export/da71adm:/sbin/sh
```

4 **(Optional) Verify that the user's directory was created.**

```
# cd /export/da71adm
# ls

local.cshrc    local.profile    local.login
```

5 **(Optional) Create a password for the non-root user.**

```
# passwd da71adm
New Password: 6a714dm
Re-ener new Pasword: 6a714dm

passwd: password successfully changed for da71adm
```

Note – If you do not perform this step, you will not be able to *switch user* (`su`) when logged in as the non-root user.

▼ To Install Sun Java System Web Server for Distributed Authentication User Interface 1

Before You Begin

- This procedure assumes that you have just completed "To Create a Non-Root User on the Distributed Authentication User Interface 1 Host Machine" on page 131.
- Before beginning the installation, read the Web Server 7.0 Release Notes to determine the latest patches you might need to install.

1 **On the AuthenticationUI-1 host machine, install required patches if necessary.**

In this case, the Release Notes indicate that based on the hardware and operating system being used, patch 118855-36 and patch 119964–08 are required.

a. **Run `patchadd` to see if the patches are already installed.**

```
# patchadd -p | grep 118855-36
```

No results are returned which indicates that the patch is not yet installed on the system.

```
# patchadd -p | grep 119964-08
```

No results are returned which indicates that the patch is not yet installed on the system.

b. **Make a directory for downloading the patches you need and change into it.**

```
# mkdir /export/patches
# cd /export/patches
```

c. **Download the patches.**

You can search for patches directly at `http://sunsolve.sun.com`. Navigate to the PatchFinder page, enter the patch number, click Find Patch, and download the appropriate patch.

Note – Signed patches are downloaded as JAR files. Unsigned patches are downloaded as ZIP files.

---

d. **Unzip the patch files.**

```
# unzip 118855–36.zip
# unzip 119964-08.zip
```

e. **Run `patchadd` to install the patches.**

```
# patchadd /export/patches/118855-36
# patchadd /export/patches/119964-08
```

Tip – You can use the `-M` option to install all patches at once. See the `patchadd` man page for more information.

---

f. **After installation is complete, run `patchadd` to verify that each patch was added successfully.**

```
# patchadd -p | grep 118855–36
```

In this example, a series of patch numbers are displayed, and the patch 118855–36 is present.

```
# patchadd -p | grep 119964-08
```

In this example, a series of patch numbers are displayed, and the patch 119964-08 is present.

2. **Create a directory into which you can download the Web Server bits and change into it.**

```
# mkdir /export/WS7
# cd /export/WS7
```

3. **Download the Sun Java System Web Server 7.0 software from** http://www.sun.com/download/products.xml?id=45ad781d**.**

   Follow the instructions on the Sun Microsystems Product Downloads web site for downloading the software.

4. **Unpack the software package.**

```
# gunzip sjsws-7_0-solaris-amd64.tar.gz
# tar xvf sjsws-7_0-solaris-amd64.tar
```

5. **Run `setup`.**

```
# cd /export/WS7
# ./setup --console
```

6. **When prompted, provide the following information.**

You are running the installation program for the Sun Java System Web Server 7.0. ... The installation program pauses as questions are presented so you can read the information and make your choice. When you are ready to continue, press Enter.	Press Enter. Continue to press Enter when prompted.
Have you read the Software License Agreement and do you accept all the terms?	Enter **yes**.
Sun Java System Web Server 7.0 Installation Directory [/sun/webserver7]	Enter **/opt/SUNWwbsvr**
Specified directory /opt/SUNWwbsvr does not exist. Create Directory? [Yes/No]	Enter **yes**.
Select Type of Installation 1. Express 2. Custom 3. Exit What would you like to do? [1]	Enter **2**.

`Component Selection`
 	
`1. Server Core`
`2. Server Core 64-biy Binaries`
`3. Administration Command Line Interface`
`4. Sample Applications`
`5. Language Pack`
`Enter the comma-separated list [1,2,3,4,5]`
	Enter **`1,3,5`**.

`Java Configuration`
`1. Install Java Standard Edition 1.5.0_09`
`2. Reuse existing Java SE 1.5.0_09 or greater`
`3. Exit`
`What would you like to do? [1]`
	Enter **`1`**.

`Administrative Options`
`1. Create an Administration Server and a`
`   Web Server Instance`
`2. Create an Administration Node`
`Enter your option. [1]`
	Enter **`1`**.

`Start servers during system startup. [yes/no]`	Enter **`no`**.
`Host Name [AuthenticationUI-1.example.com]`	Accept the default value.
`SSL Port [8989]`	Accept the default value.
`Create a non-SSL Port? [yes/no]`	Enter **`no`**.
`Runtime User ID [root]`	Enter **`da71adm`**.
`Administrator User Name [admin]`	Accept the default value.
`Administrator Password:`	Enter **`web4dmin`**.
`Retype Password:`	Enter **`web4dmin`**.
`Server Name [AuthenticationUI-1.example.com]`	Accept the default value.
`Http Port [8080]`	Enter **`1080`**.

`Document Root Directory [/opt/SUNWwbsvr/`
`https-AuthenticationUI-1.example.com/docs]`
	Accept the default value.

`Ready To Install`
`1. Install Now`
`2. Start Over`
`3. Exit Installation`
`What would you like to do?`
	Enter **`1`**.

When installation is complete, the following message is displayed:

```
Installation Successful.
```

7 **(Optional) To verify that Web Server was installed with the non-root user, examine the permissions.**

```
# cd /opt/SUNWwbsvr/admin-server
# ls -al

total 16
drwxr-xr-x   8 root     root         512 Jul 19 10:36 .
drwxr-xr-x  11 da71adm  staff        512 Jul 19 10:36 ..
drwxr-xr-x   2 root     root         512 Jul 19 10:36 bin
drwx------   2 da71adm  staff        512 Jul 19 10:36 config
drwx------   3 da71adm  staff        512 Jul 19 11:09 config-store
drwx------   3 da71adm  staff        512 Jul 19 10:40 generated
drwxr-xr-x   2 da71adm  staff        512 Jul 19 10:40 logs
drwx------   2 da71adm  staff        512 Jul 19 10:36 sessions
```

The appropriate files and directories are owned by da71adm.

8 **Start the Web Server administration server.**

```
# su da71adm
# cd /opt/SUNWwbsvr/admin-server/bin
# ./startserv
```

9 **To verify that the non-root user was able to start Web Server, access** https://AuthenticationUI-1.example.com:8989 **from a web browser.**

a. **Log in to the Web Server console as the administrator.**

User Name:	**admin**
Password:	**web4dmin**

The Web Server administration console opens.

b. **Log out of the console and close the browser.**

10 **Log out of the AuthenticationUI–1 host machine.**

▼ To Configure the WAR for Distributed Authentication User Interface 1

This procedure configures the amauthdistui.war that will be used for deployment in "To Deploy the Distributed Authentication User Interface 1 WAR" on page 139.

1 **As a root user, log in to the AuthenticationUI–1 host machine.**

2. **Switch to the non-root user.**

```
# su da71adm
```

3. **Change to the directory into which you will copy `amDistAuth.zip`.**

```
# cd /export/da71adm
```

`amDistAuth.zip` contains the files you need to install the Distributed Authentication User Interface. It is included in the Access Manager software downloaded in “6.2 Deploying and Configuring Access Manager 1 and Access Manager 2” on page 88.

4. **Copy `amDistAuth.zip` from the AccessManager–1 host machine.**

```
# ftp AccessManager-1.example.com

Connected to AccessManager-1.example.com
220 AccessManager-1.example.com FTP server ready.
Name (AccessManager-1.example.com:username):username
Password: ********
...
Using binary mode to transfer files
ftp> cd /export/AM71/applications
CWD command successful
ftp> mget amDistAuth.zip?
mget amDistAuth.zip? y
200 PORT command successful
ftp> bye
```

5. **List the contents of `/export/da71adm` to verify that `amDistAuth.zip` was transferred and is owned by the non-root user.**

```
# ls -al

total 26496
drwxr-xr-x   5 da71adm  staff        512 Jul 19 20:59 .
drwxr-xr-x   7 root     sys          512 Jul 20 10:13 ..
-rw-r--r--   1 da71adm  staff        144 Jul 19 19:53 .profile
drwx------   3 da71adm  staff        512 Jul 19 20:41 .sunw
-rw-r--r--   1 da71adm  staff    6747654 Jul 19 20:43 amDistAuth.zip
```

6. **Unzip `amDistAuth.zip`.**

```
# unzip amDistAuth.zip
```

7. **List the contents again to verify the unzip.**

```
# ls -al

total 26496
drwxr-xr-x   5 da71adm  staff        512 Jul 19 20:59 .
```

```
drwxr-xr-x   7 root     sys          512 Jul 20 10:13 ..
-rw-r--r--   1 da71adm  staff        144 Jul 19 19:53 .profile
drwx------   3 da71adm  staff        512 Jul 19 20:41 .sunw
-rw-r--r--   1 da71adm  staff        572 Jul 19 20:59 .wadmtruststore
-rw-r--r--   1 da71adm  staff    6772566 Jul 19 20:56 amauthdistui.war
-rw-r--r--   1 da71adm  staff    6747654 Jul 19 20:43 amDistAuth.zip
drwxr-xr-x   2 da71adm  staff        512 Jul 19 20:52 lib
-rw-r--r--   1 da71adm  staff        136 Jul 19 19:53 local.cshrc
-rw-r--r--   1 da71adm  staff        157 Jul 19 19:53 local.login
-rw-r--r--   1 da71adm  staff        174 Jul 19 19:53 local.profile
-rw-r--r--   1 da71adm  staff      10038 Mar 19 15:33 README.distAuthUI
-rw-r--r--   1 da71adm  staff       1865 Mar 19 15:31 setup.bat
-rw-r--r--   1 da71adm  staff       1865 Mar 19 15:31 setup.sh
drwxr-xr-x   3 da71adm  staff        512 Jun 25 20:13 WEB-INF
```

8 **Change permissions on `setup.sh`, the Distributed Authentication User Interface configuration script.**

```
# chmod +x setup.sh
```

This gives the non-root user permission to run the script that configures the Distributed Authentication User Interface WAR for its deployment.

9 **Run `setup.sh`.**

```
# ./setup.sh
```

Caution – If using a shell other than `sh`, you must modify the setup script before running it.

a. Open `setup.sh` in a text editor.
b. Add `#!/bin/sh` as the first line of the file.
c. Save and close the file.
d. Run the script.

10 **Provide the following information.**

`Debug directory (make sure this directory exists):`	Enter **`/tmp/distAuth`**
`Application username:`	Enter **`authuiadmin`**
`Application password:`	Enter **`4uthu14dmin`**
`Protocol of the server:`	Enter **`http`**
`Host name of the server:`	Enter **`LoadBalancer-3.example.com`**
`Port of the server:`	Enter **`7070`**

`Server’s deploymen URI:`	Enter **`amserver`**
`Naming URL (hit enter to accept default value, http://LoadBalancer-3.example.com:7070/amserver/namingservice)`	Press Enter to accept the default value.
`Protocol of the distauth server:`	Enter **`http`**
`Host name of the distauth server:`	Enter **`AuthenticationUI-1.example.com`**
`Port of the distaut server:`	Enter **`1080`**
`Distauth Server’s deployment URI:`	Enter **`distAuth`**
`Notification URL (hit enter to accept default value, http://AuthenticationUI-1.example.com:1080/distAuth/notificationservice)`	Press Enter to accept the default value.

After running the script, `amauthdistui.war` is updated with the above values. The next step is “To Deploy the Distributed Authentication User Interface 1 WAR” on page 139.

To Deploy the Distributed Authentication User Interface 1 WAR

Before You Begin This procedure assumes you just completed “To Configure the WAR for Distributed Authentication User Interface 1” on page 136 and are still logged into the AuthenticationUI–1 host machine as the non-root user.

1. **Start the Web Server administration server.**

```
# cd /opt/SUNWwbsvr/admin-server/bin
# ./startserv
```

2. **Add the Distributed Authentication User Interface WAR.**

```
# cd /opt/SUNWwbsvr/bin
# ./wadm add-webapp --user=admin --host=AuthenticationUI-1.example.com
  --port=8989 --config=AuthenticationUI-1.example.com
  --vs=AuthenticationUI-1.example.com
  --uri=/distAuth /export/da71adm/amauthdistui.war

Please enter admin-user-password:web4dmin

Do you trust the above certificate? [y|n] y

CLI201 Command 'add-webapp' ran successfully
```

3 **Deploy the Distributed Authentication User Interface WAR.**

```
# ./wadm deploy-config --user=admin --host=AuthenticationUI-1.example.com
  --port=8989 AuthenticationUI-1.example.com

Please enter admin-user-password: web4dmin

CLI201 Command 'deploy-config' ran successfully
```

4 **Restart the Web Server `AuthenticationUI-1` instance.**

```
# cd /opt/SUNWwbsvr/https-AuthenticationUI-1.example.com/bin
# ./stopserv; ./startserv
```

5 **Verify that the `distAuth` web module is loaded.**

```
# cd /opt/SUNWwbsvr/https-AuthenticationUI-1.example.com/
  web-app/AuthenticationUI-1.example.com
# ls -al

total 6
drwxr-xr-x   3 da71adm  staff        512 Jul 19 21:00 .
drwxr-xr-x   3 da71adm  staff        512 Jul 19 21:00 ..
drwxr-xr-x   8 da71adm  staff        512 Jul 19 21:00 distAuth
```

6 **Log out of the AuthenticationUI–1 host machine.**

▼ To Import the Access Manager Load Balancer Certificate Authority Root Certificate into Distributed Authentication User Interface 1

Import a Certificate Authority (CA) root certificate that enables the Distributed Authentication User Interface to trust the SSL certificate from the Access Manager Load Balancer 3, and establish trust with the certificate chain that is formed from the Certificate Authority to the certificate.

1 **As a root user, log in to the AuthenticationUI–1 host machine.**

2 **Copy the CA root certificate into a directory.**

Use the same root certificate installed in "To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer" on page 108. In this example, the file is `/export/software/ca.cer`.

3 **Import the CA root certificate into the Java keystore.**

```
# /opt/SUNWwbsvr/jdk/jre/bin/keytool -import -trustcacerts
  -alias OpenSSLTestCA -file /export/software/ca.cer
  -keystore /opt/SUNWwbsvr/jdk/jre/lib/security/cacerts
  -storepass changeit

Owner: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
O=Sun,L=Santa Clara, ST=California C=US
Issuer: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
O=Sun,L=Santa Clara, ST=California C=US
Serial number: 97dba0aa26db6386
Valid from: Tue Apr 18 07:66:19 PDT 2006 until: Tue Jan 13 06:55:19
PST 2009
Certificate fingerprints:
                MD5: 9f:57:ED:B2:F2:88:B6:E8:0F:1E:08:72:CF:70:32:06
     SHA1: 31:26:46:15:C5:12:5D:29:46:2A:60:A1:E5:9E:26:64:36:80:E4:70
Trust this certificate: [no] yes
Certificate was added to keystore.
```

4 **Verify that the CA root certificate was imported into the keystore.**

```
# /opt/SUNWwbsvr/jdk/jre/bin/keytool -list
  -keystore /opt/SUNWwbsvr/jdk/jre/lib/security/cacerts
  -storepass changeit | grep -i open

opensslltestca, Nov 8, 2006, trustedCertEntry
```

5 **Restart the Web Server `AuthenticationUI-1` instance.**

```
# cd /opt/SUNWwbsvr/https-AuthenticationUI-1.example.com/bin
# ./stopserv

server has been shutdown

# ./startserv

Sun Java System Web Server 7.0 B12/04/2006 07:59
info: CORE5076: Using [Java HotSpot(TM) Server VM,
Version 1.5.0_09] from [Sun Microsystems Inc.]
info: WEB0100: Loading web module in virtual server
[AuthenticationUI-1.example.com] at [/distAuth]
info: HTTP3072: http-listener-1:
http://AuthenticationUI-1.example.com:1080
ready to accept requests
info: CORE3274: successful server startup
```

6 **Log out of the AuthenticationUI–1 host machine.**

▼ To Verify that Authentication Through the Distributed Authentication User Interface 1 is Successful

Find a host that has direct network connectivity to Distributed Authentication User Interface 1 and the external facing load balancer of the Access Manager servers. One natural place is the AuthenticationUI–1 host machine itself.

1. **As a root user, log into the AuthenticationUI—1 host machine.**

2. **Modify** `AMConfig.properties`**.**

   a. **Change to the** `classes` **directory.**

   ```
   # cd /opt/SUNWwbsvr/https-AuthenticationUI-1.example.com/
     web-app/AuthenticationUI-1.example.com/distAuth/WEB-INF/classes
   ```

   Tip – Backup `AMConfig.properties` before you modify it.

   b. **Set the values of the properties as follows.**

   ```
   com.iplanet.am.naming.url=https://LoadBalancer-3.
     example.com:9443/amserver/namingservice
   com.iplanet.am.server.protocol=https
   com.iplanet.am.server.port=9443
   ```

   c. **Save the file and close it.**

3. **Restart the AuthenticationUI-1 host machine.**

   ```
   # cd /opt/SUNWwbsvr/https-AuthenticationUI-1.example.com/bin
   # ./stopserv; ./startserv
   ```

4. **Access** `http://AuthenticationUI-1.example.com:1080/distAuth/UI/Login?goto=http://LoadBalancer-3.example.com:7070` **from a web browser.**

5. **Log in to the Access Manager console as the administrator.**

Username	**amadmin**
Password	**4m4dmin1**

   After successful authentication, you should be redirected to the index page for the Web Server in which Access Manager is deployed.

6. **Log out of the Access Manager console.**

8.3 Installing and Configuring the Distributed Authentication User Interface 2

Use the following list of procedures as a checklist for installing and configuring the Distributed Authentication User Interface 2.

1. "To Create a Non-Root User on the Distributed Authentication User Interface 2 Host" on page 143
2. "To Install Sun Java System Web Server for Distributed Authentication User Interface 2" on page 144
3. "To Configure the WAR for Distributed Authentication User Interface 2" on page 148
4. "To Deploy the Distributed Authentication User Interface 2 WAR" on page 151
5. "To Import the Access Manager Load Balancer Certificate Authority Root Certificate into the Distributed Authentication User Interface 2" on page 152
6. "To Verify that Authentication Through the Distributed Authentication User Interface 2 is Successful" on page 153

▼ To Create a Non-Root User on the Distributed Authentication User Interface 2 Host

Create a non-root user with the `roleadd` command in the Solaris Operating Environment on the Distributed Authentication User Interface (AuthenticationUI–2) host machine

1 **As a root user, log in to the AuthenticationUI–2 host machine.**

2 **Use `roleadd` to create a new user.**

```
# roleadd -s /sbin/sh -m -g staff -d /export/da71adm da71adm
```

3 **(Optional) Verify that the user was created.**

```
# cat /etc/passwd

root:x:0:0:Super-User:/:/sbin/sh
daemon:x:1:1::/:
...
nobody4:x:65534:SunOS 4.x NFS Anonymous Access User:/:
da71adm:x:215933:10::/export/da71adm:/sbin/sh
```

4 **(Optional) Verify that the user's directory was created.**

```
# cd /export/da71adm
# ls

local.cshrc    local.profile    local.login
```

5 **(Optional) Create a password for the non-root user.**

```
# passwd da71adm
New Password: 6a714dm
Re-ener new Pasword:6a714dm

passwd: password successfully changed for da71adm
```

Note – If you do not perform this step, you will not be able to *switch user* (su) when logged in as the non-root user.

▼ To Install Sun Java System Web Server for Distributed Authentication User Interface 2

Before You Begin

- This procedure assumes that you have just completed "To Create a Non-Root User on the Distributed Authentication User Interface 2 Host" on page 143.
- Before beginning the installation, read the Web Server 7.0 Release Notes to determine the latest patches you might need to install.

1 **On the AuthenticationUI–2 host machine, install required patches if necessary.**

In this case, the Release Notes indicate that based on the hardware and operating system being used, patch 118855-36 and patch 119964–08 are required.

a. **Run `patchadd` to see if the patches are already installed.**

```
# patchadd -p | grep 118855-36
```

No results are returned which indicates that the patch is not yet installed on the system.

```
# patchadd -p | grep 119964-08
```

No results are returned which indicates that the patch is not yet installed on the system.

b. **Make a directory for downloading the patches you need and change into it.**

```
# mkdir /export/patches
# cd /export/patches
```

c. **Download the patches.**

You can search for patches directly at `http://sunsolve.sun.com`. Navigate to the PatchFinder page, enter the patch number, click Find Patch, and download the appropriate patch.

Note – Signed patches are downloaded as JAR files. Unsigned patches are downloaded as ZIP files.

d. **Unzip the patch files.**

```
# unzip 118855–36.zip
# unzip 119964-08.zip
```

e. **Run `patchadd` to install the patches.**

```
# patchadd /export/patches/118855-36
# patchadd /export/patches/119964-08
```

Tip – You can use the `-M` option to install all patches at once. See the `patchadd` man page for more information.

f. **After installation is complete, run `patchadd` to verify that each patch was added successfully.**

```
# patchadd -p | grep 118855–36
```

In this example, a series of patch numbers are displayed, and the patch 118855–36 is present.

```
# patchadd -p | grep 119964-08
```

In this example, a series of patch numbers are displayed, and the patch 119964-08 is present.

2 **Create a directory into which you can download the Web Server bits and change into it.**

```
# mkdir /export/WS7
# cd /export/WS7
```

3 **Download the Sun Java System Web Server 7.0 software from `http://www.sun.com/download/products.xml?id=45ad781d`.**

Follow the instructions on the Sun Microsystems Product Downloads web site for downloading the software.

4 **Unpack the software package.**

```
# gunzip sjsws-7_0-solaris-amd64.tar.gz
# tar xvf sjsws-7_0-solaris-amd64.tar
```

5 **Run `setup`.**

```
# cd /export/WS7
# ./setup --console
```

6 **When prompted, provide the following information.**

`You are running the installation program`
`for the Sun Java System Web Server 7.0.`
`...`
`The installation program pauses as questions`
`are presented so you can read the`
`information and make your choice.`
`When you are ready to continue, press Enter.`
	Press Enter.
	Continue to press Enter when prompted.

`Have you read the Software License`
`Agreement and do you accept all the terms?`
	Enter **yes**.

`Sun Java System Web Server 7.0`
`Installation Directory [/sun/webserver7]`
	Enter **/opt/SUNWwbsvr**

`Specified directory /opt/SUNWwbsvr`
`does not exist.  Create Directory? [Yes/No]`
	Enter **yes**.

`Select Type of Installation`
 	
`1. Express`
`2. Custom`
`3. Exit`
`What would you like to do? [1]`
	Enter **2**.

`Component Selection`
 	
`1. Server Core`
`2. Server Core 64-biy Binaries`
`3. Administration Command Line Interface`
`4. Sample Applications`
`5. Language Pack`
`Enter the comma-separated list [1,2,3,4,5]`
	Enter **1,3,5**.

`Java Configuration`
`1. Install Java Standard Edition 1.5.0_09`
`2. Reuse existing Java SE 1.5.0_09 or greater`
`3. Exit`
`What would you like to do? [1]`
	Enter **1**.

`Administrative Options`
`1. Create an Administration Server and a`
`   Web Server Instance`
`2. Create an Administration Node`
`Enter your option. [1]`
	Enter **1**.

`Start servers during system startup. [yes/no]`	Enter **no**.
`Host Name [AuthenticationUI-2.example.com]`	Accept the default value.
`SSL Port [8989]`	Accept the default value.
`Create a non-SSL Port? [yes/no]`	Enter **no**.

`Runtime User ID [root]`	Enter **`da71adm`**.
`Administrator User Name [admin]`	Accept the default value.
`Administrator Password:`	Enter **`web4dmin`**.
`Retype Password:`	Enter **`web4dmin`**.
`Server Name [AuthenticationUI-2.example.com]`	Accept the default value.
`Http Port [8080]`	Enter **`1080`**.

`Document Root Directory [/opt/SUNWwbsvr/`
`https-AuthenticationUI-2.example.com/docs]`
	Accept the default value.

`Ready To Install`
`1. Install Now`
`2. Start Over`
`3. Exit Installation`
`What would you like to do?`
	Enter **`1`**.

When installation is complete, the following message is displayed:

```
Installation Successful.
```

7 To verify that Web Server was installed with the non-root user, examine the permissions.

```
# cd /opt/SUNWwbsvr/admin-server
# ls -al

total 16
drwxr-xr-x   8 root     root         512 Jul 19 10:36 .
drwxr-xr-x  11 da71adm  staff        512 Jul 19 10:36 ..
drwxr-xr-x   2 root     root         512 Jul 19 10:36 bin
drwx------   2 da71adm  staff        512 Jul 19 10:36 config
drwx------   3 da71adm  staff        512 Jul 19 11:09 config-store
drwx------   3 da71adm  staff        512 Jul 19 10:40 generated
drwxr-xr-x   2 da71adm  staff        512 Jul 19 10:40 logs
drwx------   2 da71adm  staff        512 Jul 19 10:36 sessions
```

The appropriate files and directories are owned by `da71adm`.

8 Start the Web Server administration server.

```
# su da71adm
# cd /opt/SUNWwbsvr/admin-server/bin
# ./startserv
```

9 **To verify that the non-root user was able to start Web Server, access** `https://AuthenticationUI-2.example.com:8989` **from a web browser.**

a. **Log in to the Web Server console as the administrator.**

User Name: **`admin`**

Password: **`web4dmin`**

The Web Server administration console opens.

b. **Log out of the console and close the browser.**

10 **Log out of the AuthenticationUI–2 host machine.**

▼ To Configure the WAR for Distributed Authentication User Interface 2

This procedure configures the `amauthdistui.war` that will be used for deployment in "To Deploy the Distributed Authentication User Interface 2 WAR" on page 151.

1 **As a root user, log in to the AuthenticationUI–2 host machine.**

2 **Switch to the non-root user.**

```
# su da71adm
```

3 **Change to the directory into which you will copy** `amDistAuth.zip`**.**

```
# cd /export/da71adm
```

`amDistAuth.zip` contains the files you need to install the Distributed Authentication User Interface. It is included in the Access Manager software downloaded in "6.2 Deploying and Configuring Access Manager 1 and Access Manager 2" on page 88.

4 **Copy** `amDistAuth.zip` **from the AccessManager–1 host machine.**

```
# cd /export/da71adm
# ftp AccessManager-1.example.com

Connected to AccessManager-1.example.com
220 AccessManager-1.example.com FTP server ready.
Name (AccessManager-1.example.com:username):username
Password: ********
...
Using binary mode to transfer files
ftp> cd /export/AM71/applications
```

```
CWD command successful
ftp> mget amDistAuth.zip?
mget amDistAuth.zip? y
200 PORT command successful
ftp> bye
```

5 **List the contents of `/export/da71adm` to verify that `amDistAuth.zip` was transferred and is owned by the non-root user.**

```
# ls -al

total 26496
drwxr-xr-x   5 da71adm  staff        512 Jul 19 20:59 .
drwxr-xr-x   7 root     sys          512 Jul 20 10:13 ..
-rw-r--r--   1 da71adm  staff        144 Jul 19 19:53 .profile
drwx------   3 da71adm  staff        512 Jul 19 20:41 .sunw
-rw-r--r--   1 da71adm  staff    6747654 Jul 19 20:43 amDistAuth.zip
```

6 **Unzip `amDistAuth.zip`.**

```
# unzip amDistAuth.zip
```

7 **List the contents again to verify the unzip.**

```
# ls -al

total 26496
drwxr-xr-x   5 da71adm  staff        512 Jul 19 20:59 .
drwxr-xr-x   7 root     sys          512 Jul 20 10:13 ..
-rw-r--r--   1 da71adm  staff        144 Jul 19 19:53 .profile
drwx------   3 da71adm  staff        512 Jul 19 20:41 .sunw
-rw-r--r--   1 da71adm  staff        572 Jul 19 20:59 .wadmtruststore
-rw-r--r--   1 da71adm  staff    6772566 Jul 19 20:56 amauthdistui.war
-rw-r--r--   1 da71adm  staff    6747654 Jul 19 20:43 amDistAuth.zip
drwxr-xr-x   2 da71adm  staff        512 Jul 19 20:52 lib
-rw-r--r--   1 da71adm  staff        136 Jul 19 19:53 local.cshrc
-rw-r--r--   1 da71adm  staff        157 Jul 19 19:53 local.login
-rw-r--r--   1 da71adm  staff        174 Jul 19 19:53 local.profile
-rw-r--r--   1 da71adm  staff      10038 Mar 19 15:33 README.distAuthUI
-rw-r--r--   1 da71adm  staff       1865 Mar 19 15:31 setup.bat
-rw-r--r--   1 da71adm  staff       1865 Mar 19 15:31 setup.sh
drwxr-xr-x   3 da71adm  staff        512 Jun 25 20:13 WEB-INF
```

8 **Change permissions on `setup.sh`, the Distributed Authentication User Interface configuration script.**

```
# chmod +x setup.sh
```

This gives the non-root user permission to run the script that configures the Distributed Authentication User Interface WAR for its deployment.

9 Run `setup.sh`.

```
# ./setup.sh
```

Caution – If using a shell other than `sh`, you must modify the setup script before running it.

a. Open `setup.sh` in a text editor.
b. Add `#!/bin/sh` as the first line of the file.
c. Save and close the file.
d. Run the script.

10 Provide the following information.

`Debug directory (make sure this directory exists):`	Enter **`/tmp/distAuth`**
`Application username:`	Enter **`authuiadmin`**
`Application password:`	Enter **`4uthu14dmin`**
`Protocol of the server:`	Enter **`http`**
`Host name of the server:`	Enter **`LoadBalancer-3.example.com`**
`Port of the server:`	Enter **`7070`**
`Server's deploymen URI:`	Enter **`amserver`**
`Naming URL (hit enter to accept default value, http://LoadBalancer-3.example.com:7070/amserver/namingservice)`	Press Enter to accept the default value.
`Protocol of the distauth server:`	Enter **`http`**
`Host name of the distauth server:`	Enter **`AuthenticationUI-2.example.com`**
`Port of the distaut server:`	Enter **`1080`**
`Distauth Server's deployment URI:`	Enter **`distAuth`**
`Notification URL (hit enter to accept default value, http://AuthenticationUI-2.example.com:1080/distAuth/notificationservice)`	Press Enter to accept the default value.

After running the script, `amauthdistui.war` is updated with the above values. The next step is "To Deploy the Distributed Authentication User Interface 2 WAR" on page 151.

To Deploy the Distributed Authentication User Interface 2 WAR

Before You Begin This procedure assumes you just completed "To Configure the WAR for Distributed Authentication User Interface 2" on page 148 and are still logged into the AuthenticationUI–2 host machine as the non-root user.

1 **Start the Web Server administration server.**

```
# cd /opt/SUNWwbsvr/admin-server/bin
# ./startserv
```

2 **Add the Distributed Authentication User Interface WAR.**

```
# cd /opt/SUNWwbsvr/bin
# ./wadm add-webapp --user=admin --host=AuthenticationUI-2.example.com
  --port=8989 --config=AuthenticationUI-2.example.com
  --vs=AuthenticationUI-2.example.com
  --uri=/distAuth /export/da71adm/amauthdistui.war

Please enter admin-user-password:web4dmin
...
Do you trust the above certificate? [y|n] y

CLI201 Command 'add-webapp' ran successfully
```

3 **Deploy the Distributed Authentication User Interface WAR.**

```
# ./wadm deploy-config --user=admin --host=AuthenticationUI-2.example.com
  --port=8989 AuthenticationUI-2.example.com
Please enter admin-user-password: web4dmin

CLI201 Command 'deploy-config' ran successfully
```

4 **Restart the Web Server `AuthenticationUI-2` instance.**

```
# cd /opt/SUNWwbsvr/https-AuthenticationUI-2.example.com/bin
# ./stopserv; ./startserv
```

5 **Verify that the `distAuth` web module is loaded.**

```
# cd /opt/SUNWwbsvr/https-AuthenticationUI-2.example.com/
  web-app/AuthenticationUI-2.example.com
# ls -al

total 6
drwxr-xr-x   3 da71adm  staff        512 Jul 19 21:00 .
drwxr-xr-x   3 da71adm  staff        512 Jul 19 21:00 ..
drwxr-xr-x   8 da71adm  staff        512 Jul 19 21:00 distAuth
```

6 **Log out of the AuthenticationUI–2 host machine.**

▼ To Import the Access Manager Load Balancer Certificate Authority Root Certificate into the Distributed Authentication User Interface 2

Import a Certificate Authority (CA) root certificate that enables the Distributed Authentication User Interface to trust the SSL certificate from the Access Manager Load Balancer 3, and establish trust with the certificate chain that is formed from the CA to the certificate.

1 **As a root user, log in to the AuthenticationUI–2 host machine.**

2 **Copy the CA root certificate into a directory.**

Use the same root certificate installed in "To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer" on page 108. In this example, the file is `/export/software/ca.cer`.

3 **Import the CA root certificate into the Java keystore.**

```
# /opt/SUNWwbsvr/jdk/jre/bin/keytool -import -trustcacerts
  -alias OpenSSLTestCA -file /export/software/ca.cer
  -keystore /opt/SUNWwbsvr/jdk/jre/lib/security/cacerts
  -storepass password

Owner: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
O=Sun,L=Santa Clara, ST=California C=US
Issuer: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
O=Sun,L=Santa Clara, ST=California C=US
Serial number: 97dba0aa26db6386
Valid from: Tue Apr 18 07:66:19 PDT 2006 until: Tue Jan 13 06:55:19
PST 2009
Certificate fingerprints:
                MD5: 9f:57:ED:B2:F2:88:B6:E8:0F:1E:08:72:CF:70:32:06
     SHA1: 31:26:46:15:C5:12:5D:29:46:2A:60:A1:E5:9E:26:64:36:80:E4:70

Trust this certificate: [no] yes

Certificate was added to keystore.
```

4 **Verify that the CA root certificate was imported into the keystore.**

```
# /opt/SUNWwbsvr/jdk/jre/bin/keytool -list
  -keystore /opt/SUNWwbsvr/jdk/jre/lib/security/cacerts
  -storepass password | grep -i open

openssltestca, Nov 8, 2006, trustedCertEntry
```

5 **Restart the Web Server `AuthenticationUI-2` instance.**

```
# cd /opt/SUNWwbsvr/https-AuthenticationUI-2.example.com/bin
# ./stopserv

server has been shutdown

# ./startserv

Sun Java System Web Server 7.0 B12/04/2006 07:59
info: CORE5076: Using [Java HotSpot(TM) Server VM,
Version 1.5.0_09] from [Sun Microsystems Inc.]
info: WEB0100: Loading web module in virtual server
[AuthenticationUI-2.example.com] at [/distAuth]
info: HTTP3072: http-listener-1: http://AuthenticationUI-2.
example.com:1080 ready to accept requests
info: CORE3274: successful server startup
```

6 **Log out of the AuthenticationUI–2 host machine.**

▼ To Verify that Authentication Through the Distributed Authentication User Interface 2 is Successful

Find a host that has direct network connectivity to Distributed Authentication User Interface 2 and the external facing load balancer of the Access Manager servers. One natural place is the AuthenticationUI–2 host machine itself.

1 **As a root user, log into the AuthenticationUI–2 host machine.**

2 **Modify `AMConfig.properties`.**

a. **Change to the `classes` directory.**

```
# cd /opt/SUNWwbsvr/https-AuthenticationUI-2.example.com/
  web-app/AuthenticationUI-2.example.com/distAuth/WEB-INF/classes
```

Tip – Backup `AMConfig.properties` before you modify it.

---

b. **Set the values of the properties as follows.**

```
com.iplanet.am.naming.url=https://LoadBalancer-3.
  example.com:9443/amserver/namingservice
com.iplanet.am.server.protocol=https
com.iplanet.am.server.port=9443
```

c. **Save the file and close it.**

3. **Restart the AuthenticationUI-2 host machine.**

   ```
   # cd /opt/SUNWwbsvr/https-AuthenticationUI-2.example.com/bin
   # ./stopserv; ./startserv
   ```

4. **Access** `http://AuthenticationUI-2.example.com:1080/distAuth/UI/Login?goto=http://LoadBalancer-3.example.com:7070` **from a web browser.**

5. **Log in to the Access Manager console as the administrator.**

   Username **`amadmin`**

   Password **`4m4dmin1`**

   After successful authentication, you should be redirected to the index page for the Web Server in which Access Manager is deployed.

6. **Log out of the Access Manager console.**

8.4 Configuring the Distributed Authentication User Interface Load Balancer

The following figure illustrates how Load Balancer 4 is configured in front of the two instances of the Distributed Authentication User Interface.

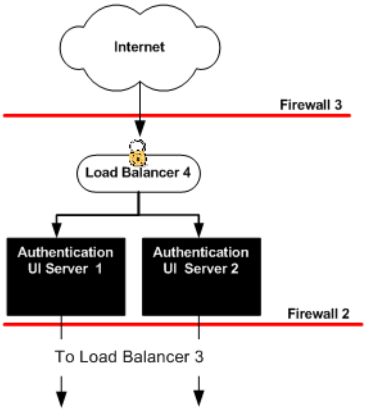


FIGURE 8–1 Distributed Authentication

Use the following list of procedures as a checklist for configuring the Distributed Authentication User Interface load balancer.

▼ To Configure the Distributed Authentication User Interface Load Balancer

Before You Begin

- This procedure assumes that you have already installed a load balancer.
- The load balancer hardware and software used in the lab facility for this deployment is BIG-IP® manufactured by F5 Networks. If you are using different load balancer software, see the documentation that comes with that product for detailed settings information.
- Contact your network administrator to obtain two available virtual IP addresses.
- Know the IP address of the load balancer hardware, the URL for the load balancer login page, and a username and password for logging in to the load balancer application.
- Get the IP addresses for Distributed Authentication User Interface 1 and Distributed Authentication User Interface 2 by running the following command on each host machine:

```
# ifconfig -a
```

1 Access `https://is-f5.example.com`**, the Big IP load balancer login page, from a web browser.**

2 Log in using the following information.

User name: **`username`**

Password: **`password`**

3 Click *Configure your BIG-IP (R) using the Configuration Utility*.

4 Create a Pool.

A pool contains all the backend server instances.

a. In the left pane, click Pools.

b. On the Pools tab, click Add.

c. In the Add Pool dialog, provide the following information:

Pool Name	**`AuthenticationUI-Pool`**
Load Balancing Method	Round Robin
Resources	Add the IP address and port number of both Distributed Authentication User Interface host machines: `AuthenticationUI-1:1080` and `AuthenticationUI-2:1080`.

d. Click Done.

5 Add a Virtual Server.

This step defines instances of the load balancer.

Tip – If you encounter JavaScript™ errors or otherwise cannot proceed to create a virtual server, try using Internet Explorer.

---

a. In the left frame, Click Virtual Servers.

b. On the Virtual Servers tab, click Add.

c. In the Add Virtual Server wizard, enter the virtual server IP address and port number.

Address	Enter the IP address for `LoadBalancer-4.example.com`
Service	**`90`**
Pool	**`AuthenticationUI-Pool`**

d. Continue to click Next until you reach the Pool Selection dialog box.

e. In the Pool Selection dialog box, assign the `AuthenticationUI-Pool` Pool.

f. Click Done.

6 Add Monitors.

Monitors are required for the load balancer to detect the backend server failures.

a. In the left frame, click Monitors.

b. Click the Basic Associations tab.

c. **Add an HTTP monitor to each Web Server node.**

   In the Node list, locate the IP address and port number for `AuthenticationUI-1:1080` and `AuthenticationUI-2:1080`, and select the Add checkbox.

d. **Click Apply.**

7 **Configure the load balancer for persistence.**

a. **In the left frame, click Pools.**

b. **Click the `AuthenticationUI-Pool` link.**

c. **Click the Persistence tab.**

d. **Under Persistence Type, choose Passive HTTP Cookie and click Apply.**

8 **To verify that the Distributed Authentication User Interface load balancer is configured properly, access `http://LoadBalancer-4.example.com:90/` from a web browser.**

If the browser successfully renders the default Web Server document root page, the load balancer has been configured properly.

▼ To Configure Load Balancer Cookies for the Distributed Authentication User Interface

Modify `AMconfig.properties` on both Distributed Authentication User Interface host machines.

1 **Log in as a root user to the AuthenticationUI–1 host machine.**

2 **Change to the `classes` directory.**

```
# cd /opt/SUNWwbsvr/https-AuthenticationUI-1.example.com/
  web-app/AuthenticationUI-1.example.com/distAuth/WEB-INF/classes
```

3 **Make the following changes to `AMconfig.properties`.**

Tip – Backup `AMConfig.properties` before you modify it.

- Uncomment the last two lines at the end of the file.
- Set the following values:

```
com.iplanet.am.lbcookie.name=AuthenticationUILBCookie
com.iplanet.am.lbcookie.value=AuthenticationUI-1
```

4 **Save the file and close it.**

5 **Restart the AuthenticationUI–1 host machine.**

6 **Log in as a root user to the AuthenticationUI–2 host machine.**

7 **Change to the `classes` directory.**

```
# cd /opt/SUNWwbsvr/https-AuthenticationUI-2.example.com/
  web-app/AuthenticationUI-2.example.com/distAuth/WEB-INF/classes
```

8 **Make the following changes to `AMconfig.properties`.**

Tip – Backup `AMConfig.properties` before you modify it.

- Uncomment the last two lines at the end of the file.
- Set the following values:

```
com.iplanet.am.lbcookie.name=AuthenticationUILBCookie
com.iplanet.am.lbcookie.value=AuthenticationUI-2
```

9 **Save the file and close it.**

10 **Restart the AuthenticationUI–2 host machine.**

▼ To Request a Secure Sockets Layer Certificate for the Distributed Authentication User Interface Load Balancer

Generate a request for a Secure Sockets Layer (SSL) certificate to send to a certificate authority.

1 **Access `https://is-f5.example.com`, the BIG-IP load balancer login page, from a web browser.**

2 **Log in to the BIG-IP console using the following information.**

User Name: **`username`**

Password: **`password`**

3 **Click** ***Configure your BIG-IP (R) using the Configuration Utility*****.**

4 **In the left pane, click Proxies.**

5 **Click the Cert-Admin tab.**

6 **On the SSL Certificate Administration page, click** *Generate New Key Pair/Certificate Request*.

7 **On the Create Certificate Request page, provide the following information:**

Key Identifier:	**`LoadBalancer-4.example.com`**
Organizational Unit Name:	**`Deployment`**
Domain Name:	**`LoadBalancer-4.example.com`**
Challenge Password:	**`password`**
Retype Password:	**`password`**

8 **Click** *Generate Key Pair/Certificate Request*.

On the SSL Certificate Request page, the request is generated in the Certificate Request field.

9 **Save the text contained in the Certificate Request field to a text file.**

10 **Log out of the console and close the browser.**

11 **Send the certificate request text you saved to the Certificate Authority of your choice.**

A Certificate Authority (CA) is an entity that issues certified digital certificates; VeriSign, Thawte, Entrust, and GoDaddy are just a few. In this deployment, CA certificates were obtained from OpenSSL. Follow the instructions provided by your Certificate Authority to submit a certificate request.

▼ To Import a CA Root Certificate on the Distributed Authentication User Interface Load Balancer

The CA root certificate proves that the particular CA (such as VeriSign or Entrust) did, in fact, issue a particular SSL certificate. You install the root certificate on Load Balancer 4 to ensure that a link between the Load Balancer 4 SSL certificate can be maintained with the issuing company. CA root certificates are publicly available.

Before You Begin You should have a CA root certificate.

1 **Access** `https://is-f5.example.com`**, the Big IP load balancer login page, from a web browser.**

2 **Log in using the following information:**

User name:	`username`
Password:	`password`

3 **In the BIG-IP load balancer console, click Proxies.**

4 **Click the Cert-Admin tab.**

5 **Click Import.**

6 **In the Import Type field, choose Certificate, and click Continue.**

7 **Click Browse in the Certificate File field on the Install SSL Certificate page.**

8 **In the Choose File dialog, choose Browser.**

9 **Navigate to the file that includes the root CA Certificate and click Open.**

10 **In the Certificate Identifier field, enter `OpenSSL_CA_cert`.**

11 **Click Install Certificate.**

12 **On the Certificate OpenSSL_CA_Cert page, click Return to Certificate Administration.**

The root certificate OpenSSL_CA_Cert is now included in the Certificate ID list.

▼ To Install an SSL Certificate on the Distributed Authentication User Interface Load Balancer

Before You Begin This procedure assumes you have received an SSL certificate from a CA and just completed "To Import a CA Root Certificate on the Distributed Authentication User Interface Load Balancer" on page 159.

1 **In the BIG-IP load balancer console, click Proxies.**

2 **Click the Cert-Admin tab.**

The key `LoadBalancer-4.example.com` is in the Key List. This was generated in "To Request a Secure Sockets Layer Certificate for the Distributed Authentication User Interface Load Balancer" on page 158.

3 **In the Certificate ID column, click the Install button for `LoadBalancer-4.example.com`.**

4 **In the Certificate File field, click Browse.**

5 **In the Choose File dialog, navigate to the file that contains the certificate text sent to you by the CA and click Open.**

6 **Click Install Certificate.**

7 **On the Certificate LoadBalancer-4.example.com page, click Return to Certificate Administration Information.**

Verify that the Certificate ID indicates `LoadBalancer-4.example.com` on the SSL Certificate Administration page.

8 **Log out of the load balancer console.**

▼ To Configure SSL Termination on the Distributed Authentication User Interface Load Balancer

Secure Socket Layer (SSL) termination at Load Balancer 4 increases performance on the Access Manager level, and simplifies SSL certificate management. For example, because Load Balancer 4 sends unencrypted data internally neither the Access Manager server nor the Distributed Authentication User Interface has to perform decryption, and the burden on its processor is relieved. Clients send SSL-encrypted data to Load Balancer 4 which, in turn, decrypts the data and sends the unencrypted data to the appropriate Distributed Authentication User Interface. Load Balancer 4 also encrypts responses from the Distributed Authentication User Interface, and sends these encrypted responses back to the client. Towards this end, you create an *SSL proxy*, the gateway for decrypting HTTP requests and encrypting the reply.

Note – Load Balancer 4 can intelligently load-balance a request based on unencrypted cookies. This would not be possible with SSL-encrypted cookies because Load Balancer 4 cannot read SSL-encrypted cookies.

Before You Begin Before creating the SSL proxy, you should have a certificate issued by a recognized CA.

1 **Access** `https://is-f5.example.com`**, the BIG-IP load balancer login page, in a web browser.**

2 **Log in using the following information:**

Username	**`username`**
Password	**`password`**

3 **Click** ***Configure your BIG-IP using the Configuration Utility*****.**

4 **In the left pane, click Proxies.**

5 **On the Proxies tab, click Add.**

6 **In the Add Proxy dialog, provide the following information:**

Proxy Type: Check the SSL checkbox.

Proxy Address: The IP address of Load Balancer 4, the Distributed Authentication User Interface load balancer.

Proxy Service: `9443`

The secure port number

Destination Address: The IP address of Load Balancer 4, the Distributed Authentication User Interface load balancer.

Destination Service: `90`

The non-secure port number

Destination Target: Choose `Local Virtual Server`.

SSL Certificate: Choose `LoadBalancer-4.example.com`.

SSL Key: Choose `LoadBalancer-4.example.com`.

Enable ARP: Check this checkbox.

7 **Click Next.**

8 **In the Rewrite Redirects field, choose `All`.**

9 **Click Done.**

The new proxy server is now added to the Proxy Server list.

10 **Log out of the load balancer console.**

11 **Access** `https://LoadBalancer-4.example.com:9443/distAuth/UI/Login?goto=https://LoadBalancer-3.example.com:9443` **from a web browser.**

Tip – A message may be displayed indicating that the browser doesn't recognize the certificate issuer. If this happens, install the CA root certificate in the browser so that the browser recognizes the certificate issuer. See your browser's online help system for information on installing a root CA certificate.

12 **Log in to the Access Manager console as the administrator.**

Username	`amadmin`
Password	`4m4dmin1`

If you can successfully log in to Access Manager, the SSL certificate is installed and the proxy service is configured properly.

13 **Log out of Access Manager, and close the browser.**

CHAPTER 9

Configuring the Protected Resource Host Machines

Each Protected Resource host machine will have two installed web containers (one Sun Java™ System Web Server and one BEA WebLogic Server application server) and an appropriate policy agent for each (a web policy agent and a J2EE policy agent, respectively). The policy agents are configured to access Load Balancer 3. This chapter contains the following tasks:

- "9.1 Configuring Protected Resource 1" on page 165
- "9.2 Configuring Protected Resource 2" on page 207

9.1 Configuring Protected Resource 1

We will install Sun Java™ System Web Server, BEA WebLogic Server, a web policy agent, and a J2EE policy agent on the ProtectedResource–1 host machine. The policy agents are configured to access Load Balancer 3 as illustrated in the following figure.

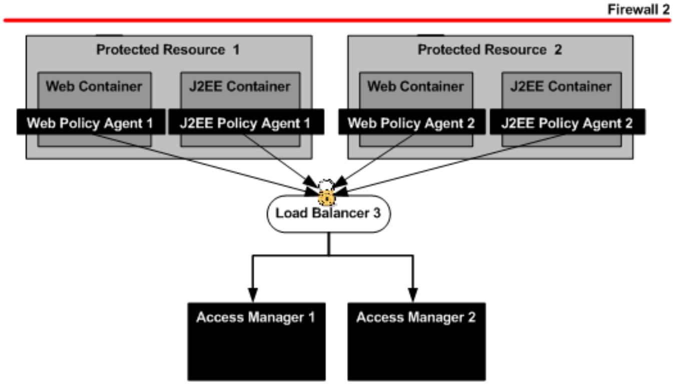


FIGURE 9–1 Protected Resources and Policy Agents

Use the following list of procedures as a checklist for configuring the ProtectedResource–1 host machine.

1. "9.1.1 Installing Web Container 1 and Web Policy Agent 1 on Protected Resource 1" on page 166
2. "9.1.2 Installing and Configuring the J2EE Container 1 and J2EE Policy Agent 1 on Protected Resource 1" on page 181
3. "9.1.3 Setting Up a Test for the J2EE Policy Agent 1" on page 195
4. "9.1.4 Configuring the J2EE Policy Agent 1 to Communicate Over SSL" on page 202

9.1.1 Installing Web Container 1 and Web Policy Agent 1 on Protected Resource 1

Install Sun Java System Web Server and a web policy agent on the ProtectedResource–1 host machine as well as supporting configurations. Use the following list of procedures as a checklist.

1. "To Create an Agent Profile for Web Policy Agent 1" on page 166
2. "To Install Sun Java System Web Server as Web Container 1 on Protected Resource 1" on page 167
3. "To Install and Configure Web Policy Agent 1 on Protected Resource 1" on page 172
4. "To Import the Certificate Authority Root Certificate into the Web Server 1 Keystore" on page 176
5. "To Configure Policy for Web Policy Agent 1 on Protected Resource 1" on page 177
6. "To Verify that Web Policy Agent 1 is Working Properly" on page 180

▼ To Create an Agent Profile for Web Policy Agent 1

Create an agent profile in Access Manager to store authentication and configuration information that will be used by the policy agent to authenticate itself to Access Manager. Creating an agent profile also creates a custom user. The policy agent will, by default, use the account with the user identifier `UrlAccessAgent` to authenticate to Access Manager.

Note – Creating an agent profile is not a requirement for web policy agents. You can use the agent's default values and not change the user name; however, in certain cases, you might want to change these default values. For example, if you want to audit the interactions between multiple agents and Access Manager, you want be able to distinguish one agent from another. This would not be possible if all the agents used the same default agent user account. For more information, see the *Sun Java System Access Manager Policy Agent 2.2 User's Guide*.

---

1 Access `http://AccessManager-1.example.com:1080/amserver/UI/Login` from a web browser.

2 Log in to the Access Manager console as the administrator.

User Name:	**amadmin**
Password:	**4m4dmin1**

3 **Under the Access Control tab, click `example`, the top-level Realm Name.**

4 **Click the Subjects tab.**

5 **Click the Agents tab.**

6 **Click New to create a new agent profile.**

7 **On the resulting page, enter the following and click OK.**

ID	**webagent-1**
Password:	**web4gent1**
Password Confirm	**web4gent1**
Device State	Choose Active.

The new agent `webagent-1` is displayed in the list of agent users.

8 **Log out of the console.**

▼ To Install Sun Java System Web Server as Web Container 1 on Protected Resource 1

Download the Sun Java System Web Server bits and install the software on the ProtectedResource–1 host machine.

1 **As a root user, log into the ProtectedResource–1 host machine.**

2 **Install required patches if necessary.**

Note – Results for your machines might be different. Read the latest version of the Web Server 7.0 Release Notes to determine if you need to install patches and, if so, what they might be. In this case, the Release Notes indicate that based on the hardware and operating system being used, patch 117461–08 is required.

---

a. **Run `patchadd` to see if the patch is installed.**

```
# patchadd -p | grep 117461–08
```

No results are returned which indicates that the patch is not yet installed on the system.

b. **Make a directory for downloading the patch you need and change into it.**

```
# mkdir /export/patches
# cd /export/patches
```

c. **Download the patches.**

You can search for patches directly at http://sunsolve.sun.com. Navigate to the PatchFinder page, enter the patch number, click Find Patch, and download the appropriate patch.

Note – Signed patches are downloaded as JAR files. Unsigned patches are downloaded as ZIP files.

---

d. **Unzip the patch file.**

```
# unzip 117461–08.zip
```

e. **Run `patchadd` to install the patches.**

```
# patchadd /export/patches/117461–08
```

f. **After installation is complete, run `patchadd` to verify that the patch was added successfully.**

```
# patchadd -p | grep 117461–08
```

In this example, a series of patch numbers are displayed, and the patch 117461–08 is present.

3 **Create a directory into which you can download the Web Server bits and change into it.**

```
# mkdir /export/ws7
# cd /export/ws7
```

4 **Download the Sun Java System Web Server 7.0 software from http://www.sun.com/download/products.xml?id=45ad781d.**

Follow the instructions on the Sun Microsystems Product Downloads web site for downloading the software. In this example, the software was downloaded to the `/export/WS7` directory.

```
# ls -al

total 294548
drwxr-xr-x   2 root     root         512 Aug  7 13:23 .
drwxr-xr-x   3 root     sys          512 Aug  7 13:16 ..
-rw-r--r--   1 root     root     150719523 Aug  7 13:24 sjsws-7_0-solaris-sparc.tar.gz
```

5 **Unpack the Web Server bits.**

```
# gunzip sjsws-7_0-solaris-sparc.tar.gz
# tar xvf sjsws-7_0-solaris-sparc.tar
```

6 **Run** `setup`**.**

```
# ./setup --console
```

7 **When prompted, provide the following information.**

You are running the installation program for the Sun Java System Web Server 7.0. ... The installation program pauses as questions are presented so you can read the information and make your choice. When you are ready to continue, press Enter.	Press Enter. Continue to press Enter when prompted.
Have you read the Software License Agreement and do you accept all the terms?	Enter **yes**.
Sun Java System Web Server 7.0 Installation Directory [/sun/webserver7]	Enter **/opt/SUNWwbsvr**
Specified directory /opt/SUNWwbsvr does not exist. Create Directory? [Yes/No]	Enter **yes**.
Select Type of Installation 1. Express 2. Custom 3. Exit What would you like to do? [1]	Enter **2**.
Component Selection 1. Server Core 2. Server Core 64-biy Binaries 3. Administration Command Line Interface 4. Sample Applications 5. Language Pack Enter the comma-separated list [1,2,3,4,5]	Enter **1,3,5**.
Java Configuration 1. Install Java Standard Edition 1.5.0_09 2. Reuse existing Java SE 1.5.0_09 or greater 3. Exit What would you like to do? [1]	Enter **1**.
Administrative Options 1. Create an Administration Server and a Web Server Instance 2. Create an Administration Node Enter your option. [1]	Enter **1**.

`Start servers during system startup. [yes/no]`	Enter **`no`**.
`Host Name [ProtectedResource-1.example.com]`	Accept the default value.
`SSL Port [8989]`	Accept the default value.
`Create a non-SSL Port? [yes/no]`	Enter **`no`**.
`Runtime User ID [webservd]`	Enter **`root`**.
`Administrator User Name [admin]`	Accept the default value.
`Administrator Password:`	Enter **`web4dmin`**.
`Retype Password:`	Enter **`web4dmin`**.
`Server Name [ProtectedResource-1.example.com]`	Accept the default value.
`Http Port [8080]`	Enter **`1080`**.
`Document Root Directory [/opt/SUNWwbsvr/ https-ProtectedResource-1.example.com/docs]`	Accept the default value.

`Ready To Install`
`1. Install Now`
`2. Start Over`
`3. Exit Installation`
`What would you like to do?`
	Enter**`1`**.

When installation is complete, the following message is displayed:

```
Installation Successful.
```

8 Start the Web Server administration server.

```
# cd /opt/SUNWwbsvr/admin-server/bin
# ./startserv

server not running
Sun Java System Web Server 7.0 B12/04/2006 10:15
info: CORE3016: daemon is running as super-user
info: CORE5076: Using [Java HotSpot(TM) Server VM, Version 1.5.0_09]
  from [Sun Microsystems Inc.]
info: WEB0100: Loading web module in virtual server [admin-server] at
  [/admingui ]
info: WEB0100: Loading web module in virtual server [admin-server] at
  [/jmxconne ctor]
 info: HTTP3072: admin-ssl-port: https://protectedresource-1.example.com:8989
  ready to accept requests
info: CORE3274: successful server startup
```

9 **Run `netstat` to verify that the port is open and listening.**

```
# netstat -an | grep 8989

*.8989               *.*                0      0 49152      0 LISTEN
```

10 **(Optional) Login to the Web Server administration console at `https://protectedresource-1.example.com:8989`.**

Username **admin**

Password **web4dmin**

You should see the Web Server console.

11 **Log out of the Web Server console.**

12 **Start the Protected Resource 1 Web Server instance.**

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-1.example.com/bin
# ./startserv

server not running
Sun Java System Web Server 7.0 B12/04/2006 10:15
info: CORE3016: daemon is running as super-user
info: CORE5076: Using [Java HotSpot(TM) Server VM,
   Version 1.5.0_09] from [Sun Microsystems Inc.]
info: HTTP3072: http-listener-1: http://ProtectedResource-1.example.com:1080
   ready to accept requests
info: CORE3274: successful server startup
```

13 **Run `netstat` to verify that the port is open and listening.**

```
# netstat -an | grep 1080

*.1080               *.*                0      0 49152      0 LISTEN
```

14 **(Optional) Access the Protected Resource 1 instance at `https://ProtectedResource-1.example.com:1080` using a web browser.**

You should see the default Web Server index page.

15 **Log out of the ProtectedResource–1 host machine.**

To Install and Configure Web Policy Agent 1 on Protected Resource 1

Caution – Due to a known problem with this version of the Web Policy Agent, you must start an X-display session on the server host using a program such as Reflections X or VNC, even though you use the command-line installer. For more information about this known problem, see "On UNIX-based machines, all web agents require that the X11 DISPLAY variable be set properly." in *Sun Java System Access Manager Policy Agent 2.2 Release Notes*.

1. **As a root user, log into the ProtectedResource–1 host machine.**

2. **Create a directory into which you can download the Web Server agent bits and change into it.**

```
# mkdir /export/WebPA1
# cd /export/WebPA1
```

3. **Download the web policy agent for Web Server from `http://www.sun.com/download/`.**

```
# ls -al

total 294548
drwxr-xr-x   2 root     root         512 Aug  7 13:23 .
drwxr-xr-x   3 root     sys          512 Aug  7 13:16 ..
-rw-r--r--   1 root     root     150719523 Aug  7 13:24 sjsws_v70_SunOS_agent.zip
```

4. **Unzip the downloaded file.**

```
# unzip sjsws_v70_SunOS_agent.zip
```

5. **Change the permissions for the resulting `agentadmin` binary.**

```
# cd /export/WebPA1/web_agents/sjsws_agent/bin
# chmod +x agentadmin
```

6. **Verify that `crypt_util` has execute permission before running the installer.**

```
# cd /export/WebPA1/web_agents/sjsws_agent/bin
# chmod +x crypt_util
```

7. **Create a temporary file for the password that will be required later during agent installation.**

```
# echo web4gent1 > /export/WebPA1/pwd.txt
# cat /export/WebPA1/pwd.txt
```

8. **Run the agent installer.**

```
# ./agentadmin --install
```

9. **When prompted, do the following.**

Do you completely agree with all the terms and conditions of this License Agreement (yes/no): [no]:	Type **yes** and press Enter.
*** Welcome to the Access Manager Policy Agent for Sun Java System Web Server If the Policy Agent is used with Federation Manager services, User needs to enter information relevant to Federation Manager. ***	
Enter the complete path to the directory which is used by Sun Java System Web Server to store its configuration Files. This directory uniquely identifies the Sun Java System Web Server instance that is secured by this Agent. [? : Help, ! : Exit] Enter the Sun Java System Web Server Config Directory Path [/var/opt/SUNWwbsvr7/ https-ProtectedResource-1.example.com/config]:	Type **/opt/SUNWwbsvr/https-ProtectedResource-1.example.com/config** and press Enter.
Enter the fully qualified host name of the server where Access Manager Services are installed. [? : Help, < : Back, ! : Exit] Access Manager Services Host:	Type **LoadBalancer-3.example.com** and press Enter.
Enter the port number of the Server that runs Access Manager Services. [? : Help, < : Back, ! : Exit] Access Manager Services port [80]:	Type **9443** and press Enter.
Enter http/https to specify the protocol used by the Server that runs Access Manager services. [? : Help, < : Back, ! : Exit] Access Manager Services Protocol [http]:	Type **https** and press Enter.
Enter the Deployment URI for Access Manager Services. [? : Help, < : Back, ! : Exit] Access Manager Services Deployment URI [/amserver]:	Press Enter to accept the default /amserver.
Enter the fully qualified host name on which the Web Server protected by the agent is installed. [? : Help, < : Back, ! : Exit] Enter the Agent Host name:	Type **ProtectedResource-1.example.com** and press Enter.
Enter the preferred port number on which the Web Server provides its services. [? : Help, < : Back, ! : Exit] Enter the port number for Web Server instance [80]:	Type **1080** and press Enter.

`Select http or https to specify the protocol`
`used by the Web server instance that will be protected`
`by Access Manager Policy Agent.`
`[ ? : Help, < : Back, ! : Exit ]`
`Enter the Preferred Protocol for Web Server`
`instance [http]:`
	Press Enter to accept the default `http`.

`Enter a valid Agent profile name. Before`
`proceeding with the agent installation, please ensure`
`that a valid Agent profile exists in Access Manager.`
`[ ? : Help, < : Back, ! : Exit ]`
`Enter the Agent Profile name [UrlAccessAgent]:`
	Type **`webagent-1`** and press Enter.

`Enter the path to a file that contains the`
`password to be used for identifying the Agent.`
`[ ? : Help, < : Back, ! : Exit ]`
`Enter the path to the password file:`
	Type **`/export/WebPA1/pwd.txt`** and press Enter.

`-----------------------------------------`
`SUMMARY OF YOUR RESPONSES`
`-----------------------------------------------`
`Sun Java System Web Server Config Directory :`
`/opt/SUNWwbsvr/https-ProtectedResource-1.`
`  example.com/config`
`Access Manager Services Host : LoadBalancer-3.`
`  example.com`
`Access Manager Services Port : 9443`
`Access Manager Services Protocol : https`
`Access Manager Services Deployment URI : /amserver`
`Agent Host name : ProtectedResource-1.example.com`
`Web Server Instance Port number : 1080`
`Protocol for Web Server instance : http`
`Agent Profile name : webagent-1`
`Agent Profile Password file name : /export/WebPA1/`
`  pwd.txt`
 	
`Verify your settings above and decide from the`
`   choices below.`
`1. Continue with Installation`
`2. Back to the last interaction`
`3. Start Over`
`4. Exit`
`Please make your selection [1]:`
	Type **`1`** and press Enter.

```
Creating directory layout and configuring Agent
file for Agent_001 instance ...DONE.

Reading data from file /export/WebPA1/pwd.txt and
encrypting it ...DONE.

Generating audit log file name ...DONE.

Creating tag swapped AMAgent.properties file for
instance Agent_001 ...DONE.

Creating a backup for file
/opt/SUNWwbsvr/https-ProtectedResource-1.example.com/
   config/obj.conf ...DONE.

Creating a backup for file
/opt/SUNWwbsvr/https-ProtectedResource-1.example.com/
   config/magnus.conf ...DONE.

Adding Agent parameters to
/opt/SUNWwbsvr/https-ProtectedResource-1.example.com/
   config/magnus.conf file ...DONE.

Adding Agent parameters to
/opt/SUNWwbsvr/https-ProtectedResource-1.example.com/
   config/obj.conf file ...DONE.


SUMMARY OF AGENT INSTALLATION
-----------------------------
Agent instance name: Agent_001
Agent Configuration file location:
/export/WebPA1/web_agents/sjsws_agent/Agent_001/
  config/AMAgent.properties
Agent Audit directory location:
/export/WebPA1/web_agents/sjsws_agent/Agent_001/
  logs/audit
Agent Debug directory location:
/export/WebPA1/web_agents/sjsws_agent/Agent_001/
  logs/debug

Install log file location:
/export/WebPA1/web_agents/sjsws_agent/logs/audit/
  install.log

Thank you for using Access Manager Policy Agent
```

10 **Modify the `AMAgent.properties` file.**

Tip – Backup `AMAgent.properties` before you modify it.

a. **Change to the `config` directory.**

```
# cd /export/WebPA1/web_agents/sjsws_agent/Agent_001/config
```

b. **Set the values of the following properties as shown.**

```
com.sun.am.policy.am.login.url = https://LoadBalancer-3.
   example.com:9443/amserver/UI/Login?realm=users
com.sun.am.load_balancer.enable = true
```

c. **Save the file and close it.**

11 **Restart the Protected Resource 1 Web Server instance.**

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-1.example.com/bin
# ./stopserv; ./startserv

server has been shutdown
Sun Java System Web Server 7.0 B12/04/2006 10:15
info: CORE3016: daemon is running as super-user info:
CORE5076: Using [Java HotSpot(TM) Server VM, Version 1.5.0_09]
  from [Sun Microsystems Inc.]
info: HTTP3072: http-listener-1: http://ProtectedResource-1.example.com:1080
  ready to accept requests
```

12 **Log out of the ProtectedResource–1 host machine.**

▼ To Import the Certificate Authority Root Certificate into the Web Server 1 Keystore

The web policy agent on Protected Resource 1 connects to Access Manager through Load Balancer 3. The load balancer is SSL-enabled, so the agent must be able to trust the load balancer SSL certificate to establish the SSL connection. For this reason, import the root certificate of the Certificate Authority (CA) that issued the Load Balancer 3 SSL server certificate into the policy agent keystore.

Before You Begin Import the same CA root certificate used in "To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer" on page 108.

1 **As a root user, log into the ProtectedResource–1 host machine.**

2 **Copy the CA root certificate into a directory.**

In this example, the file is `/export/software/ca.cer`.

3 Import the CA root certificate into the Java keystore.

```
# /opt/SUNWwbsvr/jdk/jre/bin/keytool -import -trustcacerts
  -alias OpenSSLTestCA -file /export/software/ca.cer
  -keystore /opt/SUNWwbsvr/jdk/jre/lib/security/cacerts -storepass changeit

Owner: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
O=Sun,L=Santa Clara, ST=California C=US
Issuer: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
O=Sun,L=Santa Clara, ST=California C=US
Serial number: 97dba0aa26db6386
Valid from: Tue Apr 18 07:66:19 PDT 2006 until: Tue Jan 13 06:55:19
PST 2009
Certificate fingerprints:
MD5: 9f:57:ED:B2:F2:88:B6:E8:0F:1E:08:72:CF:70:32:06
SHA1: 31:26:46:15:C5:12:5D:29:46:2A:60:A1:E5:9E:26:64:36:80:E4:70
Trust this certificate: [no] yes
Certificate was added to keystore.
```

4 Verify that the CA root certificate was imported.

```
# /opt/SUNWwbsvr/jdk/jre/bin/keytool -list
  -keystore /opt/SUNWwbsvr/jdk/jre/lib/security/cacerts
  -storepass changeit | grep -i open

opensslestca, Sep 10, 2007, trustedCertEntry,
```

5 Restart the Web Server 1 instance.

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-1.example.com/bin
# ./stopserv; ./startserv

server has been shutdown
Sun Java System Web Server 7.0 B12/04/2006 10:15
info: CORE3016: daemon is running as super-user
info: CORE5076: Using [Java HotSpot(TM) Server VM,
Version 1.5.0_09] from [Sun Microsystems Inc.]
info: HTTP3072: http-listener-1: http://ProtectedResource-1.
example.com:1080 ready to accept requests
info: CORE3274: successful server startup
```

6 Log out of the ProtectedResource–1 host machine.

To Configure Policy for Web Policy Agent 1 on Protected Resource 1

Use the Access Manager console to configure policy for Web Policy Agent 1. This policy will be used to verify that Web Policy Agent 1 is working properly.

Note – You will modify this policy later when we add a load balancer in front of it.

1 **Access** `http://AccessManager-1.example.com:1080/amserver/UI/Login` **from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

Username	**`amadmin`**
Password	**`4m4dmin1`**

3 **Create a referral policy in the top-level realm.**

a. **Under the Access Control tab, click the top-level realm,** `example`**.**

b. **Click the Policies tab.**

c. **Click New Referral.**

d. **On the New Policy page, provide the following information.**

Name:	Referral URL Policy for users realm
Active:	Mark the Yes checkbox.

e. **On the same page, in the Rules section, click New.**

f. **On the resulting page, select URL Policy Agent (with resource name) as a Service Type and click Next.**

g. **Provide the following information on the resulting page:**

Name:	**`URL Rule for ProtectedResource-1`**
Resource Name:	**`http://ProtectedResource-1.example.com:1080/*`**

h. **Click Finish.**

i. **Back on the New Policy page, under the Referrals section, click New.**

j. **Provide the following information on the New Referral — Sub Realm page.**

Name:	**`Sub-Realm users`**
Filter:	Type an asterisk (*), and click Search.
Value:	In the list, choose `users`.

k. **Click Finish.**

l. **Back on the New Policy page, click OK.**

   Under the Policies tab for the `example` realm, you should see the policy named *Referral URL Policy for users realm*.

4 **Create a policy in the `users` realm.**

The `users` realm was previously created in "7.2 Creating and Configuring a Realm for Test Users" on page 121.

a. **Click the Access Control tab.**

b. **Under Realms, click `users`.**

c. **Click the Policies tab.**

d. **Click New Policy.**

e. **On the New Policy page, provide the following information:**

Name:	**`URL Policy for ProtectedResource-1`**
Active:	Mark the Yes checkbox.

f. **On the same page, in the Rules section, click New.**

g. **Select a Service Type for the rule and click Next.**

   URL Policy Agent (with resource name) is the only choice.

h. **On the resulting page, provide the following information:**

Name:	**`URL Rule for ProtectedResource-1`**
Resource Name:	Click `http://ProtectedResource-1.example.com:1080/*`, listed in the Parent Resource Name list, to add it to the Resource Name field.
GET:	Mark this checkbox, and select Allow.
POST:	Mark this checkbox, and select Allow.

i. **Click Finish.**

5 **Create a new subject in the `users` realm for testing.**

a. **On the New Policy page, in the Subjects section, click New.**

b. **Select Access Manager Identity Subject as the subject type and click Next.**

c. **Provide the following information on the resulting page.**

Name:	**`Test Subject`**
Filter:	Choose `User` and click Search. Two users are added to the Available list.
Available:	In the list, select `Test User1` and click Add.

d. **Click Finish.**

6 **Back on the New Policy page, click Create.**

Under the Policies tab, you should see the policy named *URL Policy for ProtectedResource-1*.

7 **Log out of the console.**

▼ To Verify that Web Policy Agent 1 is Working Properly

1 **Access** `http://ProtectedResource-1.example.com:1080` **from a web browser.**

2 **Log in to Access Manager as** `testuser1`**.**

Username	**`testuser1`**
Password	**`password`**

You should see the default index page for Web Server 1 as `testuser1` was configured in the test policy to be allowed to access Protected Resource 1.

3 **Log out and close the browser.**

4 **Once again, access** `http://ProtectedResource-1.example.com:1080` **from a web browser.**

Tip – If you are not redirected to the Access Manager login page for authentication, clear your browser's cache and cookies and try again.

5 **Log in to Access Manager as** `testuser2`**.**

Username	**`testuser2`**
Password	**`password`**

You should see the message, *You're not authorized to view this page*, (or *Your client is not allowed to access the requested object*) as `testuser2` was not included in the test policy that allows access to Protected Resource 1.

9.1.2 Installing and Configuring the J2EE Container 1 and J2EE Policy Agent 1 on Protected Resource 1

You will download the BEA WebLogic Server bits and install this application server on the ProtectedResource–1 host machine. Additionally, you will download and install the appropriate J2EE policy agent, deploy the policy agent application, setup up an authentication provider, and modify the Bypass Principal List. All of these tasks must be completed before the agent can do its job. Use the following list of procedures as a checklist for installing Application Server 1 and the J2EE Policy Agent 1.

1. "To Create an Agent Profile for the J2EE Policy Agent 1" on page 181
2. "To Create Manager and Employee Groups Using Access Manager for J2EE Policy Agent Test" on page 182
3. "To Install BEA WebLogic Server as J2EE Container 1 on Protected Resource 1" on page 184
4. "To Configure BEA WebLogic Server as J2EE Container 1 on Protected Resource 1" on page 185
5. "To Install the J2EE Policy Agent 1 on Application Server 1" on page 189
6. "To Deploy the J2EE Policy Agent 1 Application" on page 191
7. "To Start the J2EE Policy Agent 1 Application" on page 192
8. "To Set Up the J2EE Policy Agent 1 Authentication Provider" on page 193
9. "To Edit the J2EE Policy Agent 1 `AMAgent.properties` File" on page 194

▼ To Create an Agent Profile for the J2EE Policy Agent 1

This new agent profile will be used by J2EE Policy Agent 1 to authenticate to Access Manager.

1 **Access** `http://LoadBalancer-3.example.com:7070/amserver/UI/Login`**, the Access Manage load balancer, from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

Username	**`amadmin`**
Password	**`4m4dmin1`**

3 **On the Access Control tab, click the top-level realm,** `example`**.**

4 **Click the Subjects tab.**

5 **Click the Agents tab.**

6 **On the Agent page, click New.**

7 **On the New Agent page, provide the following information and click OK.**

ID:	**j2eeagent-1**
Password:	**j2ee4gent1**
Password Confirm:	**j2ee4gent1**
Device State:	Choose Active.

The new agent j2eeagent–1 is displayed in the list of Agent Users.

8 **Log out of the Access Manager console.**

9 **As a root user, log into the ProtectedResource–1 host machine.**

10 **Create a directory into which you can download the J2EE policy agent bits and change into it.**

```
# mkdir /export/J2EEPA1
# cd /export/J2EEPA1
```

11 **Create a text file that contains the Agent Profile password.**

The J2EE Policy Agent installer requires this file for installation.

```
# cat > agent.pwd
j2ee4gent1
```

Hit Control D to terminate the command

```
^D
```

12 **Log out of the ProtectedResource–1 host machine.**

▼ To Create Manager and Employee Groups Using Access Manager for J2EE Policy Agent Test

A *group* represents a collection of users with a common function, feature, or interest. The groups created in this section will be used to test the policy agent after installation.

1 **Access** http://LoadBalancer-3.example.com:7070/amserver/UI/Login**, the Access Manage load balancer, from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

Username	**amadmin**
Password	**4m4dmin1**

3 **On the Access Control tab, click the** users **realm.**

4 **Click the Subjects tab.**

5 Click the Groups tab.

6 Create a manager group for the Users realm.

a. On the Groups page, click New.

b. On the New Group page, enter `Manager-Group` as the ID and click OK.

The `Manager-Group` is displayed in the list of Groups.

c. Click `Manager-Group` in the list of Groups.

d. Copy the value of the Universal ID and save it to a text file.

You will need this value in "To Configure Properties for the J2EE Policy Agent 1 Sample Application" on page 198.

e. Click the Users tab.

You should see the users that were created in Chapter 7, "Configuring an Access Manager Realm for User Authentication."

f. Select Test User1 from the list and click Add.

g. Click Save.

h. Click Back to Subjects.

7 Create an employee group for the Users realm.

a. On the Groups page, click New.

b. On the New Group page, enter `Employee-Group` as the ID and click OK.

The `Employee-Group` is displayed in the list of Groups.

c. Click `Employee-Group` in the list of Groups.

d. Copy the value of the Universal ID and save it to a text file.

You will need this value in "To Configure Properties for the J2EE Policy Agent 1 Sample Application" on page 198.

e. Click the Users tab.

You should see the users that were created in Chapter 7, "Configuring an Access Manager Realm for User Authentication."

f. **Select Test User2 from the list and click Add.**

g. **Click Save.**

h. **Click Back to Subjects.**

8 **Log out of the Access Manager console.**

▼ To Install BEA WebLogic Server as J2EE Container 1 on Protected Resource 1

BEA WebLogic Server is the application server used as the J2EE container on Protected Resource 1. After installing the bits in this procedure, see "To Configure BEA WebLogic Server as J2EE Container 1 on Protected Resource 1" on page 185.

1 **As a root user, log into the ProtectedResource–1 host machine.**

2 **Ensure that your system is properly patched.**

Refer to the BEA web site to make sure that your system has the recommended patches.

3 **Create a directory into which you can download the WebLogic Server bits and change into it.**

```
# mkdir /export/BEAWL92
# cd /export/BEAWL92
```

4 **Download the WebLogic Server bits from `http://commerce.bea.com/`.**

For this deployment, we download the Solaris version.

```
# ls -al

total 294548
drwxr-xr-x   2 root     root         512 Aug  7 13:23 .
drwxr-xr-x   3 root     sys          512 Aug  7 13:16 ..
-rw-r--r--   1 root     root     722048346 Aug  7 13:24 portal920_solaris32.bin
```

5 **Run the installer.**

```
# ./portal920_solaris32.bin
```

6 **When prompted, do the following:**

Accept the License agreement	Select Yes and click Next.
Create a new BEA Home	Type **/usr/local/bea** and click Next.
Select "Custom"	Click Next.

`Deselect the following:`
`- Workshop for WebLogic Platform`
`- WebLogic Portal`
	Click Next.

`Choose Product Installation Directories`	Type **`/usr/local/bea/weblogic92`** and click Next.
`Installation Complete`	Deselect `Run Quickstart` and click Done.

7 **Verify that the application was correctly installed.**

```
# cd /usr/local/bea
# ls -al

total 34
drwxr-xr-x   6 root     root         512 Sep 13 14:26 .
drwxr-xr-x   3 root     root         512 Sep 13 14:22 ..
-rwxr-xr-x   1 root     root         851 Sep 13 14:26 UpdateLicense.sh
-rw-r--r--   1 root     root          14 Sep 13 14:26 beahomelist
drwxr-xr-x   6 root     root         512 Sep 13 14:26 jdk150_04
-rw-r--r--   1 root     root        7818 Sep 13 14:26 license.bea
drwxr-xr-x   2 root     root         512 Sep 13 14:26 logs
-rw-r--r--   1 root     root         947 Sep 13 14:26 registry.xml
drwxr-xr-x   3 root     root         512 Sep 13 14:26 utils
drwxr-xr-x  10 root     root         512 Sep 13 14:26 weblogic92
```

To Configure BEA WebLogic Server as J2EE Container 1 on Protected Resource 1

After installing the bits, WebLogic Server must be configured.

Before You Begin This procedure assumes you have just completed "To Install BEA WebLogic Server as J2EE Container 1 on Protected Resource 1" on page 184.

1 **Run the WebLogic Server configuration script.**

```
# cd /usr/local/bea/weblogic92/common/bin
# ./config.sh
```

2 **When prompted, do the following:**

`Select "Create a new Weblogic domain"`	Click Next.
`Select "Generate a domain configured automatically to support the following BEA products:"`	Click Next.

`Configure Administrator Username and Password`	Enter the following and click Next. ■ Username: **`weblogic`** ■ Password: **`w3bl0g1c`**
`Select "Prduction Mode" and "BEA Supplied JDK's" (Sun SDK 1.5.0_04@/usr/local/bea/jdk150_04)`	Click Next.
`Customize Environment and Services Settings`	Select yes and click Next.
`Configure the Administration Server`	Accept the default values and click Next.
`Configure Managed Servers`	Select Add, enter the following values, and click Next. ■ Name: **`ApplicationServer-1`** ■ Listen Port: **`1081`**
`Configure Clusters`	Accept the default values and click Next.
`Configure Machines`	Select the Unix Machine tab, then select Add, type **`ProtectedResource-1`**, and click Next.
`Assign Servers to Machines`	From the left panel select *AdminServer ApplicationServer-1*. From the right panel select *ProtectedResource-1*. Click - -> and then click Next.
`Review WebLogic Domain`	Click Next.
`Create WebLogic Domain`	Add the following and click Create. ■ Domain name: **`ProtectedResource-1`** ■ Domain Location: **`/usr/local/bea/user_projects/domains`** (default)
`Creating Domain`	Click Done.

3 Start the WebLogic administration server.

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1
# ./startWebLogic.sh
```

When prompted, type the following credentials.

Username	**`weblogic`**
Password	**`w3bl0g1c`**

4 Run the `netstat` command to verify that the port is open and listening.

```
# netstat -an | grep 7001

XXX.XX.XX.151.7001        *.*                0      0 49152      0 LISTEN
XXX.X.X.1.7001            *.*                0      0 49152      0 LISTEN
```

Note – You can also access the administration console by pointing a web browser to `http://protectedresource-1.example.com:7001/console`.

---

5 **Change to the `AdminServer` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/servers/AdminServer
```

6 **Create a security directory and change into it.**

```
# mkdir security
# cd security
```

7 **Create a `boot.properties` file for the WebLogic Server administration server administrator credentials.**

The administrative user and password are stored in `boot.properties`. Application Server 1 uses this information during startup. WebLogic Server encrypts the file, so there is no security risk even if you enter the user name and password in clear text.

```
# cat > boot.properties
username=weblogic
password=w3bl0g1c
```

Hit Control D to terminate the command

```
^D
```

8 **Restart WebLogic to encrypt the username and password in `boot.properties`.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/bin
# ./stopWebLogic.sh
# ./startWebLogic.sh
```

9 **Start the managed servers.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/bin
# ./startManagedWebLogic.sh ApplicationServer-1 t3://localhost:7001
```

You will be prompted for the administrative user credentials.

Username	**`weblogic`**
Password	**`w3bl0g1c`**

10 **Change to the `ApplicationServer-1` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/
  servers/ApplicationServer-1
```

11 Create a security directory and change into it.

```
# mkdir security
# cd security
```

12 Create a `boot.properties` file for the WebLogic Server managed server administrator credentials.

The administrative user and password are stored in `boot.properties`. The ApplicationServer–1 managed server uses this information during startup. WebLogic Server encrypts the file, so there is no security risk even if you enter the user name and password in clear text.

```
# cat > boot.properties
username=weblogic
password=w3bl0g1c
```

Hit Control D to terminate the command

```
^D
```

13 Restart the managed server.

```
# cd /usr/local/bea/user_projects/domains/
  ProtectedResource-1/bin
# ./stopManagedWebLogic.sh ApplicationServer-1
   t3://localhost:7001
# ./startManagedWebLogic.sh ApplicationServer-1
   t3://localhost:7001
```

14 Run the `netstat` command to verify that the port is open and listening.

```
# netstat -an | grep 1081

XXX.X.X.1.1081            *.*               0      0 49152      0 LISTEN
XXX.XX.XX.151.1081        *.*               0      0 49152      0 LISTEN
```

15 Access `http://ProtectedResource-1.example.com:7001/console` from a web browser.

16 Login to the BEA WebLogic Server as the administrator.

Username	**weblogic**
Password	**w3bl0g1c**

17 Click `servers`.

On the Summary of Servers page, verify that both *AdminServer (admin)* and *ApplicationServer-1* are running and OK.

18 Log out of the console.

19 **Log out of the ProtectedResource–1 host machine.**

▼ To Install the J2EE Policy Agent 1 on Application Server 1

Before You Begin You must stop both the WebLogic Server 1 instance and the WebLogic Server 1 administration server before beginning the installation process.

1 **As a root user, log into the ProtectedResource–1 host machine.**

2 **Stop the WebLogic Server 1 administration server and the WebLogic Server 1 instance.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/bin
# ./stopManagedWebLogic.sh ApplicationServer-1 t3://localhost:7001
# ./stopWebLogic.sh
```

3 **Ensure that your system is properly patched.**

Read the appropriate policy agent Release Notes for your web container to determine the latest patches you might need to install before beginning installation. In this case, no patch is required.

Note – You can search for patches directly at `http://sunsolve.sun.com`. Navigate to the PatchFinder page, enter the patch number, click Find Patch, and download the appropriate patch.

---

4 **Change into the** `J2EEPA1` **directory.**

```
# cd /export/J2EEPA1
```

5 **Download the J2EE policy agent bits for WebLogic Server from** `http://www.sun.com/download/index.jsp`**.**

```
# ls -al

total 8692
drwxr-xr-x   2 root     root         512 Sep 13 13:19 .
drwxr-xr-x   5 root     sys          512 Aug 13 17:08 ..
-rw-r--r--   1 root     root     4433920 Sep 13 13:19 SJS_Weblogic_92_agent_2.2.tar
```

6 **Unpack the J2EE policy agent bits.**

```
# /usr/sfw/bin/gtar -xvf /export/J2EEPA1/SJS_Weblogic_92_agent_2.2.tar
```

Tip – Use the `gtar` command and not the `tar` command.

---

7 Run the J2EE policy agent installer.

```
# cd /export/J2EEPA1/j2ee_agents/am_wl92_agent/bin
# ./agentadmin --install
```

8 When prompted, provide the following information.

`Please read the following License Agreement carefully:`	Press Enter to continue. Continue to press Enter until you reach the end of the License Agreement.
`Enter startup script location.`	Enter **`/usr/local/bea/user_projects/domains/ProtectedResource-1/bin/startwebLogic.sh`**
`Enter the WebLogic Server instance name: [myserver]`	Enter **`ApplicationServer-1`**
`Access Manager Services Host:`	Enter **`LoadBalancer-3.example.com`**
`Access Manager Services port: [80]`	Enter **`7070`**
`Access Manager Services Protocol: [http]`	Accept the default value.
`Access Manager Services Deployment URI: [/amserver]`	Accept the default value.
`Enter the Agent Host name:`	Enter **`ProtectedResource-1.example.com`**
`Enter the WebLogic home directory: [/usr/local/bea/weblogic92]`	Accept the default value.
`Enter true if the agent is being installed on a Portal domain:`	Accept *false*, the default value.
`Enter the port number for Application Server instance [80]:`	Enter **`1081`**
`Enter the Preferred Protocol for Application instance [http]:`	Accept the default value.
`Enter the Deployment URI for the Agent Application [/agentapp]`	Accept the default value.
`Enter the Encryption Key [j8C9QteM1HtC2OhTTDh/f1LhT38wfX1F]:`	Accept the default value.
`Enter the Agent Profile Name:`	**`j2eeagent-1`**
`Enter the path to the password file:`	Enter **`/export/J2EEPA1/agent.pwd`**
`Are the Agent and Access Manager installed on the same instance of Application Server? [false]:`	Accept the default value.

`Verify your settings and decide from`
`the choices below:`
`1. Continue with Installation`
`2. Back to the last interaction`
`3. Start Over`
`4. Exit`
`Please make your selection [1]:`
	Accept the default value.

The installer runs and, when finished, creates a new file in the `bin` directory called `setAgentEnv_ApplicationServer-1.sh`.

9 **Modify the startup script `setDomainEnv.sh` to reference `setAgentEnv_ApplicationServer-1.sh`.**

Tip – Backup `setDomainEnv.sh` before you modify it.

---

a. **Change to the `bin` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/bin
```

b. **Insert the following line at the end of `setDomainEnv.sh`.**

```
. /usr/local/bea/user_projects/domains/ProtectedResource-1/
bin/setAgentEnv_ApplicationServer-1.sh
```

c. **Save `setDomainEnv.sh` and close the file.**

10 **Change permissions for `setAgentEnv_ApplicationServer-1.sh`.**

```
# chmod 755 setAgentEnv_ApplicationServer-1.sh
```

11 **Start the WebLogic Server administration server.**

```
# ./startWebLogic.sh &
```

Watch for startup errors.

▼ To Deploy the J2EE Policy Agent 1 Application

The agent application is a housekeeping application bundled with the agent binaries and used by the agent for notifications and other internal functionality. In order for the agent to function correctly, this application must be deployed on the agent-protected deployment container instance using the same URI that was supplied during the agent installation process. For example, during the installation process, if you entered `/agentapp` as the deployment URI for the agent application, use that same context path in the deployment container.

1 **Access `http://ProtectedResource-1.example.com:7001/console` from a web browser.**

2 **Log in to the WebLogic Server console as the administrator.**

Username	`weblogic`
Password	`w3bl0g1c`

3 **Under Domain Structure, click Deployments.**

4 **On the Summary of Deployments page, in the Change Center, click Lock & Edit.**

5 **Under Deployments, click Install.**

6 **On the Install Application Assistant page, click the `protectedresource-1.example.com` link.**

7 **In the field named Location: `protectedresource-1.example.com`, click the root directory.**

8 **Navigate to `/export/J2EEPA1/j2ee_agents/am_wl92_agent/etc`, the application directory.**

9 **Select `agentapp.war` and click Next.**

10 **In the Install Application Assistant page, choose *Install this deployment as an application* and click Next.**

11 **In the list of Servers, mark the checkbox for ApplicationServer-1 and click Next.**

12 **In the Optional Settings page, click Next.**

13 **Click Finish.**

14 **On the Settings for agentapp page, click Save.**

15 **In the Change Center, click Activate Changes.**

▼ To Start the J2EE Policy Agent 1 Application

Before You Begin This procedure assumes that you have just completed "To Deploy the J2EE Policy Agent 1 Application" on page 191.

1 **In the WebLogic Server console, on the Settings for agentapp page, click Deployments.**

2 **On the Summary of Deployments page, mark the `agentapp` checkbox and click Start > Servicing all requests.**

3 **On the Start Application Assistant page, click Yes.**

Note – You may encounter a JavaScript™ error as the agent application will not start until you start the WebLogic Server instance. In this case start the ApplicationServer-1 and perform the steps again.

▼ To Set Up the J2EE Policy Agent 1 Authentication Provider

Before You Begin This procedure assumes that you have just completed "To Start the J2EE Policy Agent 1 Application" on page 192.

1 **In the WebLogic Server console, on the Summary of Deployments page, under Domain Structure, click Security Realms.**

2 **On the Summary of Security Realms page, click Lock & Edit.**

3 **Click the `myrealm` link.**

4 **On the Settings for myrealm page, click the Providers tab.**

5 **Under Authentication Providers, click New.**

6 **On the Create a New Authentication Provider page, provide the following information and click OK.**

Name:	**`Agent-1`**
Type:	Select `AgentAuthenticator` from the drop down list.

`Agent-1` is now included in the list of Authentication Providers.

7 **In the list of Authentication Providers, click Agent-1.**

8 **In the Settings for Authentication Providers page, verify that the Control Flag is set to OPTIONAL.**

9 **In the navigation tree near the top of the page, click Providers.**

10 **In the list of Authentication Providers, click DefaultAuthenticator.**

11 **In the Settings for DefaultAuthenticator page, set the Control Flag to OPTIONAL and click Save.**

12 **In the navigation tree near the top of the page, click Providers again.**

13 **In the Change Center, click Activate Changes.**

14 **If indicated by the console, restart the servers.**

a. **Log out of the WebLogic Server console.**

b. **As a root user, log into the ProtectedResource–1 host machine.**

c. **Restart the administration server and the managed instance.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/bin
# ./stopManagedWebLogic.sh ApplicationServer-1 t3://localhost:7001
# ./stopWebLogic.sh
# ./startWebLogic.sh
# ./startManagedWebLogic.sh ApplicationServer-1 t3://localhost:7001
```

d. **Log out of the ProtectedResource–1 host machine.**

▼ To Edit the J2EE Policy Agent 1 `AMAgent.properties` File

1 **As a root user, log into the ProtectedResource–1 host machine.**

2 **Change to the directory that contains the `AMAgent.properties` file.**

```
# cd /export/J2EEPA1/j2ee_agents/am_wl92_agent/agent_001/config
```

Tip – Backup `AMAgent.properties` before you modify it.

3 **Make the following modifications to `AMAgent.properties`.**

a. **Set the following property.**

```
com.sun.identity.agents.config.bypass.principal[0] = weblogic
```

This ensures that the WebLogic administrator will be authenticated against WebLogic itself and not Access Manager.

b. **At end of the file, insert the following new property.**

```
com.sun.identity.session.resetLBCookie=true
```

You must add this property if session failover has been configured for Access Manager. If session failover is not configured and this property is added, it could negatively impact performance. If session failover is enabled for Access Manager and this property is not added, the session failover functionality will work properly but, the stickiness to the Access Manager server will not be maintained after failover occurs. This property is not required for web policy agents.

Tip – This property must be also be added to the Access Manager file, `AMConfig.properties` if added here.

4 **Save and close the file.**

5 **Log out of the ProtectedResource–1 host machine.**

9.1.3 Setting Up a Test for the J2EE Policy Agent 1

The BEA Policy Agent comes with a sample application created to help you test policies. For more information, see the file `readme.txt` in the `/export/J2EEPA1/j2ee_agents/am_wl92_agent/sampleapp` directory.

Use the following list of procedures as a checklist for setting up a test for the J2EE Policy Agent 1.

1. "To Deploy the J2EE Policy Agent 1 Sample Application" on page 195
2. "To Create a Test Referral Policy in the Access Manager Root Realm" on page 196
3. "To Create a Test Policy in the Access Manager User Realm" on page 197
4. "To Configure Properties for the J2EE Policy Agent 1 Sample Application" on page 198
5. "To Verify that J2EE Policy Agent 1 is Configured Properly" on page 200

▼ To Deploy the J2EE Policy Agent 1 Sample Application

1 **Access Application Server 1 at** `http://ProtectedResource-1.example.com:7001/console`**.**

2 **Log in to the WebLogic Server console as the administrator.**

Username	**`weblogic`**
Password	**`w3bl0g1c`**

3 **On the Summary of Deployments page, click Lock & Edit.**

4 **Under Domain Structure, click Deployments.**

5 **Under Deployments, click Install.**

6 **On the Install Application Assistant page, click the** `protectedresource-1.example.com` **link.**

7 **In the list for Location:** `protectedresource-1.example.com`**, click the root directory.**

8 **Navigate to the application directory (**`/export/J2EEPA1/j2ee_agents/am_wl9_agent/sampleapp/dist`**), select** `agentsample` **and click Next.**

9 **In the Install Application Assistant page, choose** ***Install this deployment as an application*** **and click Next.**

10 **In the list of Servers, mark the checkbox for** `ApplicationServer-1` **and click Next.**

11 **On the Optional Settings page, click Next to accept the default settings.**

12 **On the Review Your Choices page, click Finish.**

The Target Summary section indicates that the module `agentsample` will be installed on the target `ApplicationServer-1`.

13 **On the Settings for agentsample page, click Save.**

14 **On the Settings for agentsample page, click Activate Changes.**

15 **Under Domain Structure, click Deployments.**

16 **In the Deployments list, mark the checkbox for** `agentsample` **and click Start > Servicing All Requests.**

17 **On the Start Application Assistant page, click Yes.**

The state of the deployment changes from Prepared to Active.

18 **Log out of the Application Server 1 console.**

▼ To Create a Test Referral Policy in the Access Manager Root Realm

1 **Access** `http://LoadBalancer-3.example.com:7070/amserver/UI/Login`**, the Access Manage load balancer, from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

Username	**`amadmin`**
Password	**`4m4dmin1`**

3 **Under the Access Control tab, click the** `example` **realm link.**

4 **Click the Policies tab.**

5 **Under Policies, click the Referral URL Policy for users realm link.**

6 **On the Edit Policy page, under Rules, click New.**

7 **On the resulting page, select URL Policy Agent (with resource name) and click Next.**

8 **On the resulting page, provide the following information and click Finish.**

Name:	`URL Policy for ApplicationServer-1`
Resource Name:	`http://protectedresource-1.example.com:1081/agentsample/*`

Note – Make sure the hostname is typed in lowercase.

9 **On the resulting page, click Save.**

▼ To Create a Test Policy in the Access Manager User Realm

Before You Begin This procedure assumes you have just completed "To Create a Test Referral Policy in the Access Manager Root Realm" on page 196.

1 **In the Access Manager console, under the Access Control tab, click the `users` realm link.**

2 **Click the Policies tab.**

3 **Under Policies, click New Policy.**

4 **In the Name field, enter `URL Policy for ApplicationServer-1`.**

5 **Under Rules, click New.**

6 **On the resulting page, make sure the default URL Policy Agent (with resource name) is selected and click Next.**

7 **On the resulting page, provide the following information and click Finish.**

Name:	`agentsample`
Parent Resource Name:	From the list, select `http://protectedresource-1.example.com:1081/agentsample/*`
Resource Name:	The value of this property is populated when you select the Parent Resource Name. It should read `http://protectedresource-1.example.com:1081/agentsample/*`.
GET	Mark this check box and verify that Allow is selected.

POST	Mark this check box and verify that Allow is selected.

The rule agentsample is now added to the list of Rules.

8 **Under Subjects, click New.**

9 **On the resulting page, select Access Manager Identity Subject and click Next.**

10 **On the resulting page, provide the following information and click Search.**

Name:	**agentsampleGroup**
Filter:	Select Group.

Manager-Group and Employee-Group are displayed in the Available list.

11 **Select Manager-Group and Employee-Group and click Add.**

The groups are now displayed in the Selected list.

12 **Click Finish.**

13 **Click OK.**

The new policy subject is included in the list of Policies.

14 **Log out of the Access Manager console.**

▼ To Configure Properties for the J2EE Policy Agent 1 Sample Application

Modify AMAgent.properties.

1 **Log in as a root user to the ProtectedResource–1 host machine.**

2 **Change to the config directory.**

```
# cd /export/J2EEPA1/j2ee_agents/am_wl92_agent/agent_001/config
```

Tip – Backup AMAgent.properties before you modify it.

3 **Set the following properties in AMAgent.properties.**

```
com.sun.identity.agents.config.notenforced.uri[0] =
   /agentsample/public/*
com.sun.identity.agents.config.notenforced.uri[1] =
   /agentsample/images/*
com.sun.identity.agents.config.notenforced.uri[2] =
   /agentsample/styles/*
com.sun.identity.agents.config.notenforced.uri[3] =
   /agentsample/index.html
com.sun.identity.agents.config.notenforced.uri[4] =
   /agentsample
com.sun.identity.agents.config.access.denied.uri =
   /agentsample/authentication/accessdenied.html
com.sun.identity.agents.config.login.form[0] =
   /agentsample/authentication/login.html
com.sun.identity.agents.config.login.url[0] =
   http://LoadBalancer-3.example.com:7070/
   amserver/UI/Login?realm=users
com.sun.identity.agents.config.privileged.attribute.
   type[0] = group
com.sun.identity.agents.config.privileged.attribute.
   tolowercase[group] = false
```

4 **Set these remaining properties as follows.**

Note – This is specific to this deployment example. For more information see "The agentadmin -getUuid command fails for amadmin user on Access Manager 7 with various agents (6452713)" in *Sun Java System Access Manager Policy Agent 2.2 Release Notes*.

a. **Retrieve the Universal IDs.**

They were saved in "To Create Manager and Employee Groups Using Access Manager for J2EE Policy Agent Test" on page 182.

b. **Convert all uppercase to lowercase and append a back slash (\) in front of each equal sign (=).**

- **Change** `id=Manager-Group,ou=group,o=users,ou=services,dc=example,dc=com` **to** `id\=manager-group,ou\=group,o\=users,ou\=services,dc\=example,dc\=com`**.**
- **Change** `id=Employee-Group,ou=group,o=users,ou=services,dc=example,dc=com` **to** `id\=employee-group,ou\=group,o\=users,ou\=services,dc\=example,dc\=com`**.**

c. **Set the properties.**

```
com.sun.identity.agents.config.privileged.attribute.
   mapping[id\=manager-group,ou\=group,o\=users,ou\=services,
   dc\=example,dc\=com] = am_manager_role
com.sun.identity.agents.config.privileged.attribute.
   mapping[id\=employee-group,ou\=group,o\=users,ou\=services,
   dc\=example,dc\=com] = am_employee_role
```

5 **Save** `AMAgent.properties` **and close it.**

6 **Restart the Application Server 1 administration server and managed instance.**

a. **Change to the `bin` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/bin
```

b. **Stop the managed instance.**

```
# ./stopManagedWebLogic.sh ApplicationsServer-1 t3://localhost:7001
```

c. **Stop the administration server.**

```
# ./stopWebLogic.sh
```

d. **Start the administration server.**

```
# ./startWebLogic.sh &
```

e. **Start the managed instance.**

```
# ./startManagedWebLogic.sh ApplicationServer-1 t3://localhost:7001 &
```

7 **Log out of the ProtectedResource-1 host machine.**

▼ To Verify that J2EE Policy Agent 1 is Configured Properly

Use these steps to access the agent sample application and test policies against it.

1 **Access `http://protectedresource-1.example.com:1081/agentsample/index.html`, the sample application URL, from a web browser.**

The Sample Application welcome page is displayed.

2 **Click the J2EE Declarative Security link.**

3 **On the resulting page, click Invoke the Protected Servlet.**

You are redirected to the Access Manager login page.

4 **Log in to the Access Manager console as `testuser1`.**

Username	**`testuser1`**
Password	**`password`**

If you can successfully log in as `testuser1` and the J2EE Policy Agent Sample Application page is displayed, the first part of the test has succeeded and authentication is working as expected.

5 **Click the J2EE Declarative Security link again.**

6 **On the resulting page, click Invoke the Protected Servlet.**

If the Success Invocation message is displayed, the second part of the test has succeeded as the sample policy for the `manager` role has been enforced as expected.

7 **Click the J2EE Declarative Security link to return.**

8 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

If the Failed Invocation message is displayed, the third part of the test has succeeded as the sample policy for the `employee` role has been enforced as expected.

9 **Log out and close the browser.**

10 **In a new browser session, access `http://protectedresource-1.example.com:1081/agentsample/index.html`, the sample application URL, again.**

The Sample Application welcome page is displayed.

11 **Click the J2EE Declarative Security link.**

12 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

You are redirected to the Access Manager login page.

13 **Log in to the Access Manager console as `testuser2`.**

Username	**`testuser2`**
Password	**`password`**

Note – The Failed Invocation message is displayed. This is a known issue.

14 **Click the J2EE Declarative Security link.**

15 **On the resulting page, click Invoke the Protected EJB via an Unprotected.**

The Successful Invocation message is displayed as the sample policy for the `employee` role has been enforced as expected.

16 **Click the J2EE Declarative Security link to return.**

17 **On the resulting page, click Invoke the Protected Servlet.**

If the Access to Requested Resource Denied message is displayed, this part of the test has succeeded as the sample policy for the `manager` role has been enforced as expected.

18 Log out and close the browser.

9.1.4 Configuring the J2EE Policy Agent 1 to Communicate Over SSL

Configure the J2EE policy agent to point to the secure port of the Access Manager Load Balancer 3. Use the following list of procedures as a checklist for your configurations.

1. "To Configure the J2EE Policy Agent 1 for SSL Communication" on page 202
2. "To Import the Certificate Authority Root Certificate into the Application Server 1 Keystore" on page 203
3. "To Verify that J2EE Policy Agent 1 is Configured Properly" on page 204
4. "To Configure the J2EE Policy Agent 1 to Access the Distributed Authentication User Interface" on page 206

▼ To Configure the J2EE Policy Agent 1 for SSL Communication

1 Log in as a root user to the ProtectedResource–1 host machine.

2 Change to the directory that contains the `AMAgent.properties` file.

```
# cd /export/J2EEPA1/j2ee_agents/am_wl92_agent/agent_001/config
```

Tip – Backup `AMAgent.properties` before you modify it.

---

3 Modify these properties in `AMAgent.properties` as follows.

```
com.sun.identity.agents.config.login.url[0] =
   https://LoadBalancer-3.example.com:9443/amserver/UI/Login?realm=users
com.sun.identity.agents.config.cdsso.cdcservlet.url[0] =
   https://LoadBalancer-3.example.com:9443/amserver/cdcservlet
com.sun.identity.agents.config.cdsso.trusted.id.provider[0] =
   https://LoadBalancer-3.example.com:9443/amserver/cdcservlet
com.iplanet.am.naming.url=
   https://LoadBalancer-3.example.com:9443/amserver/namingservice
com.iplanet.am.server.protocol=https
com.iplanet.am.server.port=9443
```

4 Save `AMAgent.properties` and close the file.

▼ To Import the Certificate Authority Root Certificate into the Application Server 1 Keystore

The Certificate Authority (CA) root certificate enables the J2EE policy agent to trust the certificate from the Access Manager Load Balancer 3, and to establish trust with the certificate chain that is formed from the CA to the certificate. Import the same CA root certificate used in "To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer" on page 108.

Before You Begin This procedure assumes you have just completed "To Configure the J2EE Policy Agent 1 for SSL Communication" on page 202. In this example, the file is `/export/software/ca.cer`.

1 **Change to the directory where the `cacerts` keystore is located.**

```
# cd /usr/local/bea/jdk150_04/jre/lib/security
```

Tip – Backup `cacerts` before you modify it.

2 **Import the root certificate.**

```
# /usr/local/bea/jdk150_04/bin/keytool -import
  -trustcacerts -alias OpenSSLTestCA -file /export/software/ca.cer
  -keystore /usr/local/bea/jdk150_04/jre/lib/security/cacerts
  -storepass changeit

Owner: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
 O=Sun, L=Santa Clara, ST=California, C=US
Issuer: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
 O=Sun, L=Santa Clara, ST=California, C=US
Serial number: 97dba0aa26db6386
Valid from: Tue Apr 18 07:55:19 PDT 2006
 until: Tue Jan 13 06:55:19 PST 2009
Certificate fingerprints:
    MD5: 9F:57:ED:B2:F2:88:B6:E8:0F:1E:08:72:CF:70:32:06
    SHA1: 31:26:46:15:C5:12:5D:29:46:2A:60:A1:E5:9E:28:64:36:80:E4:70
Trust this certificate? [no]: yes
Certificate was added to keystore
```

3 **Verify that the certificate was successfully added to the keystore.**

```
# /usr/local/bea/jdk150_04/bin/keytool -list
  -keystore /usr/local/bea/jdk150_04/jre/lib/security/cacerts
  -storepass changeit | grep -i openssl

opensslTestca, Sept 19, 2007, trustedCertEntry,
```

4 **Restart the Application Server 1 administration server and managed instance.**

a. **Change to the `bin` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/bin
```

b. **Stop the managed instance.**

```
# ./stopManagedWebLogic.sh ApplicationsServer-1 t3://localhost:7001
```

c. **Stop the administration server.**

```
# ./stopWebLogic.sh
```

d. **Start the administration server.**

```
# ./startWebLogic.sh &
```

e. **Start the managed instance.**

```
# ./startManagedWebLogic.sh ApplicationServer-1 t3://localhost:7001 &
```

5 **Log out of the ProtectedResource–1 host machine.**

▼ To Verify that J2EE Policy Agent 1 is Configured Properly

Use these steps to access the agent sample application and test the policies.

1 **Access `http://ProtectedResource-1.example.com:1081/agentsample/index.html`, the sample application URL, from a web browser.**

The Sample Application welcome page is displayed.

2 **Click the J2EE Declarative Security link.**

3 **On the resulting page, click Invoke the Protected Servlet.**

You are redirected to the Access Manager login page.

4 **Log in to the Access Manager console as `testuser1`.**

Username	**`testuser1`**
Password	**`password`**

If you can successfully log in as `testuser1` and the J2EE Policy Agent Sample Application page is displayed, this first part of the test has succeeded and authentication is working as expected.

5 **Click the J2EE Declarative Security link to go back.**

6 **On the resulting page, click Invoke the Protected Servlet.**

If the Success Invocation message is displayed, this second part of the test has succeeded as the sample policy for the `manager` role has been enforced as expected.

7 **Click the J2EE Declarative Security link to go back.**

8 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

If the Failed Invocation message is displayed, this third part of the test succeeded as the sample policy for the `employee` role has been enforced as expected.

9 **Log out and close the browser.**

10 **In a new browser session, go to** `http://ProtectedResource-1.example.com:1081/agentsample/index.html`**, the sample application URL.**

You are redirected to the Access Manager login page.

11 **Log in to the Access Manager console as** `testuser2`**.**

Username	**`testuser2`**
Password	**`password`**

Note – The Failed Invocation message is displayed. This is a known issue.

---

12 **Click the J2EE Declarative Security link.**

13 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

The Successful Invocation message is displayed. The sample policy for the `employee` role has been enforced as expected.

14 **Click the J2EE Declarative Security link to go back.**

15 **On the resulting page, click Invoke the Protected Servlet.**

If the Access to Requested Resource Denied message is displayed, this part of the test is successful as the sample policy for the `manager` role has been enforced as expected.

16 **Log out and close the browser.**

▼ To Configure the J2EE Policy Agent 1 to Access the Distributed Authentication User Interface

1. **Log in as a root user to the ProtectedResource–1 host machine.**

2. **Change to the directory that contains the `AMAgent.properties` file.**

   ```
   # cd /export/J2EEPA1/j2ee_agents/am_wl92_agent/agent_001/config
   ```

   Tip – Backup `AMAgent.properties` before you modify it.

3. **Set the following properties in `AMAgent.properties`.**

   ```
   com.sun.identity.agents.config.login.url[0] =
      https://LoadBalancer-4.example.com:9443/distAuth/UI/Login?realm=users
   ```

4. **Save `AMAgent.properties` and close it.**

5. **Restart the Application Server 1 managed instance.**

   a. **Change to the `bin` directory.**

   ```
   # cd /usr/local/bea/user_projects/domains/ProtectedResource-1/bin
   ```

   b. **Stop the managed instance.**

   ```
   # ./stopManagedWebLogic.sh ApplicationsServer-1 t3://localhost:7001
   ```

   c. **Start the managed instance.**

   ```
   # ./startManagedWebLogic.sh ApplicationServer-1 t3://localhost:7001
   ```

6. **Log out of the ProtectedResource–1 host machine.**

7. **Verify that the agent is configured properly.**

   a. **Access `http://protectedresource-1.example.com:1081/agentsample/index.html`, the sample application URL, from a web browser.**

   The Sample Application Welcome page is displayed.

   b. **Click the J2EE Declarative Security link.**

   c. **On the resulting page, click Invoke the Protected Servlet.**

   You are redirected to the Distributed Authentication User Interface at `https://loadbalancer-4.example.com:9443/distAuth/UI/Login`.

d. **(Optional) Double-click the gold lock in the lower left corner of the browser.**

   In the Properties page, you see the certificate for `LoadBalancer–4.example.com`.

e. **Log in to the Access Manager console as** `testuser1`**.**

Username	**`testuser1`**
Password	**`password`**

   If you can successfully log in as `testuser1` and the J2EE Policy Agent Sample Application page is displayed, user authentication worked through the Distributed Authentication User Interface and the agent is configured properly.

f. **Log out of the console.**

9.2 Configuring Protected Resource 2

We will install Sun Java^{TM} System Web Server, BEA WebLogic Server, a web policy agent, and a J2EE policy agent on the ProtectedResource–2 host machine. The policy agents are configured to access Load Balancer 3 as illustrated in the following figure.

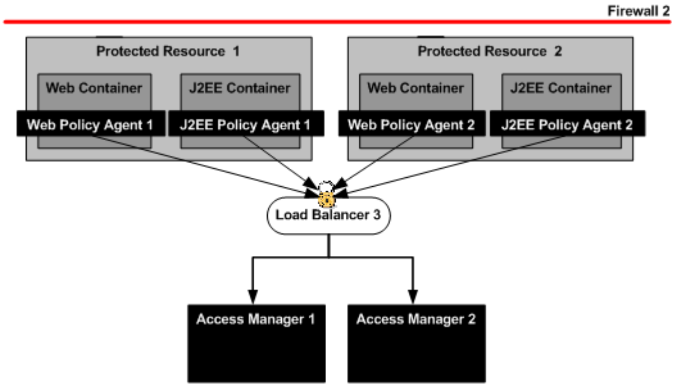


FIGURE 9–2 Protected Resources and Policy Agents

Use the following list of procedures as a checklist for configuring Protected Resource 2.

1. “9.2.1 Installing Web Container 2 and Web Policy Agent 2 on Protected Resource 2” on page 208
2. “9.2.2 Installing and Configuring the J2EE Container 2 and J2EE Policy Agent 2 on Protected Resource 2” on page 222
3. “9.2.3 Setting Up a Test for the J2EE Policy Agent 2” on page 235

4. “9.2.4 Configuring the J2EE Policy Agent 2 to Communicate Over SSL” on page 242

9.2.1 Installing Web Container 2 and Web Policy Agent 2 on Protected Resource 2

In this section, you install Sun Java System Web Server and a web policy agent on the ProtectedResource–2 host machine. Use the following list of procedures as a checklist.

1. “To Create an Agent Profile for Web Policy Agent 2” on page 208
2. “To Install Sun Java System Web Server as Web Container 2 on Protected Resource 2” on page 209
3. “To Install and Configure Web Policy Agent 2 on Protected Resource 2” on page 213
4. “To Import the Certificate Authority Root Certificate into the Web Server 2 Keystore” on page 218
5. “To Configure Policy for Web Policy Agent 2 on Protected Resource 2” on page 219
6. “To Verify that Web Policy Agent 2 is Working Properly” on page 222

▼ To Create an Agent Profile for Web Policy Agent 2

You create an agent profile in Access Manager to store authentication and configuration information that will be used by the policy agent to authenticate itself to Access Manager. Creating an agent profile also creates a custom user. The policy agent will, by default, use the account with the user identifier `UrlAccessAgent` to authenticate to Access Manager.

Note – Creating an agent profile is not a requirement for web policy agents. You can use the agent's default values and not change the user name; however, in certain cases, you might want to change these default values. For example, if you want to audit the interactions between multiple agents and the Access Manager server, you want be able to distinguish one agent from another. This would not be possible if all the agents used the same default agent user account. For more information, see the *Sun Java System Access Manager Policy Agent 2.2 User's Guide*.

---

1 Access `http://AccessManager-1.example.com:1080/amserver/UI/Login` from a web browser.

2 Log in to the Access Manager console as the administrator.

User Name: **`amadmin`**

Password: **`4m4dmin1`**

3 Under the Access Control tab, click `example`, the top-level Realm Name.

4 Click the Subjects tab.

5 **Click the Agents tab.**

6 **Click New to create a new agent profile.**

7 **On the resulting page, enter the following and click OK.**

ID	**webagent-2**
Password:	**web4gent2**
Password Confirm	**web4gent2**
Device State	Choose Active.

The new agent `webagent-2` is displayed in the list of agent users.

8 **Log out of the console.**

▼ To Install Sun Java System Web Server as Web Container 2 on Protected Resource 2

Download the Sun Java System Web Server bits and install the software on the ProtectedResource–2 host machine.

1 **As a root user, log into the ProtectedResource–2 host machine.**

2 **Install required patches if necessary.**

Results for your machines might be different. Read the latest version of the Web Server 7.0 Release Notes to determine if you need to install patches and, if so, what they might be. In this case, the Release Notes indicate that based on the hardware and operating system being used, patch 117461–08 is required.

a. **Run `patchadd` to see if the patch is installed.**

```
# patchadd -p | grep 117461–08
```

No results are returned which indicates that the patch is not yet installed on the system.

b. **Make a directory for downloading the patch you need and change into it.**

```
# mkdir /export/patches
# cd /export/patches
```

c. **Download the patches.**

You can search for patches directly at `http://sunsolve.sun.com`. Navigate to the PatchFinder page, enter the patch number, click Find Patch, and download the appropriate patch.

Note – Signed patches are downloaded as JAR files. Unsigned patches are downloaded as ZIP files.

---

d. **Unzip the patch file.**

```
# unzip 117461–08.zip
```

e. **Run `patchadd` to install the patches.**

```
# patchadd /export/patches/117461–08
```

f. **After installation is complete, run `patchadd` to verify that the patch was added successfully.**

```
# patchadd -p | grep 117461–08
```

In this example, a series of patch numbers are displayed, and the patch 117461–08 is present.

3 **Create a directory into which you can download the Web Server bits and change into it.**

```
# mkdir /export/ws7
# cd /export/ws7
```

4 **Download the Sun Java System Web Server 7.0 software from `http://www.sun.com/download/products.xml?id=45ad781d`.**

Follow the instructions on the Sun Microsystems Product Downloads web site for downloading the software. In this example, the software was downloaded to the `/export/ws7` directory.

```
# ls -al

total 294548
drwxr-xr-x   2 root     root         512 Aug  7 13:23 .
drwxr-xr-x   3 root     sys          512 Aug  7 13:16 ..
-rw-r--r--   1 root     root     150719523 Aug  7 13:24 sjsws-7_0-solaris-sparc.tar.gz
```

5 **Unpack the Web Server bits.**

```
# gunzip sjsws-7_0-solaris-sparc.tar.gz
# tar xvf sjsws-7_0-solaris-sparc.tar
```

6 **Run `setup`.**

```
# ./setup --console
```

7 **When prompted, provide the following information.**

`You are running the installation program`
`for the Sun Java System Web Server 7.0.`
`...`
`The installation program pauses as questions`
`are presented so you can read the`
`information and make your choice.`
`When you are ready to continue, press Enter.`
	Press Enter. Continue to press Enter when prompted.

`Have you read the Software License`
`Agreement and do you accept all the terms?`
	Enter **`yes`**.

`Sun Java System Web Server 7.0`
`Installation Directory [/sun/webserver7]`
	Enter **`/opt/SUNWwbsvr`**

`Specified directory /opt/SUNWwbsvr`
`does not exist.  Create Directory? [Yes/No]`
	Enter **`yes`**.

`Select Type of Installation`
 	
`1. Express`
`2. Custom`
`3. Exit`
`What would you like to do? [1]`
	Enter **`2`**.

`Component Selection`
 	
`1. Server Core`
`2. Server Core 64-biy Binaries`
`3. Administration Command Line Interface`
`4. Sample Applications`
`5. Language Pack`
`Enter the comma-separated list [1,2,3,4,5]`
	Enter **`1,3,5`**.

`Java Configuration`
`1. Install Java Standard Edition 1.5.0_09`
`2. Reuse existing Java SE 1.5.0_09 or greater`
`3. Exit`
`What would you like to do? [1]`
	Enter **`1`**.

`Administrative Options`
`1. Create an Administration Server and a`
`   Web Server Instance`
`2. Create an Administration Node`
`Enter your option. [1]`
	Enter **`1`**.

`Start servers during system startup. [yes/no]`	Enter **`no`**.
`Host Name [ProtectedResource-2.example.com]`	Accept the default value.
`SSL Port [8989]`	Accept the default value.
`Create a non-SSL Port? [yes/no]`	Enter **`no`**.

`Runtime User ID [webservd]`	Enter **`root`**.
`Administrator User Name [admin]`	Accept the default value.
`Administrator Password:`	Enter **`web4dmin`**.
`Retype Password:`	Enter **`web4dmin`**.
`Server Name [ProtectedResource-2.example.com]`	Accept the default value.
`Http Port [8080]`	Enter **`1080`**.

`Document Root Directory [/opt/SUNWwbsvr/`
`https-ProtectedResource-2.example.com/docs]`
	Accept the default value.

`Ready To Install`
`1. Install Now`
`2. Start Over`
`3. Exit Installation`
`What would you like to do?`
	Enter**`1`**.

When installation is complete, the following message is displayed:

```
Installation Successful.
```

8 **Start the Web Server administration server.**

```
# cd /opt/SUNWwbsvr/admin-server/bin
# ./startserv

server not running
Sun Java System Web Server 7.0 B12/04/2006 10:15
info: CORE3016: daemon is running as super-user
info: CORE5076: Using [Java HotSpot(TM) Server VM, Version 1.5.0_09]
  from [Sun M icrosystems Inc.]
info: WEB0100: Loading web module in virtual server [admin-server] at
  [/admingui ]
info: WEB0100: Loading web module in virtual server [admin-server] at
  [/jmxconne ctor]
 info: HTTP3072: admin-ssl-port: https://protectedresource-2.example.com:8989
  ready to accept requests
info: CORE3274: successful server startup
```

9 **Run `netstat` to verify that the port is open and listening.**

```
# netstat -an | grep 8989

*.8989               *.*                0      0 49152      0 LISTEN
```

10 **(Optional) Login to the Web Server administration console at `https://protectedresource-2.example.com:8989`.**

Username	**admin**
Password	**web4dmin**

You should see the Web Server console.

11 **(Optional) Log out of the Web Server console.**

12 **Start the Protected Resource 2 Web Server instance.**

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-2.example.com/bin
# ./startserv

server not running
Sun Java System Web Server 7.0 B12/04/2006 10:15
info: CORE3016: daemon is running as super-user
info: CORE5076: Using [Java HotSpot(TM) Server VM,
   Version 1.5.0_09] from [Sun Microsystems Inc.]
info: HTTP3072: http-listener-1: http://ProtectedResource-2.example.com:1080
   ready to accept requests
info: CORE3274: successful server startup
```

13 **Run `netstat` to verify that the port is open and listening.**

```
# netstat -an | grep 1080

*.1080               *.*                0      0 49152      0 LISTEN
```

14 **Access the Protected Resource 2 instance at `https://ProtectedResource-2.example.com:1080` using a web browser.**

You should see the default Web Server index page.

15 **Log out of the ProtectedResource–2 host machine.**

▼ To Install and Configure Web Policy Agent 2 on Protected Resource 2

Caution – Due to a known problem with this version of the Web Policy Agent, you must start an X-display session on the server host using a program such as Reflections X or VNC, even though you use the command-line installer. For more information about this known problem, see "On UNIX-based machines, all web agents require that the X11 DISPLAY variable be set properly." in *Sun Java System Access Manager Policy Agent 2.2 Release Notes*.

1 **As a root user, log into the ProtectedResource–2 host machine.**

2. **Ensure that your system is properly patched.**

   This should have been done in "To Install Sun Java System Web Server as Web Container 2 on Protected Resource 2" on page 209.

3. **Create a directory into which you can download the Web Server agent bits and change into it.**

```
# mkdir /export/WebPA2
# cd /export/WebPA2
```

4. **Download the web policy agent for Web Server from** `http://www.sun.com/download/`**.**

```
# ls -al

total 294548
drwxr-xr-x   2 root     root         512 Aug  7 13:23 .
drwxr-xr-x   3 root     sys          512 Aug  7 13:16 ..
-rw-r--r--   1 root     root     150719523 Aug  7 13:24 sjsws_v70_SunOS_agent.zip
```

5. **Unzip the downloaded file.**

```
# unzip sjsws_v70_SunOS_agent.zip
```

6. **Change the permissions for the resulting** `agentadmin` **binary.**

```
# cd /export/WebPA2/web_agents/sjsws_agent/bin
# chmod +x agentadmin
```

7. **Verify that** `crypt_util` **has execute permission before running the installer.**

```
# cd /export/WebPA2/web_agents/sjsws_agent/bin
# chmod +x crypt_util
```

8. **Create a temporary file for the password that will be required during agent installation.**

```
# echo web4gent2 > /export/WebPA2/pwd.txt
# cat /export/WebPA2/pwd.txt
```

9. **Run the agent installer.**

```
# ./agentadmin --install
```

10. **When prompted, do the following.**

`Do you completely agree with all the terms and conditions of this License Agreement (yes/no): [no]:`	Type **yes** and press Enter.

`*********************************************`
Welcome to the Access Manager Policy Agent for
Sun Java System Web Server If the Policy Agent is
used with Federation Manager services, User needs to
enter information relevant to Federation Manager.
`***************************************************`	

Enter the complete path to the directory which is used by Sun Java System Web Server to store its configuration Files. This directory uniquely identifies the Sun Java System Web Server instance that is secured by this Agent. [? : Help, ! : Exit] Enter the Sun Java System Web Server Config Directory Path [/var/opt/SUNWwbsvr7/ https-ProtectedResource-2.example.com/config]:	Type **/opt/SUNWwbsvr/https-ProtectedResource-2.example.com/config** and press Enter.
Enter the fully qualified host name of the server where Access Manager Services are installed. [? : Help, < : Back, ! : Exit] Access Manager Services Host:	Type **LoadBalancer-3.example.com** and press Enter.
Enter the port number of the Server that runs Access Manager Services. [? : Help, < : Back, ! : Exit] Access Manager Services port [80]:	Type **9443** and press Enter.
Enter http/https to specify the protocol used by the Server that runs Access Manager services. [? : Help, < : Back, ! : Exit] Access Manager Services Protocol [http]:	Type **https** and press Enter.
Enter the Deployment URI for Access Manager Services. [? : Help, < : Back, ! : Exit] Access Manager Services Deployment URI [/amserver]:	Press Enter to accept the default /amserver.
Enter the fully qualified host name on which the Web Server protected by the agent is installed. [? : Help, < : Back, ! : Exit] Enter the Agent Host name:	Type **ProtectedResource-2.example.com** and press Enter.
Enter the preferred port number on which the Web Server provides its services. [? : Help, < : Back, ! : Exit] Enter the port number for Web Server instance [80]:	Type **1080** and press Enter.

Select http or https to specify the protocol used by the Web server instance that will be protected by Access Manager Policy Agent. [? : Help, < : Back, ! : Exit] Enter the Preferred Protocol for Web Server instance [http]:	Press Enter to accept the default `http`.
Enter a valid Agent profile name. Before proceeding with the agent installation, please ensure that a valid Agent profile exists in Access Manager. [? : Help, < : Back, ! : Exit] Enter the Agent Profile name [UrlAccessAgent]:	Type **`webagent-2`** and press Enter.
Enter the path to a file that contains the password to be used for identifying the Agent. [? : Help, < : Back, ! : Exit] Enter the path to the password file:	Type **`/export/WebPA2/pwd.txt`** and press Enter.
--- SUMMARY OF YOUR RESPONSES --- Sun Java System Web Server Config Directory : /opt/SUNWwbsvr/https-ProtectedResource-2.example.com/ config Access Manager Services Host : LoadBalancer-3.example.com Access Manager Services Port : 9443 Access Manager Services Protocol : https Access Manager Services Deployment URI : /amserver Agent Host name : ProtectedResource-2.example.com Web Server Instance Port number : 1080 Protocol for Web Server instance : http Agent Profile name : webagent-2 Agent Profile Password file name : /export/WebPA2/pwd.txt Verify your settings above and decide from the choices below. 1. Continue with Installation 2. Back to the last interaction 3. Start Over 4. Exit Please make your selection [1]:	Type **`1`** and press Enter.

```
Creating directory layout and configuring Agent
file for Agent_001 instance ...DONE.

Reading data from file /export/WebPA2/pwd.txt and
encrypting it ...DONE.

Generating audit log file name ...DONE.

Creating tag swapped AMAgent.properties file for
instance Agent_001 ...DONE.

Creating a backup for file
/opt/SUNWwbsvr/https-ProtectedResource-2.example.com/
   config/obj.conf ...DONE.

Creating a backup for file
/opt/SUNWwbsvr/https-ProtectedResource-2.example.com/
   config/magnus.conf ...DONE.

Adding Agent parameters to
/opt/SUNWwbsvr/https-ProtectedResource-2.example.com/
   config/magnus.conf file ...DONE.

Adding Agent parameters to
/opt/SUNWwbsvr/https-ProtectedResource-2.example.com/
   config/obj.conf file ...DONE.


SUMMARY OF AGENT INSTALLATION
-----------------------------
Agent instance name: Agent_001
Agent Configuration file location:
/export/WebPA2/web_agents/sjsws_agent/Agent_001/
  config/AMAgent.properties
Agent Audit directory location:
/export/WebPA2/web_agents/sjsws_agent/Agent_001/
  logs/audit
Agent Debug directory location:
/export/WebPA2/web_agents/sjsws_agent/Agent_001/
  logs/debug

Install log file location:
/export/WebPA2/web_agents/sjsws_agent/logs/audit/
  install.log

Thank you for using Access Manager Policy Agent
```

11 Modify the `AMAgent.properties` file.

Tip – Backup `AMAgent.properties` before you modify it.

a. **Change to the `config` directory.**

```
# cd /export/WebPA2/web_agents/sjsws_agent/Agent_001/config
```

b. **Set the values of the following properties as shown.**

```
com.sun.am.policy.am.login.url = https://LoadBalancer-3.
   example.com:9443/amserver/UI/Login?realm=users
com.sun.am.load_balancer.enable = true
```

c. **Save the file and close it.**

12 **Restart the Protected Resource 2 Web Server instance.**

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-2.example.com/bin
# ./stopserv; ./startserv

server has been shutdown
Sun Java System Web Server 7.0 B12/04/2006 10:15
info: CORE3016: daemon is running as super-user info:
CORE5076: Using [Java HotSpot(TM) Server VM, Version 1.5.0_09]
  from [Sun Microsystems Inc.]
info: HTTP3072: http-listener-1: http://ProtectedResource-2.example.com:1080
  ready to accept requests
```

13 **Log out of the ProtectedResource–2 host machine.**

To Import the Certificate Authority Root Certificate into the Web Server 2 Keystore

The web policy agent on Protected Resource 2 connects to Access Manager through Load Balancer 3. The load balancer is SSL-enabled, so the agent must be able to trust the load balancer SSL certificate to establish the SSL connection. For this reason, import the root certificate of the Certificate Authority (CA) that issued the Load Balancer 3 SSL server certificate into the policy agent keystore.

Before You Begin Import the same CA root certificate used in "To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer" on page 108.

1 **As a root user, log into the ProtectedResource–2 host machine.**

2 **Copy the CA root certificate into a directory.**

In this example, the file is `/export/software/ca.cer`.

3 Import the CA root certificate into the Java keystore.

```
# /opt/SUNWwbsvr/jdk/jre/bin/keytool -import -trustcacerts
  -alias OpenSSLTestCA -file /export/software/ca.cer
  -keystore /opt/SUNWwbsvr/jdk/jre/lib/security/cacerts -storepass changeit

Owner: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
O=Sun,L=Santa Clara, ST=California C=US
Issuer: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
O=Sun,L=Santa Clara, ST=California C=US
Serial number: 97dba0aa26db6386
Valid from: Tue Apr 18 07:66:19 PDT 2006 until: Tue Jan 13 06:55:19
PST 2009
Certificate fingerprints:
MD5: 9f:57:ED:B2:F2:88:B6:E8:0F:1E:08:72:CF:70:32:06
SHA1: 31:26:46:15:C5:12:5D:29:46:2A:60:A1:E5:9E:26:64:36:80:E4:70
Trust this certificate: [no] yes
Certificate was added to keystore.
```

4 Verify that the CA root certificate was imported into the keystore.

```
# /opt/SUNWwbsvr/jdk/jre/bin/keytool -list
  -keystore /opt/SUNWwbsvr/jdk/jre/lib/security/cacerts
  -storepass changeit | grep -i open

openssltestca, Sep 10, 2007, trustedCertEntry,
```

5 Restart the Protected Resource 2 Web Server instance.

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-2.example.com/bin
# ./stopserv
# ./startserv

server has been shutdown
Sun Java System Web Server 7.0 B12/04/2006 10:15
info: CORE3016: daemon is running as super-user
info: CORE5076: Using [Java HotSpot(TM) Server VM,
Version 1.5.0_09] from [Sun Microsystems Inc.]
info: HTTP3072: http-listener-1: http://ProtectedResource-2.
example.com:1080 ready to accept requests
info: CORE3274: successful server startup
```

6 Log out of the ProtectedResource–2 host machine.

To Configure Policy for Web Policy Agent 2 on Protected Resource 2

Use the Access Manager console to configure policy for Web Policy Agent 2. This policy will be used to verify that Web Policy Agent 2 is working properly. You will modify this policy later when we add a load balancer in front of it.

1 **Access** `http://AccessManager-1.example.com:1080/amserver/UI/Login` **from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

Username	**`amadmin`**
Password	**`4m4dmin1`**

3 **Create a referral policy in the top-level realm.**

a. **Under the Access Control tab, click the top-level realm,** `example`**.**

b. **Click the Policies tab.**

c. **Click** ***Referral URL Policy for users realm*****.**

d. **On the same page, in the Rules section, click New.**

e. **On the resulting page, select URL Policy Agent (with resource name) as a Service Type and click Next.**

f. **Provide the following information on the resulting page.**

Name:	**`URL Rule for ProtectedResource-2`**
Resource Name:	**`http://ProtectedResource-2.example.com:1080/*`**

g. **Click Finish.**

h. **Click Save.**

i. **On the Edit Policy page, click Back to Policies.**

Under the Policies tab for the `example` realm, you should see the policy named *Referral URL Policy for users realm* with `http://ProtectedResource-2.example.com:1080/*` added in the list of protected resources.

4 **Create a policy in the** `users` **realm.**

The `users` realm was previously created in "7.2 Creating and Configuring a Realm for Test Users" on page 121.

a. **Click the Access Control tab.**

b. **Under Realms, click** `users`**.**

c. **Click the Policies tab.**

d. **Click New Policy.**

e. **On the New Policy page, provide the following information:**

Name:	`URL Policy for ProtectedResource-2`
Active:	Mark the Yes checkbox.

f. **On the same page, in the Rules section, click New.**

g. **Select a Service Type for the rule and click Next.**

URL Policy Agent (with resource name) is the only choice.

h. **On the resulting page, provide the following information:**

Name:	`URL Rule for ProtectedResource-2`
Resource Name:	Click `http://ProtectedResource-2.example.com:1080/*`, listed in the Parent Resource Name list, to add it to the Resource Name field.
GET:	Mark this checkbox, and select Allow.
POST:	Mark this checkbox, and select Allow.

i. **Click Finish.**

5 **Create a new subject in the `users` realm for testing.**

a. **On the New Policy page, in the Subjects section, click New.**

b. **Select Access Manager Identity Subject as the subject type and click Next.**

c. **Provide the following information on the resulting page.**

Name:	`Test Subject`
Filter:	Choose `User` and click Search. Two users are added to the Available list.
Available:	In the list, select `Test User1` and click Add.

d. **Click Finish.**

6 **Back on the New Policy page, click Create.**

Under the Policies tab, you should see the policy named *URL Policy for ProtectedResource-2*.

7 **Log out of the console.**

▼ To Verify that Web Policy Agent 2 is Working Properly

1. **Access** `http://ProtectedResource-2.example.com:1080` **from a web browser.**

2. **Log in to Access Manager as** `testuser1`**.**

Username	**`testuser1`**
Password	**`password`**

   You should see the default index page for Web Server 2 as `testuser1` was configured in the test policy to be allowed to access Protected Resource 2.

3. **Log out and close the browser.**

4. **Once again, access** `http://ProtectedResource-2.example.com:1080` **from a web browser.**

   Tip – If you are not redirected to the Access Manager login page for authentication, clear your browser's cache and cookies and try again.

5. **Log in to Access Manager as** `testuser2`**.**

Username	**`testuser2`**
Password	**`password`**

   You should see the message, *You're not authorized to view this page*, (or *Your client is not allowed to access the requested object*) as `testuser2` was not included in the test policy that allows access to Protected Resource 2.

9.2.2 Installing and Configuring the J2EE Container 2 and J2EE Policy Agent 2 on Protected Resource 2

In this section, you will download the BEA WebLogic Server bits and install this application server on the ProtectedResource–2 host machine. Additionally, you will download and install the appropriate J2EE policy agent, deploy the policy agent application, setup up an authentication provider, and modify the Bypass Principal List. Use the following list of procedures as a checklist for installing Application Server 2 and the J2EE Policy Agent 2.

1. "To Install BEA WebLogic Server as J2EE Container 2 on Protected Resource 2" on page 223
2. "To Configure BEA WebLogic Server as J2EE Container 2 on Protected Resource 2" on page 224
3. "To Create an Agent Profile for the J2EE Policy Agent 2" on page 228
4. "To Install the J2EE Policy Agent 2 on Application Server 2" on page 229
5. "To Deploy the J2EE Policy Agent 2 Application" on page 231

6. “To Start the J2EE Policy Agent 2 Application” on page 232
7. “To Set Up the J2EE Policy Agent 2 Authentication Provider” on page 233
8. “To Edit the J2EE Policy Agent 2 AMAgent.properties File” on page 234

▼ To Install BEA WebLogic Server as J2EE Container 2 on Protected Resource 2

BEA WebLogic Server is the application server used as the J2EE container on Protected Resource 2. After installing the bits in this procedure, see “To Configure BEA WebLogic Server as J2EE Container 2 on Protected Resource 2” on page 224.

1 **As a root user, log into the ProtectedResource–2 host machine.**

2 **Ensure that your system is properly patched.**

Refer to the BEA web site to make sure that your system has the recommended patches.

3 **Create a directory into which you can download the WebLogic Server bits and change into it.**

```
# mkdir /export/BEAWL92
# cd /export/BEAWL92
```

4 **Download the WebLogic Server bits from http://commerce.bea.com/.**

For this deployment, we download the Solaris version.

```
# ls -al

total 294548
drwxr-xr-x   2 root     root         512 Aug  7 13:23 .
drwxr-xr-x   3 root     sys          512 Aug  7 13:16 ..
-rw-r--r--   1 root     root     722048346 Aug  7 13:24 portal920_solaris32.bin
```

5 **Run the installer.**

```
# ./portal920_solaris32.bin
```

6 **When prompted, do the following:**

Accept the License agreement	Select Yes and click Next.
Create a new BEA Home	Type **/usr/local/bea** and click Next.
Select "Custom"	Click Next.
Deselect the following: - Workshop for WebLogic Platform - WebLogic Portal	Click Next.

`Choose Product Installation Directories`	Type **`/usr/local/bea/weblogic92`** and click Next.
`Installation Complete`	Deselect *Run Quickstart* and click Done.

7 Verify that the application was correctly installed.

```
# cd /usr/local/bea
# ls -al

total 34
drwxr-xr-x   6 root     root         512 Sep 13 14:26 .
drwxr-xr-x   3 root     root         512 Sep 13 14:22 ..
-rwxr-xr-x   1 root     root         851 Sep 13 14:26 UpdateLicense.sh
-rw-r--r--   1 root     root          14 Sep 13 14:26 beahomelist
drwxr-xr-x   6 root     root         512 Sep 13 14:26 jdk150_04
-rw-r--r--   1 root     root        7818 Sep 13 14:26 license.bea
drwxr-xr-x   2 root     root         512 Sep 13 14:26 logs
-rw-r--r--   1 root     root         947 Sep 13 14:26 registry.xml
drwxr-xr-x   3 root     root         512 Sep 13 14:26 utils
drwxr-xr-x  10 root     root         512 Sep 13 14:26 weblogic92
```

▼ To Configure BEA WebLogic Server as J2EE Container 2 on Protected Resource 2

After installing the bits, WebLogic Server must be configured for use as the J2EE container on Protected Resource 2.

Before You Begin This procedure assumes you have just completed "To Install BEA WebLogic Server as J2EE Container 2 on Protected Resource 2" on page 223.

1 Run the WebLogic Server configuration script.

```
# cd /usr/local/bea/weblogic92/common/bin
# ./config.sh
```

2 When prompted, do the following:

`Select "Create a new Weblogic domain"`	Click Next.
`Select "Generate a domain configured automatically to support the following BEA products:"`	Click Next.
`Configure Administrator Username and Password`	Enter the following and click Next. ■ Username: **`weblogic`** ■ Password: **`w3bl0g1c`**

`Select "Prduction Mode" and "BEA Supplied JDK's" (Sun SDK 1.5.0_04@/usr/local/bea/jdk150_04)`	Click Next.
`Customize Environment and Services Settings`	Select yes and click Next.
`Configure the Administration Server`	Accept the default values and click Next.
`Configure Managed Servers`	Select Add, enter the following values, and click Next. ■ Name: **`ApplicationServer-2`** ■ Listen Port: **`1081`**
`Configure Clusters`	Accept the default values and click Next.
`Configure Machines`	Select the Unix Machine tab, then select Add, type **`ProtectedResource-2`**, and click Next.
`Assign Servers to Machines`	From the left panel select *AdminServer ApplicationServer-2*. From the right panel select *ProtectedResource-2*. Click --> and then click Next.
`Review WebLogic Domain`	Click Next.
`Create WebLogic Domain`	Add the following and click Create. ■ Domain name: **`ProtectedResource-2`** ■ Domain Location: **`/usr/local/bea/user_projects/domains`** (default)
`Creating Domain`	Click Done.

3 Start the WebLogic administration server.

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2
# ./startWebLogic.sh
```

When prompted, type the following credentials.

Username	**`weblogic`**
Password	**`w3bl0g1c`**

4 Run the `netstat` command to verify that the port is open and listening.

```
# netstat -an | grep 7001

XXX.XX.XX.151.7001        *.*               0      0 49152      0 LISTEN
XXX.X.X.1.7001            *.*               0      0 49152      0 LISTEN
```

Note – You can also access the administration console by pointing a browser to `http://protectedresource-2.example.com:7001/console`.

---

5 **Change to the `AdminServer` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/servers/AdminServer
```

6 **Create a security directory and change into it.**

```
# mkdir security
# cd security
```

7 **Create a `boot.properties` file for the WebLogic Server administration server.**

The administrative user and password are stored in `boot.properties`. Application Server 2 uses this information during startup. WebLogic Server encrypts the file, so there is no security risk even if you enter the user name and password in clear text.

```
# cat > boot.properties
username=weblogic
password=w3bl0g1c
```

Hit Control D to terminate the command

```
^D
```

8 **Restart the WebLogic administration server to encrypt the username and password in `boot.properties`.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/bin
# ./stopWebLogic.sh
# ./startWebLogic.sh
```

9 **Start the ApplicationServer-2 managed instance.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/bin
# ./startManagedWebLogic.sh ApplicationServer-2 t3://localhost:7001
```

You will be prompted for the following credentials.

Username	**weblogic**
Password	**w3bl0g1c**

10 **Change to the `ApplicationServer-2` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/
  servers/ApplicationServer-2
```

11 Create a security directory and change into it.

```
# mkdir security
# cd security
```

12 Create a `boot.properties` file for the ApplicationServer-2 managed instance.

The administrative user and password are stored in `boot.properties`. The WebLogic Server managed instance uses this information during startup. WebLogic Server encrypts the file, so there is no security risk even if you enter the user name and password in clear text.

```
# cat > boot.properties
username=weblogic
password=w3bl0g1c
```

Hit Control D to terminate the command

```
^D
```

13 Restart the managed server.

```
# cd /usr/local/bea/user_projects/domains/
     ProtectedResource-2/bin
# ./stopManagedWebLogic.sh ApplicationServer-2
     t3://localhost:7001
# ./startManagedWebLogic.sh ApplicationServer-2
     t3://localhost:7001
```

14 Run the `netstat` command to verify that the port is open and listening.

```
# netstat -an | grep 1081

XXX.X.X.1.1081             *.*               0      0 49152      0 LISTEN
XXX.XX.XX.151.1081         *.*               0      0 49152      0 LISTEN
```

15 Access `http://ProtectedResource-2.example.com:7001/console` from a web browser.

16 Login to the BEA WebLogic Server as the administrator.

Username	**`weblogic`**
Password	**`w3bl0g1c`**

17 Click `servers`.

On the Summary of Servers page, verify that both *AdminServer (admin)* and *ApplicationServer-2* are running and OK.

18 Log out of the console.

19 Log out of the ProtectedResource–2 host machine.

▼ To Create an Agent Profile for the J2EE Policy Agent 2

This new agent profile will be used by J2EE Policy Agent 2 to authenticate to Access Manager.

1. **Access `http://LoadBalancer-3.example.com:7070/amserver/UI/Login`, the Access Manage load balancer, from a web browser.**

2. **Log in to the Access Manager console as the administrator.**

Username	**`amadmin`**
Password	**`4m4dmin1`**

3. **On the Access Control tab, click the top-level realm, `example`.**

4. **Click the Subjects tab.**

5. **Click the Agents tab.**

6. **On the Agent page, click New.**

7. **On the New Agent page, provide the following information and click OK.**

ID:	**`j2eeagent-2`**
Password:	**`j2ee4gent2`**
Password Confirm:	**`j2ee4gent2`**
Device State:	Choose `Active`.

   The new agent `j2eeagent–2` is displayed in the list of Agent Users.

8. **Log out of the Access Manager console.**

9. **As a root user, log into the ProtectedResource–2 host machine.**

10. **Create a directory into which you can download the J2EE policy agent bits and change into it.**

    ```
    # mkdir /export/J2EEPA2
    # cd /export/J2EEPA2
    ```

11. **Create a text file that contains the Agent Profile password.**

    The J2EE Policy Agent installer requires this file for installation.

    ```
    # cat > agent.pwd
    j2ee4gent2
    ```

    Hit Control D to terminate the command

```
^D
```

12. **Log out of the ProtectedResource–2 host machine.**

▼ To Install the J2EE Policy Agent 2 on Application Server 2

Before You Begin You must stop both the WebLogic Server 2 managed instance and the WebLogic Server 2 administration server before beginning the installation process.

1. **As a root user, log into the ProtectedResource–2 host machine.**

2. **Stop the WebLogic Server 2 administration server and the WebLogic Server 2 managed instance.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/bin
# ./stopManagedWebLogic.sh ApplicationServer-2 t3://localhost:7001
# ./stopWebLogic.sh
```

3. **Ensure that your system is properly patched.**

Read the appropriate policy agent Release Notes for your web container to determine the latest patches you might need to install before beginning installation. In this case, no patch is required.

Note – You can search for patches directly at `http://sunsolve.sun.com`. Navigate to the PatchFinder page, enter the patch number, click Find Patch, and download the appropriate patch.

---

4. **Change into the `J2EEPA2` directory.**

```
# cd /export/J2EEPA2
```

5. **Download the J2EE policy agent bits for WebLogic Server from `http://www.sun.com/download/index.jsp`.**

```
# ls -al

total 8692
drwxr-xr-x   2 root     root         512 Sep 13 13:19 .
drwxr-xr-x   5 root     sys          512 Aug 13 17:08 ..
-rw-r--r--   1 root     root     4433920 Sep 13 13:19 SJS_Weblogic_92_agent_2.2.tar
```

6. **Unpack the J2EE policy agent bits.**

```
# /usr/sfw/bin/gtar -xvf /export/J2EEPA2/SJS_Weblogic_92_agent_2.2.tar
```

Tip – Use the gtar command and not the tar command.

7. **Run the J2EE policy agent installer.**

```
# cd /export/J2EEPA2/j2ee_agents/am_wl92_agent/bin
# ./agentadmin --install
```

8. **When prompted, provide the following information.**

`Please read the following License Agreement carefully:`	Press Enter to continue. Continue to press Enter until you reach the end of the License Agreement.
`Enter startup script location.`	Enter . **/usr/local/bea/user_projects/domains/ProtectedResource-2/bin/startwebLogic.sh**
`Enter the WebLogic Server instance name: [myserver]`	Enter **ApplicationServer-2**.
`Access Manager Services Host:`	Enter **LoadBalancer-3.example.com**.
`Access Manager Services port: [80]`	Enter **7070**.
`Access Manager Services Protocol: [http]`	Accept the default value.
`Access Manager Services Deployment URI: [/amserver]`	Accept the default value.
`Enter the Agent Host name:`	**ProtectedResource-2.example.com**
`Enter the WebLogic home directory: [/usr/local/bea/weblogic92]`	Accept the default value.
`Enter true if the agent is being installed on a Portal domain:`	Accept *false*, the default value.
`Enter the port number for Application Server instance [80]:`	Enter **1081**.
`Enter the Preferred Protocol for Application instance [http]:`	Accept the default value.
`Enter the Deployment URI for the Agent Application [/agentapp]`	Accept the default value.
`Enter the Encryption Key [j8C9QteM1HtC2OhTTDh/f1LhT38wfX1F]:`	Accept the default value.
`Enter the Agent Profile Name:`	**j2eeagent-2**
`Enter the path to the password file:`	Enter **/export/J2EEPA2/agent.pwd**.

`Are the Agent and Access Manager installed on`
`the same instance of Application Server? [false]:`
	Accept the default value.

`Verify your settings and decide from`
`the choices below:`
`1. Continue with Installation`
`2. Back to the last interaction`
`3. Start Over`
`4. Exit`
`Please make your selection [1]:`
	Accept the default value.

The installer runs and, when finished, creates a new file in the `bin` directory called `setAgentEnv_ApplicationServer-2.sh`.

9 **Modify the startup script `setDomainEnv.sh` to reference `setAgentEnv_ApplicationServer-2.sh`.**

Tip – Backup `setDomainEnv.sh` before you modify it.

a. **Change to the `bin` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/bin
```

b. **Insert the following line at the end of `setDomainEnv.sh`.**

```
. /usr/local/bea/user_projects/domains/ProtectedResource-2/
bin/setAgentEnv_ApplicationServer-2.sh
```

c. **Save `setDomainEnv.sh` and close the file.**

10 **Change permissions for `setAgentEnv_ApplicationServer-2.sh`.**

```
# chmod 755 setAgentEnv_ApplicationServer-2.sh
```

11 **Start the WebLogic Server administration server.**

```
# ./startWebLogic.sh &
```

Watch for startup errors.

12 **Log out of the ProtectedResource–2 host machine.**

▼ To Deploy the J2EE Policy Agent 2 Application

The agent application is a housekeeping application bundled with the agent binaries and used by the agent for notifications and other internal functionality. In order for the agent to function correctly, this application must be deployed on the agent-protected deployment container instance using the same URI that was supplied during the agent installation process. For example, during the installation process, if you entered `/agentapp` as the deployment URI for the agent application, use that same context path in the deployment container.

1 **Access** `http://ProtectedResource-2.example.com:7001/console` **from a web browser.**

2 **Log in to the WebLogic Server console as the administrator.**

Username	**`weblogic`**
Password	**`w3bl0g1c`**

3 **Under Domain Structure, click Deployments.**

4 **On the Summary of Deployments page, in the Change Center, click Lock & Edit.**

5 **Under Deployments, click Install.**

6 **On the Install Application Assistant page, click the** `protectedresource-2.example.com` **link.**

7 **In the field named Location:** `protectedresource-2.example.com`**, click the root directory.**

8 **Navigate to** `/export/J2EEPA2/j2ee_agents/am_wl92_agent/etc`**, the application directory.**

9 **Select** `agentapp.war` **and click Next.**

10 **In the Install Application Assistant page, choose** ***Install this deployment as an application*** **and click Next.**

11 **In the list of Servers, mark the checkbox for ApplicationServer-2 and click Next.**

12 **In the Optional Settings page, click Next.**

13 **Click Finish.**

14 **On the Settings for agentapp page, click Save.**

15 **In the Change Center, click Activate Changes.**

▼ To Start the J2EE Policy Agent 2 Application

Before You Begin This procedure assumes that you have just completed "To Deploy the J2EE Policy Agent 2 Application" on page 231.

1 **In the WebLogic Server console, on the Settings for agentapp page, click Deployments.**

2 **On the Summary of Deployments page, mark the `agentapp` checkbox and click Start > Servicing all requests.**

3 **On the Start Application Assistant page, click Yes.**

Note – You may encounter a JavaScript error as the agent application will not start until you start the WebLogic Server. In this case start the ApplicationServer-2 managed instance and perform the steps again.

▼ To Set Up the J2EE Policy Agent 2 Authentication Provider

Before You Begin This procedure assumes that you have just completed "To Start the J2EE Policy Agent 2 Application" on page 232.

1 **In the WebLogic Server console, on the Summary of Deployments page, under Domain Structure, click Security Realms.**

2 **On the Summary of Security Realms page, click Lock & Edit.**

3 **Click the `myrealm` link.**

4 **On the Settings for myrealm page, click the Providers tab.**

5 **Under Authentication Providers, click New.**

6 **On the Create a New Authentication Provider page, provide the following information and click OK.**

Name:	**`Agent-2`**
Type:	Select `AgentAuthenticator` from the drop down list.

`Agent-2` is now included in the list of Authentication Providers.

7 **In the list of Authentication Providers, click Agent-2.**

8 **In the Settings for Authentication Providers page, verify that the Control Flag is set to OPTIONAL.**

9 **In the navigation tree near the top of the page, click Providers.**

10 **In the list of Authentication Providers, click DefaultAuthenticator.**

11 **In the Settings for DefaultAuthenticator page, set the Control Flag to OPTIONAL and click Save.**

12 **In the navigation tree near the top of the page, click Providers again.**

13 **In the Change Center, click Activate Changes.**

14 **(Optional) If indicated by the console, restart the servers.**

a. **Log out of the WebLogic Server console.**

b. **As a root user, log into the ProtectedResource–2 host machine.**

c. **Restart the administration server and the managed instance.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/bin
# ./stopManagedWebLogic.sh ApplicationServer-2 t3://localhost:7001
# ./stopWebLogic.sh
# ./startWebLogic.sh
# ./startManagedWebLogic.sh ApplicationServer-2 t3://localhost:7001
```

d. **Log out of the ProtectedResource–2 host machine.**

▼ To Edit the J2EE Policy Agent 2 `AMAgent.properties` File

1 **As a root user, log into the ProtectedResource–2 host machine.**

2 **Change to the directory that contains the `AMAgent.properties` file.**

```
# cd /export/J2EEPA2/j2ee_agents/am_wl92_agent/agent_001/config
```

Tip – Backup `AMAgent.properties` before you modify it.

3 **Make the following modifications to `AMAgent.properties`.**

a. **Set the following property.**

```
com.sun.identity.agents.config.bypass.principal[0] = weblogic
```

This ensures that the WebLogic administrator will be authenticated against WebLogic itself and not Access Manager.

b. **At end of the file, insert the following new property.**

```
com.sun.identity.session.resetLBCookie=true
```

You must add this property if session failover has been configured for Access Manager. If session failover is not configured and this property is added, it could negatively impact performance. If session failover is enabled for Access Manager and this property is not added, the session failover functionality will work properly but, the stickiness to the Access Manager server will not be maintained after failover occurs. This property is not required for web policy agents.

Caution – This property must be also be added to the Access Manager file, `AMConfig.properties` if added here.

4 **Save and close the file.**

5 **Log out of the ProtectedResource–2 host machine.**

9.2.3 Setting Up a Test for the J2EE Policy Agent 2

Use the following list of procedures as a checklist for setting up a test for the J2EE Policy Agent 2.

1. "To Deploy the J2EE Policy Agent 2 Sample Application" on page 235
2. "To Create a Test Referral Policy in the Access Manager Root Realm" on page 236
3. "To Create a Test Policy in the Access Manager User Realm" on page 237
4. "To Configure Properties for the J2EE Policy Agent 2 Sample Application" on page 238
5. "To Verify that J2EE Policy Agent 2 is Configured Properly" on page 240

▼ To Deploy the J2EE Policy Agent 2 Sample Application

The BEA Policy Agent comes with a sample application created to help test policies. For more information, see the file `readme.txt` in the `/export/J2EEPA2/j2ee_agents/am_wl92_agent/sampleapp` directory.

1 **Access** `http://ProtectedResource-2.example.com:7001/console` **from a web browser.**

2 **Log in to the WebLogic Server console as the administrator.**

Username	**`weblogic`**
Password	**`w3bl0g1c`**

3 **On the Summary of Deployments page, click Lock & Edit.**

4 **Under Domain Structure, click Deployments.**

5 **Under Deployments, click Install.**

6 **On the Install Application Assistant page, click the** `protectedresource-2.example.com` **link.**

7 **In the list for Location:** `protectedresource-2.example.com`**, click the root directory.**

8 **Navigate to the application directory (**`/export/J2EEPA2/j2ee_agents/am_wl9_agent/sampleapp/dist`**), select** `agentsample` **and click Next.**

9 **In the Install Application Assistant page, choose** ***Install this deployment as an application*** **and click Next.**

10 **In the list of Servers, mark the checkbox for** `ApplicationServer-2` **and click Next.**

11 **On the Optional Settings page, click Next to accept the default settings.**

12 **On the Review Your Choices page, click Finish.**

The Target Summary section indicates that the module `agentsample` will be installed on the target `ApplicationServer-2`.

13 **On the Settings for agentsample page, click Save.**

14 **On the Settings for agentsample page, click Activate Changes.**

15 **Under Domain Structure, click Deployments.**

16 **In the Deployments list, mark the checkbox for** `agentsample` **and click Start > Servicing All Requests.**

17 **On the Start Application Assistant page, click Yes.**

The state of the deployment changes from Prepared to Active.

18 **Log out of the console.**

▼ To Create a Test Referral Policy in the Access Manager Root Realm

1 **Access** `http://LoadBalancer-3.example.com:7070/amserver/UI/Login`**, the Access Manager load balancer, from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

Username	**`amadmin`**
Password	**`4m4dmin1`**

3 **Under the Access Control tab, click the** `example` **realm link.**

4 Click the Policies tab.

5 Under Policies, click the Referral URL Policy for users realm link.

6 On the Edit Policy page, under Rules, click New.

7 On the resulting page, select URL Policy Agent (with resource name) and click Next.

8 On the resulting page, provide the following information and click Finish.

Name:	**`URL Policy for ApplicationServer-2`**
Resource Name:	**`http://protectedresource-2.example.com:1081/agentsample/*`**

Note – Make sure the hostname is typed in lowercase.

9 On the resulting page, click Save.

▼ To Create a Test Policy in the Access Manager User Realm

Before You Begin This procedure assumes you have just completed "To Create a Test Referral Policy in the Access Manager Root Realm" on page 196.

1 In the Access Manager console, under the Access Control tab, click the `users` realm link.

2 Click the Policies tab.

3 Under Policies, click New Policy.

4 In the Name field, enter `URL Policy for ApplicationServer-2`.

5 Under Rules, click New.

6 On the resulting page, make sure the default URL Policy Agent (with resource name) is selected and click Next.

7 On the resulting page, provide the following information and click Finish.

Name:	**`agentsample`**
Parent Resource Name:	From the list, select **`http://protectedresource-2.example.com:1081/agentsample/*`**
Resource Name:	The value of this property is populated when you select the Parent Resource Name. It should read **`http://protectedresource-2.example.com:1081/agentsample/*`**.

GET	Mark this check box and verify that Allow is selected.
POST	Mark this check box and verify that Allow is selected.

The rule `agentsample` is now added to the list of Rules.

8 Under Subjects, click New.

9 On the resulting page, select Access Manager Identity Subject and click Next.

10 On the resulting page, provide the following information and click Search.

Name:	**`agentsampleGroup`**
Filter:	Select Group.

`Manager-Group` and `Employee-Group` are displayed in the Available list.

11 Select `Manager-Group` and `Employee-Group` and click Add.

The groups are now displayed in the Selected list.

12 Click Finish.

13 Click OK.

The new policy subject is included in the list of Policies.

14 Log out of the Access Manager console.

▼ To Configure Properties for the J2EE Policy Agent 2 Sample Application

Modify `AMAgent.properties`.

1 Log in as a root user to the ProtectedResource–2 host machine.

2 Change to the `config` directory.

```
# cd /export/J2EEPA2/j2ee_agents/am_wl92_agent/agent_001/config
```

Tip – Backup `AMAgent.properties` before you modify it.

3 Modify these properties in `AMAgent.properties` as follows.

```
com.sun.identity.agents.config.notenforced.uri[0] =
   /agentsample/public/*
com.sun.identity.agents.config.notenforced.uri[1] =
   /agentsample/images/*
com.sun.identity.agents.config.notenforced.uri[2] =
   /agentsample/styles/*
com.sun.identity.agents.config.notenforced.uri[3] =
   /agentsample/index.html
com.sun.identity.agents.config.notenforced.uri[4] =
   /agentsample
com.sun.identity.agents.config.access.denied.uri =
   /agentsample/authentication/accessdenied.html
com.sun.identity.agents.config.login.form[0] =
   /agentsample/authentication/login.html
com.sun.identity.agents.config.login.url[0] =
   http://LoadBalancer-3.example.com:7070/
   amserver/UI/Login?realm=users
com.sun.identity.agents.config.privileged.attribute.
   type[0] = group
com.sun.identity.agents.config.privileged.attribute.
   tolowercase[group] = false
```

4 **Set these remaining properties as follows.**

Note – This is specific to this deployment example. For more information see "The agentadmin -getUuid command fails for amadmin user on Access Manager 7 with various agents (6452713)" in *Sun Java System Access Manager Policy Agent 2.2 Release Notes*.

---

a. **Retrieve the Universal IDs.**

They were saved in "To Create Manager and Employee Groups Using Access Manager for J2EE Policy Agent Test" on page 182.

b. **Convert all uppercase to lowercase and append a back slash (\) in front of each equal sign (=).**

- **Change** `id=Manager-Group,ou=group,o=users,ou=services,dc=example,dc=com` **to** `id\=manager-group,ou\=group,o\=users,ou\=services,dc\=example,dc\=com`**.**

- **Change** `id=Employee-Group,ou=group,o=users,ou=services,dc=example,dc=com` **to** `id\=employee-group,ou\=group,o\=users,ou\=services,dc\=example,dc\=com`**.**

c. **Set the properties.**

```
com.sun.identity.agents.config.privileged.attribute.
   mapping[id\=manager-group,ou\=group,o\=users,ou\=services,
   dc\=example,dc\=com] = am_manager_role
com.sun.identity.agents.config.privileged.attribute.
   mapping[id\=employee-group,ou\=group,o\=users,ou\=services,
   dc\=example,dc\=com] = am_employee_role
```

5 **Save `AMAgent.properties` and close the file.**

6 **Restart the Application Server 2 administration server and managed server.**

a. **Change to the `bin` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/bin
```

b. **Stop the managed server.**

```
# ./stopManagedWebLogic.sh ApplicationsServer-2 t3://localhost:7001
```

c. **Stop the administration server.**

```
# ./stopWebLogic.sh
```

d. **Start the administration server.**

```
# ./startWebLogic.sh &
```

e. **Start the managed server.**

```
# ./startManagedWebLogic.sh ApplicationServer-2 t3://localhost:7001 &
```

7 **Log out of the ProtectedResource–2 host machine.**

▼ To Verify that J2EE Policy Agent 2 is Configured Properly

Use these steps to access the agent sample application and test policies against it.

1 **Access `http://ProtectedResource-2.example.com:1081/agentsample/index.html`, the sample application URL, from a web browser.**

The Sample Application welcome page is displayed.

2 **Click the J2EE Declarative Security link.**

3 **On the resulting page, click Invoke the Protected Servlet.**

You are redirected to the Access Manager login page.

4 **Log in to the Access Manager console as `testuser1`.**

Username	**`testuser1`**
Password	**`password`**

If you can successfully log in as `testuser1` and the J2EE Policy Agent Sample Application page is displayed, the first part of the test has succeeded and authentication is working as expected.

5 **Click the J2EE Declarative Security link again.**

6 **On the resulting page, click Invoke the Protected Servlet.**

If the Success Invocation message is displayed, the second part of the test has succeeded as the sample policy for the `manager` role has been enforced as expected.

7 **Click the J2EE Declarative Security link to return.**

8 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

If the Failed Invocation message is displayed, the third part of the test has succeeded as the sample policy for the `employee` role has been enforced as expected.

9 **Log out and close the browser.**

10 **In a new browser session, access `http://ProtectedResource-2.example.com:1081/agentsample/index.html`, the sample application URL, again.**

The Sample Application welcome page is displayed.

11 **Click the J2EE Declarative Security link.**

12 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

You are redirected to the Access Manager login page.

Tip – If you are not redirected to the Access Manager login page for authentication, clear your browser's cache and cookies and try again.

---

13 **Log in to the Access Manager console as `testuser2`**

Username	**`testuser2`**
Password	**`password`**

The Failed Invocation message is displayed. This is a known issue.

14 **Click the J2EE Declarative Security link.**

15 **On the resulting page, click Invoke the Protected EJB via an Unprotected.**

The Successful Invocation message is displayed as the sample policy for the `employee` role has been enforced as expected.

16 **Click the J2EE Declarative Security link to return.**

17 **On the resulting page, click Invoke the Protected Servlet.**

If the Access to Requested Resource Denied message is displayed, this part of the test has succeeded as the sample policy for the `manager` role has been enforced as expected.

18 **Log out and close the browser.**

9.2.4 Configuring the J2EE Policy Agent 2 to Communicate Over SSL

Use the following list of procedures as a checklist to configure the policy agent to point to the secure port of the Access Manager Load Balancer 3.

1. "To Configure the J2EE Policy Agent 2 for SSL Communication" on page 242
2. "To Import the CA Root Certificate into the Application Server 2 Keystore" on page 243
3. "To Verify that J2EE Policy Agent 2 is Configured Properly" on page 244
4. "To Configure the J2EE Policy Agent 2 to Access the Distributed Authentication User Interface" on page 246

▼ To Configure the J2EE Policy Agent 2 for SSL Communication

1 **Log in as a root user to the ProtectedResource–2 host machine.**

2 **Change to the `config` directory.**

```
# cd /export/J2EEPA2/j2ee_agents/am_wl92_agent/agent_001/config
```

Tip – Backup `AMAgent.properties` before you modify it.

---

3 **Modify these properties in `AMAgent.properties` as follows.**

```
com.sun.identity.agents.config.login.url[0] =
   https://LoadBalancer-3.example.com:9443/amserver/UI/Login?realm=users
com.sun.identity.agents.config.cdsso.cdcservlet.url[0] =
   https://LoadBalancer-3.example.com:9443/amserver/cdcservlet
com.sun.identity.agents.config.cdsso.trusted.id.provider[0] =
   https://LoadBalancer-3.example.com:9443/amserver/cdcservlet
com.iplanet.am.naming.url=
   https://LoadBalancer-3.example.com:9443/amserver/namingservice
com.iplanet.am.server.protocol=https
com.iplanet.am.server.port=9443
```

4 **Save `AMAgent.properties` and close the file.**

▼ To Import the CA Root Certificate into the Application Server 2 Keystore

The Certificate Authority (CA) root certificate enables the J2EE policy agent to trust the certificate from the Access Manager Load Balancer 3, and to establish trust with the certificate chain that is formed from the CA to the certificate. Import the same CA root certificate used in "To Import a Certificate Authority Root Certificate on the Access Manager Load Balancer" on page 108.

Before You Begin This procedure assumes you have just completed "To Configure the J2EE Policy Agent 2 for SSL Communication" on page 242. In this example, the root certificate is a file named `/export/software/ca.cer`.

1. **Change to the directory where the `cacerts` keystore is located.**

```
# cd /usr/local/bea/jdk150_04/jre/lib/security
```

Tip – Backup `cacerts` before you modify it.

2. **Import the root certificate.**

```
# /usr/local/bea/jdk150_04/bin/keytool -import
  -trustcacerts -alias OpenSSLTestCA -file /export/software/ca.cer
  -keystore /usr/local/bea/jdk150_04/jre/lib/security/cacerts
  -storepass changeit

Owner: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
 O=Sun, L=Santa Clara, ST=California, C=US
Issuer: EMAILADDRESS=nobody@nowhere.com, CN=OpenSSLTestCA, OU=Sun,
 O=Sun, L=Santa Clara, ST=California, C=US
Serial number: 97dba0aa26db6386
Valid from: Tue Apr 18 07:55:19 PDT 2006
 until: Tue Jan 13 06:55:19 PST 2009
Certificate fingerprints:
    MD5: 9F:57:ED:B2:F2:88:B6:E8:0F:1E:08:72:CF:70:32:06
    SHA1: 31:26:46:15:C5:12:5D:29:46:2A:60:A1:E5:9E:28:64:36:80:E4:70
Trust this certificate? [no]: yes
Certificate was added to keystore
```

3. **Verify that the certificate was successfully added to the keystore.**

```
# /usr/local/bea/jdk150_04/bin/keytool -list
  -keystore /usr/local/bea/jdk150_04/jre/lib/security/cacerts
  -storepass changeit | grep -i openssl

opensslltestca, Sept 19, 2007, trustedCertEntry,
```

4 **Restart the Application Server 1 administration server and managed instance.**

a. **Change to the `bin` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/bin
```

b. **Stop the managed instance.**

```
# ./stopManagedWebLogic.sh ApplicationsServer-2 t3://localhost:7001
```

c. **Stop the administration server.**

```
# ./stopWebLogic.sh
```

d. **Start the administration server.**

```
# ./startWebLogic.sh &
```

e. **Start the managed instance.**

```
# ./startManagedWebLogic.sh ApplicationServer-2 t3://localhost:7001 &
```

5 **Log out of the ProtectedResource–2 host machine.**

▼ To Verify that J2EE Policy Agent 2 is Configured Properly

Use these steps to access the agent sample application and test policies against it.

1 **Access `http://ProtectedResource-2.example.com:1081/agentsample/index.html`, the sample application URL, from a web browser.**

The Sample Application welcome page is displayed.

2 **Click the J2EE Declarative Security link.**

3 **On the resulting page, click Invoke the Protected Servlet.**

You are redirected to the Access Manager login page.

4 **Log in to the Access Manager console as `testuser1`.**

Username	**`testuser1`**
Password	**`password`**

If you can successfully log in as `testuser1` and the J2EE Policy Agent Sample Application page is displayed, this first part of the test has succeeded and authentication is working as expected.

5 **Click the J2EE Declarative Security link to return.**

6 **On the resulting page, click Invoke the Protected Servlet.**

If the Success Invocation message is displayed, this second part of the test has succeeded as the sample policy for the `manager` role has been enforced as expected.

7 **Click the J2EE Declarative Security link to go back.**

8 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

If the Failed Invocation message is displayed, this third part of the test succeeded as the sample policy for the `employee` role has been enforced as expected.

9 **Log out and close the browser.**

10 **In a new browser session, access `http://ProtectedResource-2.example.com:1081/agentsample/index.html`, the sample application URL.**

The Sample Application welcome page is displayed.

11 **Click the J2EE Declarative Security link.**

12 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

You are redirected to the Access Manager login page.

Tip – If you are not redirected to the Access Manager login page for authentication, clear your browser's cache and cookies and try again.

---

13 **Log in to the Access Manager console as `testuser2`.**

Username	**`testuser2`**
Password	**`password`**

The Failed Invocation message is displayed. This is a known issue.

14 **Click the J2EE Declarative Security link.**

15 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

The Successful Invocation message is displayed. The sample policy for the `employee` role has been enforced as expected.

16 **Click the J2EE Declarative Security link to return.**

17 **On the resulting page, click Invoke the Protected Servlet.**

If the Access to Requested Resource Denied message is displayed, this part of the test is successful as the sample policy for the `manager` role has been enforced as expected.

18 **Log out and close the browser.**

▼ To Configure the J2EE Policy Agent 2 to Access the Distributed Authentication User Interface

Modify `AMAgent.properties`.

1 **Log in as a root user to the ProtectedResource–2 host machine.**

2 **Change to the `config` directory.**

```
# cd /export/J2EEPA2/j2ee_agents/am_wl92_agent/agent_001/config
```

Tip – Backup `AMAgent.properties` before you modify it.

3 **Set the following property.**

```
com.sun.identity.agents.config.login.url[0] =
   https://LoadBalancer-4.example.com:9443/distAuth/UI/Login?realm=users
```

4 **Save `AMAgent.properties` and close the file.**

5 **Restart the Application Server 1 managed server.**

a. **Change to the `bin` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/bin
```

b. **Stop the managed server.**

```
# ./stopManagedWebLogic.sh ApplicationsServer-2 t3://localhost:7001
```

c. **Start the managed server.**

```
# ./startManagedWebLogic.sh ApplicationServer-2 t3://localhost:7001
```

6 **Log out of the ProtectedResource–2 host machine.**

7 Verify that the agent is configured properly.

a. **Access** `http://ProtectedResource-2.example.com:1081/agentsample/index.html`**, the sample application URL, form a web browser.**

The Sample Application Welcome page is displayed.

b. **Click the J2EE Declarative Security link.**

c. **On the resulting page, click Invoke the Protected Servlet.**

You are redirected to the Distributed Authentication User Interface at `https://loadbalancer-4.example.com:9443/distAuth/UI/Login`.

d. **(Optional) Double-click the gold lock in the lower left corner of the browser.**

In the Properties page, you see the certificate for `LoadBalancer–4.example.com`.

e. **Log in to the Access Manager console as** `testuser1`**.**

Username	**`testuser1`**
Password	**`password`**

If you can successfully log in as `testuser1` and the J2EE Policy Agent Sample Application page is displayed, user authentication worked through the Distributed Authentication User Interface.

f. **Log out of the console.**

CHAPTER 10

Setting Up Load Balancers for the Policy Agents

Two load balancers are configured for the policy agents in this deployment example. Load Balancer 5 balances traffic passing through the web policy agents. Load Balancer 6 balances traffic passing through the J2EE policy agents. Both load balancers are configured for *simple persistence* so that browser requests from the same IP address will always be directed to the same policy agent. This chapter contains detailed procedures for the following tasks:

- "10.1 Configuring the Web Policy Agents Load Balancer" on page 249
- "10.2 Configuring the J2EE Policy Agents Load Balancer" on page 258

10.1 Configuring the Web Policy Agents Load Balancer

Load Balancer 5 handles traffic for the web policy agents, and is configured for simple persistence so that browser requests from the same IP address will always be directed to the same policy agent. From a performance perspective, each policy agent validates the user session and evaluates applicable policies. The results are subsequently cached by the individual policy agent to improve performance. If load balancer persistence is not set, each agent must build up its own cache, effectively doubling the workload on the Access Manager servers, and cutting overall system capacity in half. The problem becomes even more acute as the number of policy agents increases. Simple persistence guarantees that the requests from the same user session will always be sent to the same policy agent.

Tip – In situations where each web policy agent instance is protecting identical resources, some form of load balancer persistence is highly recommended for the performance reasons. The actual type of persistence may vary when a different load balancer is used, as long as it achieves the goal of sending the requests from the same user session to the same policy agent.

The following illustration shows the architecture of the policy agents and load balancers.

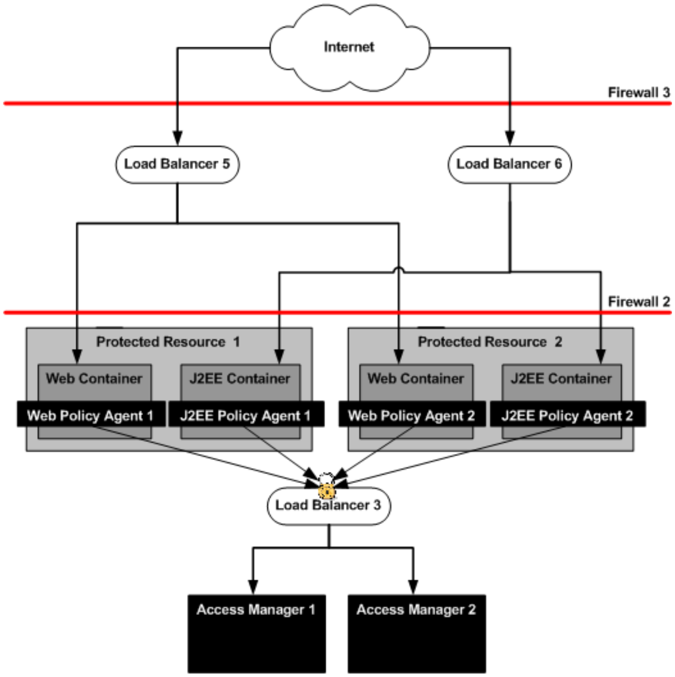


FIGURE 10–1 Policy Agents and Load Balancers

Note – When firewalls are configured, Load Balancer 5 can be located in a less secure zone.

Use the following list of procedures as a checklist for configuring the web policy agents' load balancer:

1. "To Configure the Web Policy Agents Load Balancer" on page 250
2. "To Point the Web Policy Agents to Load Balancer 5" on page 253
3. "To Configure Policy for the Web Policy Agents Using Access Manager" on page 255
4. "To Verify the Web Policy Agents Load Balancer Configuration is Working Properly" on page 257

▼ To Configure the Web Policy Agents Load Balancer

Before You Begin The load balancer hardware and software used for this deployment is BIG-IP® manufactured by F5 Networks. If you are using different load balancer software, see the documentation that comes with that product for detailed settings information.

1 Access `https://is-f5.example.com`**, the Big IP load balancer login page, from a web browser.**

2 **Log in using the following credentials:**

User name:	**`username`**
Password:	**`password`**

3 **Click** *Configure your BIG-IP (R) using the Configuration Utility***.**

4 **Create a Pool.**

A pool contains all the backend server instances.

a. **In the left pane, click Pools.**

b. **On the Pools tab, click Add.**

c. **In the Add Pool dialog, provide the following information:**

Pool Name	**`WebAgent-Pool`**
Load Balancing Method	Round Robin
Resources	Add the IP address and port number of both Protected Resource host machines: `ProtectedResource-1:1080` and `ProtectedResource-2:1080`.

d. **Click Done.**

5 **Add a Virtual Server.**

This step defines instances of the load balancer.

Tip – If you encounter JavaScript™ errors or otherwise cannot proceed to create a virtual server, try using Internet Explorer.

a. **In the left frame, click Virtual Servers.**

b. **On the Virtual Servers tab, click Add.**

c. **In the Add a Virtual Server dialog box, provide the following information:**

Address	Enter the IP address for `LoadBalancer-5.example.com`
Service	**`90`**
Pool	**`WebAgent-Pool`**

d. **Continue to click Next until you reach the Pool Selection dialog box.**

- **e. In the Pool Selection dialog box, assign the** `WebAgent-Pool` **Pool.**
- **f. Click Done.**

6 Add Monitors.

Monitors are required for the load balancer to detect the backend server failures.

- **a. In the left frame, click Monitors.**
- **b. Click Add.**

  In the Add Monitor dialog provide the following information:

Name:	**`WebAgent-http`**
Inherits From:	Choose **`http`**.

- **c. Click Next.**
- **d. On the resulting Configure Basic Properties page, click Next.**
- **e. In the Send String field under Configure ECV HTTP Monitor, enter `GET /monitor.html` and click Next.**
- **f. On the Destination Address and Service (Alias) page, click Done.**

  The monitor just added is in the list of monitors under the Monitors tab.

- **g. Click the Basic Associations tab.**
- **h. Mark the Add checkbox next to the IP addresses for** `ProtectedResource-1` **and** `ProtectedResource-2`**.**
- **i. At the top of the Node column, choose the monitor that you just added,** `WebAgent-http`**.**
- **j. Click Apply.**

7 Configure the load balancer for simple persistence.

The web policy agents load balancer is configured with *simple persistence*. With simple persistence, all requests sent *within a specified interval* from the same user are routed to the same agent. This significantly reduces the number of agent requests to sent to Access Manager for validation thus reducing the overall load on the Access Manager servers.

Note – Simple persistence tracks connections based on the client IP address only, returning a client to the same node to which it connected previously.

a. **In the left frame, click Pools.**

b. **Click the** `WebAgent-Pool` **link.**

c. **Click the Persistence tab.**

d. **Under Persistence Type, select the Simple.**

e. **Set the timeout interval.**

   In the Timeout field, enter 300 seconds.

f. **Click Apply.**

8 **Log out of the console.**

▼ To Point the Web Policy Agents to Load Balancer 5

Modify `AMAgent.properties` to point Protected Resource 1 and Protected Resource 2 to Load Balancer 5.

1 **As a root user, log in to the ProtectedResource–1 host machine.**

2 **Change to the** `config` **directory.**

```
# cd /export/WebPA1/web_agents/sjsws_agent/Agent_001/config
```

Tip – Backup `AMAgent.properties` before you modify it.

---

3 **Make the following changes to** `AMAgent.properties`**.**

a. **Add the following entry:**

```
com.sun.am.policy.agents.config.fqdn.map =
 valid|LoadBalancer-5.example.com
```

b. **Append the following to the end of the value string for the** `com.sun.am.policy.agents.config.notenforced_list` **property:**

```
http://ProtectedResource-1.example.com:1080/monitor.html
http://LoadBalancer-5.example.com:90/monitor.html
```

4 **Save the file and close it.**

5 **Create a `monitor.html` file to be used by the load balancer.**

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-1.example.com/docs
# cat > monitor.html
<HTML>
</HTML>
```

Hit Control D to terminate the command

```
^D
```

6 **Restart Web Server 1 on the Protected Resource 1 host machine.**

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-1.example.com/bin
# ./stopserv; ./startserv
```

7 **Log out of the ProtectedResource–1 host machine.**

8 **As a root user, log in to the ProtectedResource–2 host machine.**

9 **Change to the `config` directory.**

```
# cd /export/WebPA2/web_agents/sjsws_agent/Agent_001/config
```

10 **Make the following changes to the `AMAgent.properties` file.**

Tip – Backup `AMAgent.properties` before you modify it.

a. **Add the following entry:**

```
com.sun.am.policy.agents.config.fqdn.map =
 valid|LoadBalancer-5.example.com
```

b. **Append the following to the end of the value string for the `com.sun.am.policy.agents.config.notenforced_list` property:**

```
http://ProtectedResource-2.example.com:1080/monitor.html
http://LoadBalancer-5.example.com:90/monitor.html
```

11 **Save the file and close it.**

12 **Create a `monitor.html` file to be used by the load balancer.**

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-2.example.com/docs
# cat > monitor.html
<HTML>
</HTML>
```

Hit Control D to terminate the command

```
^D
```

13 **Restart Web Server 2 on the Protected Resource 2 host machine.**

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-2.example.com/bin
# ./stopserv; ./startserv
```

14 **Log out of the ProtectedResource–2 host machine.**

▼ To Configure Policy for the Web Policy Agents Using Access Manager

Use the Access Manager console to configure policy for the Web Policy Agents.

1 **Access the Access Manager server,** `http://AccessManager-1.example.com:1080/amserver/UI/Login`**, from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

Username	**`amadmin`**
Password	**`4m4dmin1`**

3 **Modify the referral policy for access to Load Balancer 5.**

a. **On the Access Control tab, click the top-level realm** `example`**.**

b. **Click the Policies tab.**

c. **Click the** `Referral URL Policy for users realm` **link.**

d. **On the Edit Policy page, under Rules, click New.**

e. **On the resulting page, select** `URL Policy Agent (with resource name)` **and click Next.**

This selection is used to define policies that protect HTTP and HTTPS URLs.

f. **On the resulting page, provide the following information:**

Name:	**`URL Rule for LoadBalancer-5`**
Resource Name:	**`http://LoadBalancer-5.example.com:90/*`**

g. **Click Finish.**

h. **On the resulting page, click Save.**

   The new rule is in the Rules list.

4 **Create a policy in the `users` sub-realm.**

a. **On the Access Control tab, click the `users` link.**

b. **Click the Policies tab, and then New Policy.**

c. **In the Name field, enter `URL Policy for LoadBalancer-5`.**

d. **Under Rules, click New.**

e. **On the resulting page, accept the default `URL Policy Agent (with resource name)` and click Next.**

f. **On the resulting page, provide the following information:**

Name:	**`LoadBalancer-5`**.
Parent Resource Name:	In the list, select **`http://LoadBalancer-5.example.com:90/*`**.
Resource Name:	**`http://LoadBalancer-5.example.com:90/*`** is automatically entered when you select the Parent Resource Name.
GET	Mark this checkbox and select Allow.
POST	Mark this checkbox and select Allow.

g. **Click Finish.**

h. **On the New Policy page again, under Subjects, click New.**

i. **On the resulting page, verify that Access Manager Identity Subject is selected, and click Next.**

j. **On the resulting page, provide the following information:**

Name:	**`LoadBalancer-5_Groups`**
Filter:	In the drop-down list, select Group and click Search.

   The search returns a list of available groups.

k. **Select `Employee-Group` and `Manager-Group` and click Add.**

   The `Employee-Group` and `Manager-Group` groups are in the Selected List.

l. Click Finish.

m. On the resulting page, click OK.

The policy you just created is now included in the list of Policies.

5 Log out of the Access Manager console and close the browser.

▼ To Verify the Web Policy Agents Load Balancer Configuration is Working Properly

1 Access `http://loadbalancer-5.example.com:90/index.html`, the Access Manager load balancer, from a web browser.

2 Log in to the Access Manager console as `testuser1`.

Username	**`testuser1`**
Password	**`password`**

If the default Web Server `index.html` page is displayed, the load balancer is configured properly.

3 Verify that Load Balancer 5 monitors are monitoring the Web Server instances properly.

a. Log in as a root user to the ProtectedResource–1 host machine.

b. Run the `tail` command.

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-1.example.com/logs
# tail -f access
```

If you see frequent entries similar to the one below, the custom monitor is configured properly.

```
IP_address - - [21/Sep/2007:13:59:48 -0700]
"GET /monitor.html" 200 15
```

If you do not see "`GET /monitor.html`", you must troubleshoot the load balancer configuration.

c. Log in as a root user to the ProtectedResource–2 host machine.

d. Run the `tail` command.

```
# cd /opt/SUNWwbsvr/https-ProtectedResource-2.example.com/logs
# tail -f access
```

If you see frequent entries similar to the one below, the custom monitor is configured properly.

```
IP_address - - [21/Sep/2007:13:59:48 -0700]
"GET /monitor.html" 200 15
```

If you do not see "`GET /monitor.html`", you must troubleshoot the load balancer configuration.

e. **Log out of both Protected Resource host machines after you have verified that the monitors are working properly.**

10.2 Configuring the J2EE Policy Agents Load Balancer

Load Balancer 6 handles traffic for the J2EE policy agents, and is configured for simple persistence so that browser requests from the same IP address will always be directed to the same policy agent. From a performance perspective, each policy agent validates the user session and evaluates applicable policies. The results are subsequently cached by the individual policy agent to improve performance. If load balancer persistence is not set, each agent must build up its own cache, effectively doubling the workload on the Access Manager servers, and cutting overall system capacity in half. The problem becomes even more acute as the number of policy agents increases. Simple persistence guarantees that the requests from the same user session will always be sent to the same policy agent.

Tip – In situations where each J2EE policy agent instance is protecting identical resources, some form of load balancer persistence is highly recommended for the performance reasons. The actual type of persistence may vary when a different load balancer is used, as long as it achieves the goal of sending the requests from the same user session to the same policy agent.

The following illustration shows the architecture of the policy agents and load balancers.

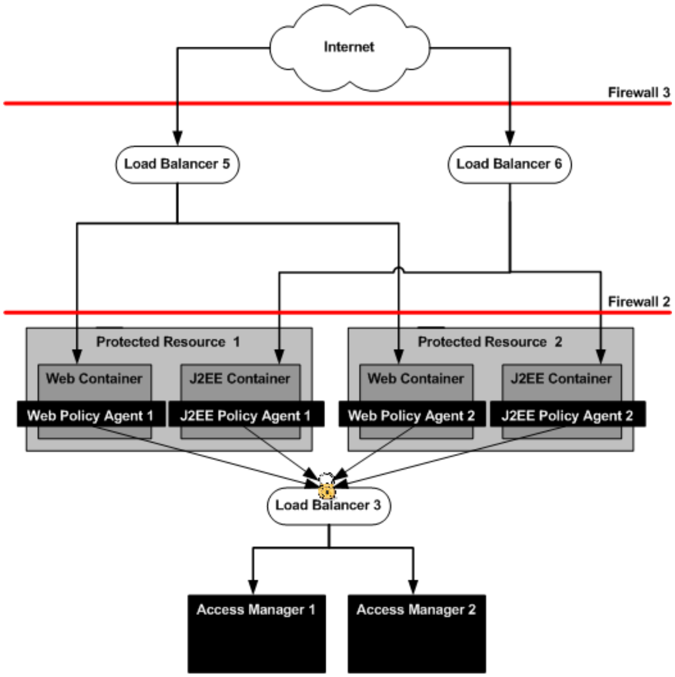


FIGURE 10–2 Policy Agents and Load Balancers

Note – When firewalls are configured, Load Balancer 6 can be located in a less secure zone.

Use the following list of procedures as a checklist for configuring the J2EE policy agents' load balancer:

1. "To Configure the J2EE Policy Agents Load Balancer" on page 259
2. "To Point the J2EE Policy Agents to Load Balancer 6" on page 261
3. "To Create Polices for the Agent Resources" on page 262
4. "To Verify the J2EE Policy Agent Load Balancer Configuration is Working Properly" on page 264

▼ To Configure the J2EE Policy Agents Load Balancer

Before You Begin The load balancer hardware and software used for this deployment is BIG-IP® manufactured by F5 Networks. If you are using different load balancer software, see the documentation that comes with that product for detailed settings information.

1 Access `https://is-f5.example.com`**, the Big IP load balancer login page, from a web browser.**

2 Log in using the following information:

User name:	**`username`**
Password:	**`password`**

3 Click *Configure your BIG-IP (R) using the Configuration Utility*.

4 Create a Pool.

A pool contains all the backend server instances.

a. In the left pane, click Pools.

b. On the Pools tab, click Add.

c. In the Add Pool dialog, provide the following information:

Pool Name	`J2EEAgent-Pool`
Load Balancing Method	Round Robin
Resources	Add the Application Server IP addresses and port numbers: `ProtectedResource-1:1081` and `ProtectedResource-2:1081`.

d. Click Done.

e. In the List of Pools, click `J2EEAgent-Pool`**.**

f. Click the Persistence tab and provide the following information:

Persistence Type: Choose `Active Http Cookie`

Note – `Active Http Cookie` persistence uses an HTTP cookie stored on a client computer to allow the client to reconnect to the same server previously visited.

Method: Choose `Insert`

g. Click Apply.

5 **Add a Virtual Server.**

If you encounter JavaScript errors or otherwise cannot proceed to create a virtual server, try using Internet Explorer for this step.

a. **In the left frame, click Virtual Servers.**

b. **On the Virtual Servers tab, click Add.**

c. **In the Add a Virtual Server dialog box, provide the following information:**

Address	Enter the IP address for `LoadBalancer-6.example.com`
Services Port	**`91`**
Pool	**`J2EEAgent-Pool`**

d. **Continue to click Next until you reach the Pool Selection dialog box.**

e. **In the Pool Selection dialog box, assign the `J2EEAgent-Pool` pool.**

f. **Click Done.**

6 **Add Monitors.**

a. **Click the Basic Associations tab.**

b. **Mark the Add checkbox for the IP address for ProtectedResource-1 and ProtectedResource-2.**

c. **At the top of the Node column, select `tcp`.**

d. **Click Apply.**

7 **Log out of the load balancer console.**

▼ To Point the J2EE Policy Agents to Load Balancer 6

Modify the `AMAgent.properties` file to point Protected Resource 1 and Protected Resource 2 to Load Balancer 6.

1 **As a root user, log in to the ProtectedResource–1 host machine.**

2 **Change to the `config` directory.**

```
# cd /export/J2EEPA1/j2ee_agents/am_wl92_agent/agent_001/config
```

Tip – Backup `AMAgent.properties` before you modify it.

3 **Make the following change to the `AMAgent.properties` file.**

```
com.sun.identity.agents.config.fqdn.mapping[LoadBalancer-6.example.com] =
LoadBalancer-6.example.com
```

4 **Save the file and close it.**

5 **Log out of the ProtectedResource–1 host machine.**

6 **As a root user, log in to the ProtectedResource–2 host machine.**

7 **Change to the `config` directory.**

```
# cd /export/J2EEPA2/j2ee_agents/am_wl92_agent/agent_001/config
```

Tip – Backup `AMAgent.properties` before you modify it.

8 **Make the following change to the `AMAgent.properties` file.**

```
com.sun.identity.agents.config.fqdn.mapping[LoadBalancer-6.example.com] =
LoadBalancer-6.example.com
```

9 **Save the file and close it.**

10 **Log out of the ProtectedResource–2 host machine.**

▼ To Create Polices for the Agent Resources

The policies you create here are used in “To Verify the J2EE Policy Agent Load Balancer Configuration is Working Properly” on page 264.

1 **Access the Access Manager server,** `http://AccessManager-1.example.com:1080/amserver/UI/Login`**, from a web browser.**

2 **Log in to the Access Manager console as the administrator.**

Username	**`amadmin`**
Password	**`4m4dmin1`**

3 Modify the referral policy for access to Load Balancer 6.

a. On the Access Control tab, click the top-level realm `example`.

b. Click the Policies tab.

c. Click the `Referral URL Policy for users realm` link.

d. On the Edit Policy page, under Rules, click New.

e. On the resulting page, select `URL Policy Agent (with resource name)` and click Next.

This selection is used to define policies that protect HTTP and HTTPS URLs.

f. On the resulting page, provide the following information:

Name:	**`URL Rule for LoadBalancer-6`**
Resource Name:	**`http://loadbalancer-6.example.com:91/*`**

Note – Make sure all letters are lowercase.

g. Click Finish.

h. On the resulting page, click Save.

The new rule is in the Rules list.

4 Create a policy in the `users` sub-realm.

a. On the Access Control tab, click the `users` link.

b. Click the Policies tab, and then New Policy.

c. In the Name field, enter `URL Policy for LoadBalancer-6`.

d. Under Rules, click New.

e. On the resulting page, accept the default `URL Policy Agent (with resource name)` and click Next.

f. On the resulting page, provide the following information:

Name:	**`LoadBalancer-6`**.
Parent Resource Name:	From the list, select, `http://loadbalancer-6.example.com:91/*`.

Resource Name:	**`http://loadbalancer-6.example.com:91/*`** is automatically entered when you select the Parent Resource Name.
GET	Mark the checkbox and select Allow.
POST	Mark the checkbox and select Allow.

g. Click Finish.

h. On the New Policy page again, under Subjects, click New.

i. On the resulting page, verify that Access Manager Identity Subject is selected, and click Next.

j. On the resulting page, provide the following information:

Name:	**`LoadBalancer-6_Groups`**
Filter:	In the drop-down list, select Group and click Search.

The search returns a list of available groups.

k. Select `Employee-Group` and `Manager-Group` and click Add.

The `Employee-Group` and `Manager-Group` groups are in the Selected List.

l. Click Finish.

m. On the resulting page, click OK.

The policy you just created is now included in the list of Policies.

5 Log out of the Access Manager console and close the browser.

▼ To Verify the J2EE Policy Agent Load Balancer Configuration is Working Properly

1 Restart the Application Servers.

a. As a root user, log in to the ProtectedResource–1 host machine.

b. Change to the `bin` directory.

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-1/bin
```

c. **Stop Application Server 1 managed instance.**

```
# ./stopManagedWebLogic.sh ApplicationsServer-1 t3://localhost:7001
```

d. **Stop the Application Server 1 administration server.**

```
# ./stopWebLogic.sh
```

e. **Start the Application Server 1 administration server.**

```
# ./startWebLogic.sh &
```

f. **Start Application Server 1 managed instance.**

```
# ./startManagedWebLogic.sh ApplicationServer-1 t3://localhost:7001
```

g. **Log out of the ProtectedResource–1 host machine.**

h. **As a root user, log in to the ProtectedResource–2 host machine.**

i. **Change to the `bin` directory.**

```
# cd /usr/local/bea/user_projects/domains/ProtectedResource-2/bin
```

j. **Stop the Application Server 2 managed instance.**

```
# ./stopManagedWebLogic.sh ApplicationsServer-2 t3://localhost:7001
```

k. **Stop the Application Server 2 administration server.**

```
# ./stopWebLogic.sh
```

l. **Start the Application Server 2 administration server.**

```
# ./startWebLogic.sh &
```

m. **Start the Application Server 2 managed instance.**

```
# ./startManagedWebLogic.sh ApplicationServer-2 t3://localhost:7001
```

n. **Log out of the ProtectedResource–2 host machine.**

2 **Access `http://LoadBalancer-6.example.com:91/agentsample/index.html` from a web browser.**

The Sample Application welcome page is displayed.

3 **Click the J2EE Declarative Security link.**

4 **On the resulting page click Invoke the Protected Servlet.**

The policy agent redirects to the Access Manager login page.

5 **Log in to the Access Manager console as `testuser1`.**

Username	**`testuser1`**
Password	**`password`**

If you can successfully log in as `testuser1` and the J2EE Policy Agent Sample Application page is displayed, this first part of the test succeeded and authentication is working as expected.

6 **Click the J2EE Declarative Security link to return.**

7 **On the resulting page, click Invoke the Protected Servlet.**

If the Successful Invocation message is displayed, this second part of the test has succeeded and the sample policy for the employee role has been enforced as expected.

8 **Close the browser.**

9 **Open a new browser and access `http://LoadBalancer-6.example.com:91/agentsample/index.html`.**

The Sample Application welcome page is displayed.

10 **Click the J2EE Declarative Security link.**

11 **On the resulting page click Invoke the Protected Servlet.**

The policy agent redirects to the Access Manager login page.

12 **Log in to the Access Manager console as `testuser2`.**

Username	**`testuser2`**
Password	**`password`**

If the Access to Requested Resource Denied message is displayed, this third part of the test succeeded and the sample policy for the manager role has been enforced as expected.

13 **Click the J2EE Declarative Security link to return.**

14 **On the resulting page, click Invoke the Protected EJB via an Unprotected Servlet.**

If the Successful Invocation message is displayed, the sample policy for the employee role has been enforced as expected.

15 **Close the browser.**

◆ ◆ ◆ CHAPTER 11

Implementing Session Failover

Sun Java[TM] System Access Manager provides a web container-independent session failover feature that uses Sun Java System Message Queue. Message Queue is a messaging middleware product that enables distributed applications to communicate and exchange data by sending and receiving messages. Access Manager uses Message Queue as a communications broker, and uses the Berkeley DB by Sleepycat Software, Inc. for the backend session store databases. This chapter contains the following sections:

- "11.1 Session Failover Architecture" on page 267
- "11.2 Installing the Access Manager Session Failover Components" on page 268
- "11.3 Configuring and Verifying Session Failover" on page 276

11.1 Session Failover Architecture

When session failover is implemented for Access Manager, session information is replicated in two backend session store databases. This ensures that if one Access Manager fails or stops, the information stored in the backend session databases can be used to keep the user continuously authenticated. If session failover is not implemented and the Access Manager server in which a user's session was created fails, the user will have to reauthenticate to perform an operation that requires a session token. The following diagram illustrates the session failover architecture.

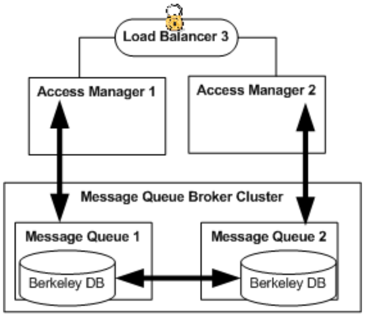


FIGURE 11–1 Session Failover

Note – For more information about Access Manager and session failover, see Chapter 6, "Implementing Session Failover," in *Sun Java System Access Manager 7.1 Postinstallation Guide*.

11.2 Installing the Access Manager Session Failover Components

Use the following list of procedures as a checklist for installing the Access Manager session failover components.

1. "To Install Access Manager Session Failover Components on Message Queue 1" on page 268
2. "To Install Access Manager Session Failover Components on Message Queue 2" on page 272

▼ To Install Access Manager Session Failover Components on Message Queue 1

1 **As a root user, log in to the MessageQueue–1 host machine.**

2 **Create a directory into which the Message Queue and Berkeley Database bits can be downloaded and change into it.**

```
# mkdir /export/AMSFO
# cd /export/AMSFO
```

3 **Copy** `amSessionTools.zip` **from the AccessManager–1 host machine to the MessageQueue–1 host machine.**

Note – `amSessionTools.zip` is included in the `AccessManager7_1RTM.zip` file downloaded in "To Generate an Access Manager WAR File on the Access Manager 1 Host Machine" on page 88. `amSessionTools.zip` can be found under the `tools` directory.

---

4 **Unzip `amSessionTools.zip`.**

```
# cd /export/AMSFO
# unzip amSessionTools.zip -d amSessionTools
```

5 **Modify the permissions on the `setup` script and run it to initialize the session failover tools.**

```
# cd /export/AMSFO/amSessionTools
# chmod +x setup
# ./setup
```

6 **When prompted, enter `amserver` as the *Directory to install the scripts (example: amserver)*.**

Note – The directory location should be relative to the current directory.

When the script is finished, the following messages are displayed:

```
The scripts are properly setup under directory
   /export/AMSFO/amSessionTools/amserver
JMQ is properly setup under directory jmq.
BerkeleyDB is properly setup under directory bdb.
```

7 **Change to the `bin` directory.**

```
# cd /export/AMSFO/amSessionTools/jmq/imq/bin
```

8 **Run the `imqbrokerd` command to create a new broker instance named `msgqbroker`.**

```
# ./imqbrokerd -name msgqbroker -port 7777 &
```

9 **Run `netstat` to verify that the new Message Queue broker instance is up and running.**

```
# netstat -an | grep 7777

*.7777        *.*            0            0    49152        0    LISTEN
```

10 **Add a new user named `msgquser`.**

This user will connect to the Message Queue broker instance on servers where Message Queue is installed. This user will be used only for session failover purposes, and does not assume the privileges of the guest user. It is a good practice to create a custom user for such purposes, and not to rely on the known user accounts or default user accounts to help prevent brute force or DOS attacks.

```
# ./imqusermgr add -u msgquser -g admin -p m5gqu5er -i msgqbroker
```

11 Disable the guest user.

This step ensures that the guest user will not be able to access the Access Manager server.

```
# ./imqusermgr update -u guest -a false -i msgqbroker

User repository for broker instance: msgqbroker

Are you sure you want to update user guest? (y/n) y

User guest successfully updated.
```

12 Modify the amsfo.conf file.

amsfo.conf has parameters that are consumed by the Access Manager session failover startup script, amsfo.

a. Change to the lib directory.

```
# cd /export/AMSFO/amSessionTools/amserver/config/lib
```

Tip – Backup amsfo.conf before you modify it.

b. Set the following properties:

```
CLUSTER_LIST=MessageQueue-1.example.com:7777,MessageQueue-2.example.com:7777
BROKER_INSTANCE_NAME=msgqbroker
USER_NAME=msgquser
BROKER_PORT=7777
```

Note – The port used for BROKER_PORT should be the same as the one used in the value of the CLUSTER_LIST.

c. Save the file and close it.

13 Run the amsfopassword command.

This command generates an encrypted password, creates a new file named .password, and stores the encrypted password in the new file.

Caution – amsfopassword creates the .password file in a default location based on where the scripts were installed. If a different location is used, the PASSWORDFILE property in amsfo.conf should be changed accordingly.

a. Change to the bin directory.

```
# cd /export/AMSFO/amSessionTools/amserver/bin
```

b. **Run `amsfopassword`.**

```
# cd /export/AMSFO/amSessionTools/amserver/bin
# ./amsfopassword -e m5gqu5er -f /export/AMSFO/amSessionTools/amserver/.password

os.name=SunOS
SUCCESSFUL
```

c. **(Optional) View the encrypted password.**

```
# more /export/AMSFO/amSessionTools/amserver/.password

M27OGb6U4ufRu+oWAzBdWw==
```

14 **(Optional) Modify the `amsessiondb` script if necessary.**

The `amsessiondb` script (located in the `/export/AMSFO/amSessionTools/amserver/bin` directory) starts the Berkeley Database client, creates the database, and sets specific database values. It is called when the `amsfo` script is run for the first time. The `amsessiondb` script contains variables that specify default paths and directories. If any of the following components are not installed in their default directories, edit the `amsessiondb` script to set the variables to the correct locations.

```
JAVA_HOME=/usr/jdk/entsys-j2se
IMQ_JAR_PATH=/export/AMSFO/amSessionTools/jmq/imq/lib
JMS_JAR_PATH=/export/AMSFO/amSessionTools/jmq/imq/lib
BDB_JAR_PATH=/export/AMSFO/amSessionTools/bdb/usr/lib
BDB_SO_PATH=/export/AMSFO/amSessionTools/bdb/usr/lib
AM_HOME=/export/AMSFO/amSessionTools
```

Tip – Backup `amsessiondb` before you modify it.

15 **Restart the Access Manager session failover components.**

a. **Change to the `bin` directory.**

```
# cd /export/AMSFO/amSessionTools/jmq/imq/bin
```

b. **Stop the Message Queue instance using the product's command line interface.**

See the Message Queue documentation for more information.

c. **Run the `netstat` command to verify that the MessageQueue-1 broker instance is stopped.**

```
# netstat -an | grep 7777
```

If `netstat` returns no result, the MessageQueue-1 broker instance is stopped.

Tip – If the MessageQueue-1 broker instance is not stopped, kill the process using the following procedure.

a. Get the Java process IDs.

```
# ps -ef | grep java
```

b. Kill the Java process IDs that were returned.

```
# kill -9 #### ####
```

c. Run `netstat` again.

---

d. **Restart the MessageQueue-1 broker instance.**

```
# ./amfso start
```

e. **Run the `netstat` command to verify that the Message Queue port is open and listening.**

```
# netstat -an | grep 7777

*.7777            *.*            0            0    49152            0    LISTEN
```

16 **Log out of the MessageQueue–1 host machine.**

▼ To Install Access Manager Session Failover Components on Message Queue 2

1 **As a root user, log in to the MessageQueue–2 host machine.**

2 **Create a directory into which the Message Queue and Berkeley Database bits can be downloaded and change into it.**

```
# mkdir /export/AMSFO
# cd /export/AMSFO
```

3 **Copy `amSessionTools.zip` from the AccessManager–1 host machine to the MessageQueue–2 host machine.**

Note – `amSessionTools.zip` is included in the `AccessManager7_1RTM.zip` file downloaded in "To Generate an Access Manager WAR File on the Access Manager 1 Host Machine" on page 88. `amSessionTools.zip` can be found under the `tools` directory.

---

4 **Unzip `amSessionTools.zip`.**

```
# cd /export/AMSFO
# unzip amSessionTools.zip -d amSessionTools
```

5 **Modify the permissions on the `setup` script and run it to initialize the session failover tools.**

```
# cd /export/AMSFO/amSessionTools
# chmod +x setup
# ./setup
```

6 **When prompted, enter `amserver` as the** ***Directory to install the scripts (example: amserver)*****.**

Note – The directory location should be relative to the current directory.

When complete, the following messages are displayed:

```
The scripts are properly setup under directory
   /export/AMSFO/amSessionTools/amserver
JMQ is properly setup under directory jmq.
BerkeleyDB is properly setup under directory bdb.
```

7 **Change to the `bin` directory.**

```
# cd /export/AMSFO/amSessionTools/jmq/imq/bin
```

8 **Run the `imqbrokerd` command to create a new broker instance named `msgqbroker`.**

```
# ./imqbrokerd -name msgqbroker -port 7777 &
```

9 **Run `netstat` to verify that the new Message Queue instance is up and running.**

```
# netstat -an | grep 7777

*.7777        *.*            0            0    49152        0    LISTEN
```

10 **Add a new user named `msgquser`.**

This is the user that will be used to connect to the Message Queue broker on servers where Message Queue is installed. This user will be used only for session failover purposes, and does not assume the privileges of the guest user. It is a good practice to create a custom user for such purposes, and not to rely on the known user accounts or default user accounts. This helps to prevent brute force or DOS attacks.

```
# ./imqusermgr add -u msgquser -g admin -p m5gqu5er -i msgqbroker
```

11 **Disable the guest user.**

This step ensures that the guest user will not be able to access the Access Manager server.

```
# ./imqusermgr update -u guest -a false -i msgqbroker
```

```
User repository for broker instance: msgqbroker

Are you sure you want to update user guest? (y/n) y

User guest successfully updated.
```

12 **Stop the Message Queue instance using the product's command line interface.**

13 **Modify the `amsfo.conf` file.**

`amsfo.conf` has parameters that are consumed by the Access Manager session failover startup script, `amsfo`.

a. **Change to the `lib` directory.**

```
# cd /export/AMSFO/amSessionTools/amserver/config/lib
```

Tip – Backup `amsfo.conf` before you modify it.

b. **Set the following properties:**

```
CLUSTER_LIST=MessageQueue-1.example.com:7777,MessageQueue-2.example.com:7777
BROKER_INSTANCE_NAME=msgqbroker
USER_NAME=msgquser
BROKER_PORT=7777
```

Note – The port used for `BROKER_PORT` should be the same as the one used in the value of the `CLUSTER_LIST`.

c. **Save the file and close it.**

14 **Run the `amsfopassword` command.**

This command generates an encrypted password, creates a new file named `.password`, and stores the encrypted password in the new file.

Caution – `amsfopassword` creates the `.password` file in a default location based on where the scripts were installed. If a different location is used, the `PASSWORDFILE` property in `amsfo.conf` should be changed accordingly.

a. **Change to the `bin` directory.**

```
# cd /export/AMSFO/amSessionTools/amserver/bin
```

b. **Run `amsfopassword`.**

```
# cd /export/AMSFO/amSessionTools/amserver/bin
# ./amsfopassword -e m5gqu5er -f /export/AMSFO/amSessionTools/amserver/.password
```

```
os.name=SunOS
SUCCESSFUL
```

c. **(Optional) View the encrypted password.**

```
# more /export/AMSFO/amSessionTools/amserver/.password

M27OGb6U4ufRu+oWAzBdWw==
```

15 **(Optional) Modify the `amsessiondb` script if necessary.**

The `amsessiondb` script (located in the `/export/AMSFO/amSessionTools/amserver/bin` directory) starts the Berkeley Database client, creates the database, and sets specific database values. It is called when the `amsfo` script is run for the first time. The `amsessiondb` script contains variables that specify default paths and directories. If any of the following components are not installed in their default directories, edit the `amsessiondb` script to set the variables to the correct locations.

```
JAVA_HOME=/usr/jdk/entsys-j2se
IMQ_JAR_PATH=/export/AMSFO/amSessionTools/jmq/imq/lib
JMS_JAR_PATH=/export/AMSFO/amSessionTools/jmq/imq/lib
BDB_JAR_PATH=/export/AMSFO/amSessionTools/bdb/usr/lib
BDB_SO_PATH=/export/AMSFO/amSessionTools/bdb/usr/lib
AM_HOME=/export/AMSFO/amSessionTools
```

Tip – Backup `amsessiondb` before you modify it.

16 **Restart the Access Manager session failover components.**

a. **Change to the `bin` directory.**

```
# cd /export/AMSFO/amSessionTools/amserver/bin
```

b. **Stop the MessageQueue-2 broker instance.**

```
# ./amsfo stop
```

The port socket should be relinquished before you restart. If not, session failover problems may occur.

c. **Run the `netstat` command to verify that the MessageQueue-2 broker instance is stopped.**

```
# netstat -an | grep 7777
```

If `netstat` returns no result, the MessageQueue-2 broker instance is stopped.

Tip – If the MessageQueue-2 broker instance is not stopped, kill the process using the following procedure.

a. Get the Java process IDs.

```
# ps -ef | grep java
```

b. Kill the Java process IDs that were returned.

```
# kill -9 #### ####
```

c. Run `netstat` again.

---

d. Restart the MessageQueue-2 broker instance.

```
# ./amfso start
```

e. Run the `netstat` command to verify that the Message Queue port is open and listening.

```
# netstat -an | grep 7777

*.7777            *.*             0             0    49152            0    LISTEN
```

17 Log out of the MessageQueue–2 host machine.

11.3 Configuring and Verifying Session Failover

Use the following list of procedures as a checklist for configuring and verifying session failover.

1. "To Configure Access Manager for Session Failover" on page 276
2. "To Verify That the Administrator Session Fails Over" on page 278
3. "To Verify that the User Session Fails Over" on page 279

▼ To Configure Access Manager for Session Failover

1 Access `http://LoadBalancer-3.example.com:7070/amserver/UI/Login` from a web browser.

2 Log in to the Access Manager console as the administrator.

Username **`amadmin`**

Password **`4m4dmin1`**

3 Click the Configuration tab.

4 Under Global properties, click Session.

5 Under Secondary Configuration Instance, click New.

6 In the Add Sub Configuration page, provide the following information.

Name	Enter the load balancer URL `https://loadbalancer-3.example.com:9443` **Tip –** The case of the load balancer URL should match that of the Primary Site ID.
Session Store User	Enter `msgquser`
Session Store Password	Enter `m5gqu5er`
Session Store Password (confirm)	Enter `m5gqu5er`
Maximum Wait Time	Keep the default value of 5000.
Database URL	Enter `MessageQueue-1.example.com:7777, MessageQueue-2.example.com:7777`. This is the Message Queue broker address list. Enter multiple values using a comma and no space.

7 Click Add.

8 Click Save.

9 Log out of the Access Manager console.

10 Restart the Web Server 1 instance.

a. Log in to the Access Manager 1 host machine.

b. Restart the Web Server 1 instance.

```
# cd /opt/SUNWwbsvr/https-AccessManager-1.example.com/bin
# ./stopserv; ./startserv
```

c. Log out of the Access Manager 1 host machine.

11 Restart the Web Server 2 instance.

a. Log in to the Access Manager 2 host machine.

b. **Restart the Web Server 2 instance.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-2.example.com/bin
# ./stopserv; ./startserv
```

c. **Log out of the Access Manager 2 host machine.**

▼ To Verify That the Administrator Session Fails Over

Before You Begin Both Access Manager 1 and Access Manager 2 should be up and running before you begin this verification procedure.

1 **As a root user, log in to the AccessManager–2 host machine.**

2 **Change to the `bin` directory.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-2.example.com/bin
```

3 **Stop Access Manager 2.**

```
# ./stopserv
```

4 **Access `http://LoadBalancer-3.example.com:7070/amserver/UI/Login` from a web browser.**

a. **Log in to the Access Manager console as the administrator.**

Username	**amadmin**
Password	**4m4dmin1**

b. **Click the Sessions tab.**

c. **In the View field, select `Access Manager-1.example.com:1080` from the drop down list.**

Verify that only `amadmin` exists in the Sessions table.

d. **In the View field, select `Access Manager-2.example.com:1080` from the drop down list.**

You will see an error message indicating the server is down.

e. **Leave this browser window 1 open.**

5 **Start Access Manager 2.**

```
# ./startserv
```

6 **As a root user, log in to the AccessManager–1 host machine.**

7 **Change to the `bin` directory.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-1.example.com/bin
```

8 **Stop Access Manager 1.**

```
# ./stopserv
```

9 **Going back to the Access Manager console in browser window 1, under the Sessions tab, select `Access Manager-1.example.com:1080` from the View drop down list.**

You will see an error message indicating the server is down.

10 **Now select `Access Manager-2.example.com:1080` from the View drop down list.**

Verify that only `amadmin` exists in the Sessions table. This indicates that although AccessManager–1 was stopped, the Access Manager LoadBalancer-3 directed the request to AccessManager–2 and a session for `amadmin` was successfully created in AccessManager–2. If session failover was not enabled, it would have resulted in a login page.

▼ To Verify that the User Session Fails Over

Before You Begin This procedure assumes that you have just completed "To Verify That the Administrator Session Fails Over" on page 278.

1 **Access `http://LoadBalancer-3.example.com:7070/amserver/UI/Login?realm=users` from a second browser window.**

2 **Log in to the Access Manager console as `testuser1`.**

Username	**`testuser1`**
Password	**`password`**

The Edit User page for `testuser1` is displayed. Because Access Manager 1 was stopped, the user session is created in Access Manager 2.

3 **Leave browser window 2 open.**

4 **Using browser window 1, click the Sessions tab.**

5 **In the View field, select `Access Manager-2.example.com:1080` from the drop down list.**

Verify that `amadmin` and `testuser1` exist in the Sessions table.

6 **On the AccessManager–1 host machine, change to the `bin` directory.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-1.example.com/bin
```

7 **Start AccessManager–1.**

```
# ./startserv
```

Both Access Manager–1 and Access Manager–2 are up and running.

8 **On the AccessManager–2 host machine, change to the `bin` directory.**

```
# cd /opt/SUNWwbsvr/https-AccessManager-2.example.com/bin
```

9 **Stop Access Manager–2.**

```
# ./stopserv
```

10 **Using browser window 1, click the Sessions tab.**

a. **In the View field, select `Access Manager-1.example.com:1080`.**

Verify that `amadmin` and `testuser1` exist in the Sessions table. This indicates that the session successfully failed over to AccessManager–1.

Tip – If `testuser1` is not displayed, refresh the browser window 2 page.

---

b. **In the View field, select `Access Manager-2.example.com:1080`**

You will see an error message indicating the server is down.

11 **Log out of the consoles and the host machines.**

PART III

Reference: Summaries of Server and Component Configurations

This section contains component descriptions and configurations for the software and hardware used in this deployment example.

- Appendix A, “Directory Servers”
- Appendix B, “Access Manager Servers”
- Appendix C, “Distributed Authentication User Interfaces”
- Appendix D, “Protected Resources”
- Appendix E, “Load Balancers”
- Appendix F, “Message Queue Servers”
- Appendix G, “Known Issues and Limitations”

APPENDIX A

Directory Servers

This appendix collects the information regarding the Directory Server instances. It contains the following tables:

- DirectoryServer–1 Host Machine Configuration
- DirectoryServer–2 Host Machine Configuration
- User Entries

TABLE A–1 DirectoryServer–1 Host Machine Configuration

Components	Description	
Host Name	DirectoryServer–1.example.com	
Installation Directory	/var/opt/mps/serverroot/	
Administrator User	cn=Directory Manager	
Administrator Password	d1rm4n4ger	
Access Manager Configuration Data Instance	Directory Server instance that stores Access Manager configuration data.	
	Instance Name	am-config
	Instance Directory	/var/opt/mps/am-config
	Port Number	1389
	Base Suffix	dc=example,dc=com
	Administrative User	cn=Directory Manager
	Administrative User Password	d1rm4n4ger
	Replication Manager	cn=replication manager,cn=replication,cn=config

TABLE A–1 DirectoryServer–1 Host Machine Configuration *(Continued)*

Components	Description	
	Replication Manager Password	replm4n4ger
User Data Instance	Directory Server instance that stores user data.	
	Note – In this deployment, user data is stored on the same host machine as the Access Manager configuration data. User data can also be stored on a different host machine.	
	Instance Name	am-users
	Instance Directory	/var/opt/mps/am-users
	Port Number	1489
	Base Suffix	dc=company,dc=com
	Users Suffix	ou=users,dc=company,dc=com
	Administrative User	cn=Directory Manager
	Administrative User Password	d1rm4n4ger
	Replication Manager	cn=replication manager,cn=replication,cn=config
	Replication Manager Password	replm4n4ger

TABLE A–2 DirectoryServer–2 Host Machine Configuration

Component	Description	
Host Name	DirectoryServer–2.example.com	
Installation Directory	/var/opt/mps/serverroot/	
Administrator User	cn=Directory Manager	
Administrator Password	d1rm4n4ger	
Access Manager Configuration Data Instance	Directory Server instance that stores Access Manager configuration data.	
	Instance Name	am-config
	Instance Directory	/var/opt/mps/am-config
	Port Number	1389
	Base suffix	dc=example,dc=com
	Administrative User	cn=Directory Manager

TABLE A–2 DirectoryServer–2 Host Machine Configuration *(Continued)*

Component	Description	
	Administrative User Password	d1rm4n4ger
	Replication Manager	cn=replication manager,cn=replication,cn=config
	Replication Manager Password	replm4n4ger
User Data Instance	Directory Server instance that stores user data.	
	Note – In this deployment, user data is stored on the same host machine as the Access Manager configuration data. User data can also be stored on a different host machine.	
	Instance Name	am-users
	Instance Directory	/var/opt/mps/am-users
	Port Number	1489
	Base Suffix	dc=company,dc=com
	Users Suffix	ou=users,dc=company,dc=com
	Administrative User	cn=Directory Manager
	Administrative User Password	d1rm4n4ger
	Replication Manager	cn=replication manager,cn=replication,cn=config
	Replication Manager Password	replm4n4ger

TABLE A–3 User Entries

UserID	Description	
testuser1	Used to verify that the policy agents work properly.	
	Password	password
	DN	uid=testuser1,ou=users,dc=company,dc=com
testuser2	Used to verify that the policy agents work properly.	
	Password	password
	DN	uid=testuser2,ou=users,dc=company,dc=com

APPENDIX B

Access Manager Servers

This appendix collects the information regarding the Access Manager servers. It contains the following tables:

- AccessManager–1 Host Machine Configuration
- AccessManager–2 Host Machine Configuration

TABLE B–1 AccessManager–1 Host Machine Configuration

Component	Description	
Host Name	AccessManager-1.example.com	
Non-Root User	am71adm	
Non-Root User Password	am71a6m	
Web Server Administration Server	Manages the Web Server application and all instances.	
	Instance Name	admin-server
	Instance Directory	/opt/SUNWwbsvr/admin-server
	SSL Port	8989
	SSL Service URL	https://AccessManager–1.example.com:8989
	Administrative User	admin
	Administrative User Password	web4dmin
Web Server Instance	Contains the deployed Access Manager applications	
	Instance name	AccessManager-1.example.com
	Instance Directory	/opt/SUNWwbsvr/https-AccessManager-1.example.com

TABLE B–1 AccessManager–1 Host Machine Configuration *(Continued)*

Component	Description	
	Port	1080
	Service URL	http://AccessManager-1.example.com:1080
	Administrative User	amadmin
	Administrative User Password	4m4dmin1
	Deployment URI	amserver

TABLE B–2 AccessManager–2 Host Machine Configuration

Component	Description	
Host Name	AccessManager-2.example.com	
Non-Root User	am71adm	
Non-Root User Password	am71a6m	
Web Server Administration Server	Manages the Web Server application and all instances.	
	Instance Name	admin-server
	Instance Directory	/opt/SUNWwbsvr/admin-server
	SSL Port	8989
	SSL Service URL	https://AccessManager–2.example.com:8989
	Administrative User	admin
	Administrative User Password	web4dmin
Web Server Instance	Contains the Access Manager applications	
	Instance Name	AccessManager-2.example.com
	Instance Directory	/opt/SUNWwbsvr/https-AccessManager-2.example.com
	Port	1080
	Service URL	http://AccessManager-2.example.com:1080
	Administrative User	amadmin
	Administrative User Password	4m4dmin1
	Deployment URI	amserver

APPENDIX C

Distributed Authentication User Interfaces

This appendix collects the information regarding the Distributed Authentication User Interfaces. It contains the following tables:

- AuthenticationUI-1 Host Machine Configuration
- AuthenticationUI-2 Host Machine Configuration

TABLE C–1 AuthenticationUI–1 Host Machine Configuration

Component	Description	
Host Name	AuthenticationUI-1.example.com	
Non-Root User	da71adm	
Non-Root User Password	6a714dm	
Web Server Administration Server	Manages the Web Server application and all instances.	
	Instance Name	admin-server
	Instance Directory	/opt/SUNWwbsvr/admin-server
	SSL Port	8989
	SSL Service URL	https://AuthenticationUI-1.example.com:8989
	Agent Profile	admin
	Agent Profile Password	web4dmin
Web Server Instance	Contains the Distributed Authentication User Interface module.	
	Instance Name	AuthenticationUI-1.example.com
	Instance Directory	/opt/SUNWwbsvr/https-AuthenticationUI-1.example.com
	Port	1080

TABLE C–1 AuthenticationUI–1 Host Machine Configuration *(Continued)*

Component	Description	
	Service URL	http://AuthenticationUI-1.example.com:1080
	Application User	authuiadmin
	Application User Password	4uthu14dmin
	Deployment URI	distAuth

TABLE C–2 AuthenticationUI–2 Host Machine Configuration

Component	Description	
Host Name	AuthenticationUI-2.example.com	
Non-Root User	da71adm	
Non-Root User Password	6a714dm	
Web Server Administration	Manages the Web Server and all its instances.	
	Instance Name	admin-server
	Instance Directory	/opt/SUNWwbsvr/admin-server
	Port Number	8989
	Service URL	https://AuthenticationUI-2.example.com:8989
	Administrative User	admin
	Administrative User Password	web4dmin
Web Server Instance	Contains the Distributed Authentication User Interface module.	
	Instance Name	AuthenticationUI-2.example.com
	Instance Directory	/opt/SUNWwbsvr/https-AuthenticationUI-2.example.com
	Port	1080
	Service URL	http://AuthenticaitonUI-2.example.com:1080
	Agent Profile	authuiadmin
	Agent Profile Password	4uthu14dmin
	Deployment URI	distAuth

APPENDIX D

Protected Resources

This appendix collects the information regarding the Protected Resource host machines. It contains the following tables:

- Protected Resource 1 Web Server and Web Policy Agent Host Machine Configurations
- Protected Resource 1 Application Server and J2EE Policy Agent Host Machine Configurations
- Protected Resource 2 Web Server and Web Policy Agent Host Machine Configurations
- Protected Resource 2 Application Server and J2EE Policy Agent Host Machine Configurations

TABLE D–1 Protected Resource 1 Web Server and Web Policy Agent Host Machine Configurations

Component	Description	
Host Name	ProtectedResource-1.example.com	
Web Server Administration Server	Manages the Web Server application and all instances.	
	Instance Name	admin-server
	Instance Directory	/opt/SUNWwbsvr/admin-server
	SSL Port	8989
	SSL Service URL	https://ProtectedResource-1.example.com:8989
	Administrative User	admin
	Administrative User Password	web4dmin
Web Server Instance	Contains the web policy agent.	
	Instance Name	ProtectedResource-1.example.com

TABLE D–1 Protected Resource 1 Web Server and Web Policy Agent Host Machine Configurations *(Continued)*

Component	Description	
	Instance Directory	/opt/SUNWwbsvr/https-ProtectedResource-1.example.com
	Port	1080
	Protected Resource URL	http://ProtectedResource–1.example.com:1080
	Web Agent Profile	webagent-1
	Web Agent Profile Password	web4gent1

TABLE D–2 Protected Resource 1 Application Server and J2EE Policy Agent Host Machine Configurations

Component	Description	
Host Name	ProtectedResource-1.example.com	
BEA WebLogic Application Server Home	/usr/local/bea/	
BEA WebLogic Application Server Domain	/usr/local/bea/user_projects/domains/ProtectedResource-1	
WebLogic Administration Server	Manages the domain and all managed servers	
	Server Name	AdminServer
	Server Directory	/usr/local/bea/user_projects/domains/ProtectedResource-1/servers/AdminServer
	Port	7001
	Console URL	http://protectedresource–1.example.com:7001/console
	Administrative User	weblogic
	Administrative User Password	w3bl0g1c
WebLogic Managed Server	Contains configuration information for this managed server and the J2EE Policy Agent.	
	Server Name	ApplicationServer-1
	Server Directory	/usr/local/bea/user_projects/domains/ProtectedResource-1/servers/ApplicationServer-1
	Port	1081
	J2EE Policy Agent Profile	j2eeagent-1

TABLE D–2 Protected Resource 1 Application Server and J2EE Policy Agent Host Machine Configurations *(Continued)*

Component	Description	
	J2EE Policy Agent Profile Password	j2ee4gent1

TABLE D–3 Protected Resource 2 Web Server and Web Policy Agent Host Machine Configurations

Component	Description	
Host Name	ProtectedResource-2.example.com	
Web Server Administration Server	Manages the Web Server application and all instances.	
	Instance Name	admin-server
	Instance Directory	/opt/SUNWwbsvr/admin-server
	SSL Port	8989
	SSL Service URL	https://ProtectedResource-2.example.com:8989
	Administrative User	admin
	Administrative User Password	web4dmin
Web Server Instance	Contains the web policy agent.	
	Instance Name	ProtectedResource-2.example.com
	Instance Directory	/opt/SUNWwbsvr/https-ProtectedResource-2.example.com
	Port	1080
	Protected Resource URL	http://ProtectedResource–2.example.com:1080
	Web Agent Profile	webagent-2
	Web Agent Profile Password	web4gent2

TABLE D–4 Protected Resource 2 Application Server and J2EE Policy Agent Host Machine Configurations

Component	Description
Host Name	ProtectedResource-2.example.com
BEA WebLogic Application Server Home	/usr/local/bea/
BEA WebLogic Application Server Domain	/usr/local/bea/user_projects/domains/ProtectedResource-2

TABLE D–4 Protected Resource 2 Application Server and J2EE Policy Agent Host Machine Configurations *(Continued)*

Component	Description	
WebLogic Administration Server	Manages the domain and all managed servers	
	Server Name	AdminServer
	Server Directory	/usr/local/bea/user_projects/domains/ProtectedResource-2/servers/AdminServer
	Port	7001
	Console URL	http://protectedresource–2.example.com:7001/console
	Administrative User	weblogic
	Administrative User Password	w3bl0g1c
WebLogic Managed Server	Contains configuration information for this managed server and the J2EE Policy Agent.	
	Server Name	ApplicationServer-2
	Server Directory	/usr/local/bea/user_projects/domains/ProtectedResource-2/servers/ApplicationServer-2
	Port	1081
	J2EE Policy Agent Profile	j2eeagent-2
	J2EE Policy Agent Profile Password	j2ee4gent2

APPENDIX E

Load Balancers

This appendix collects the information regarding the load balancers. It contains the following table:

- Load Balancer Configurations

The BIG-IP load balancer login page and configuration console for all load balancers in this deployment example is accessed from the URL, `is-f5.example.com`.

Login	username
Password	password

TABLE E–1 Load Balancer Configurations

Load Balancer	Description	
Load Balancer 1	Distribution for the two Directory Server instances that contain Access Manager configuration data instance.	
	Virtual Server	LoadBalancer-1.example.com
	Port	389
	Pool Name	DirectoryServer-ConfigData-Pool
	Access URL	LoadBalancer-1.example.com:389
	Monitor	ldap-tcp
Load Balancer 2	Distribution for the two Directory Server instances that contains user data.	
	Virtual Server	LoadBalancer-2.example.com
	Port	489
	Pool Name	DirectoryServer-UserData-Pool

TABLE E–1 Load Balancer Configurations *(Continued)*

Load Balancer	Description	
	Access URL	LoadBalancer-2.example.com:489
	Monitor	ldap-tcp
Load Balancer 3	Distribution for the two Web Server applications installed on the Access Manager host machines. **Note –** SSL is terminated at this load balancer before the request is forwarded to Access Manager. This load-balancer is the single point-of-failure for Access Manager and can be considered a limitation of this deployment example.	
	Virtual Server	LoadBalancer-3.example.com
	Port (external access)	9443
	Port (internal access)	7070
	Pool Name	AccessManager-Pool
	External Access URL	LoadBalancer-3.example.com:9443
	Internal Access URL	LoadBalancer-3.example.com:7070
	Monitor	AccessManager-http
Load Balancer 4	Distribution for the two Web Server applications installed on the Distributed Authentication UI host machines. **Note –** SSL is terminated at this load balancer before the request is forwarded to the Distributed Authentication User Interface.	
	Virtual Server	LoadBalancer-4.example.com
	Port (external access)	9443
	Port (internal access)	90
	Pool Name	AuthenticationUI-Pool
	External Access URL	LoadBalancer-4.example.com:9443
	Internal Access URL	LoadBalancer-4.example.com:90
	Monitor	HTTP
Load Balancer 5	Distribution for Web Policy Agents.	
	Virtual Server	LoadBalancer-5
	Port	90
	Pool Name	WebAgent-Pool

TABLE E–1 Load Balancer Configurations *(Continued)*

Load Balancer	Description	
	Access URL	LoadBalancer-5.example.com:90
	Monitor	WebAgent-http
Load Balancer 6	Distribution for J2EE Policy Agents	
	Virtual Server	LoadBalancer-6
	Port	91
	Pool Name	J2EEAgent-Pool
	Access URL	LoadBalancer-6.example.com:91
	Monitor	tcp

APPENDIX F

Message Queue Servers

Message Queue serves as a communications broker that enables Access Manager to communicate data with the session store. This appendix collects the information regarding the Message Queue servers. It contains the following tables:

- Message Queue 1 Host Machine Configuration
- Message Queue 2 Host Machine Configuration

TABLE F–1 Message Queue 1 Host Machine Configuration

Component	Description
Host Name	MessageQueue-1.example.com
Session Tools Scripts Directory	/export/AMSFO/amSessionTools/amserver
Message Queue Directory	/export/AMSFO/amSessionTools/jmq
Berkeley Database Directory	/export/AMSFO/amSessionTools/bdb
Instance Name	msgqbroker
Port Number	7777
Administrative User	msgquser
Administrative User Password	m5gqu5er

TABLE F–2 Message Queue 2 Host Machine Configuration

Component	Description
Host Name	MessageQueue-2.example.com

TABLE F–2 Message Queue 2 Host Machine Configuration *(Continued)*

Component	Description
Session Tools Scripts Directory	/export/AMSFO/amSessionTools/amserver
Message Queue Directory	/export/AMSFO/amSessionTools/jmq
Berkeley Database Directory	/export/AMSFO/amSessionTools/bdb
Instance Name	msgqbroker
Port Number	7777
Administrative User	msgquser
Administrative User Password	m5gqu5er

◆ ◆ ◆ APPENDIX G

Known Issues and Limitations

The issues in this appendix will be updated as more information becomes available.

TABLE G–1 Known Issues and Limitations

Reference Number	Description
6462076	**Single WAR Configurator fails against Directory Server** Access Manager, when deployed as a single WAR, will not configure Directory Server 6 with a single component root suffix (as in `dc=example`) although it works as expected with multi-component root suffixes (as in `dc=example,dc=com`). **Workaround:** Use multi-component root suffixes.
6472662	**When SSL terminates at the Access Manager load balancer, the console application changes protocol from HTTPS to HTTP.** When you try to access the Access Manager load balancer with a URL such as **`https://`***loadbalancer***`:`***port***`/amserver/console`** or **`https://`***loadbalancer***`:`***port***`/amserver/UI/Login`**, you cannot access the login page because the console application changes the protocol from HTTPS to HTTP. **Workaround:** Add `<property name="relativeRedirectAllowed" value="true"/>` to the `sun-web.xml` file for the individual instances of Access Manager and restart them. **Caution –** After applying the workaround, the only supported URL is **`https://`***loadbalancer***`:`***port***`/amserver/UI/Login`**. It is highly recommended that you access the Access Manager instances directly to perform any administrative tasks rather than accessing them through a load balancer. This workaround was tested on Sun Java Systems Web Server 7.

TABLE G–1 Known Issues and Limitations *(Continued)*

Reference Number	Description
6476271	**BEA servers do not start up when startup script is not configured properly.** The BEA administration server and managed server instances will not start up if the start up script is not configured properly. When using J2EE Policy Agent 2.2 on BEA Application Server 9.2, you must append the following to the end of the `setDomainEnv.sh` file: ■ `. /usr/local/bea/user_projects/domains/ProtectedResource-1/bin/setAgentEnv_ApplicationServer-1.sh` for Protected Resource 1. ■ `. /usr/local/bea/user_projects/domains/ProtectedResource-2/bin/setAgentEnv_ApplicationServer-2.sh` for Protected Resource 1. The `setDomainEnv.sh` file contains the call to `commEnv.sh`.

6477741	**Exception is thrown when you run the agentadmin utility.**.
	The following exception is thrown when you run the `agentadmin` utility from the J2EE Policy Agent 2.2 server (BEA Appserver 9.2).
	`#` **`./agentadmin --getUuid amadmin user example`**
	
	`Failed to create debug directory`
	`Failed to create debug directory`
	`Failed to create debug directory`
	`Failed to create debug directory`
	`Failed to create debug directory`