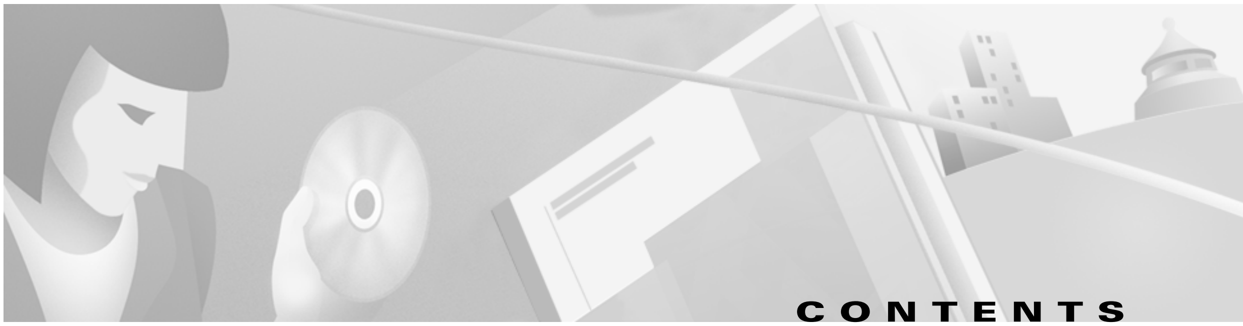
Cisco Subscriber Edge Services Manager and Subscriber Policy Engine Installation and Configuration Guide

SESM Release 3.1(1) and SPE Version 1.0
August 2001

Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

Customer Order Number: DOC-7812342=
Text Part Number: 78-12342-03

Cisco Subscriber Edge Services Manager and Subscriber Policy Engine Installation and Configuration Guide



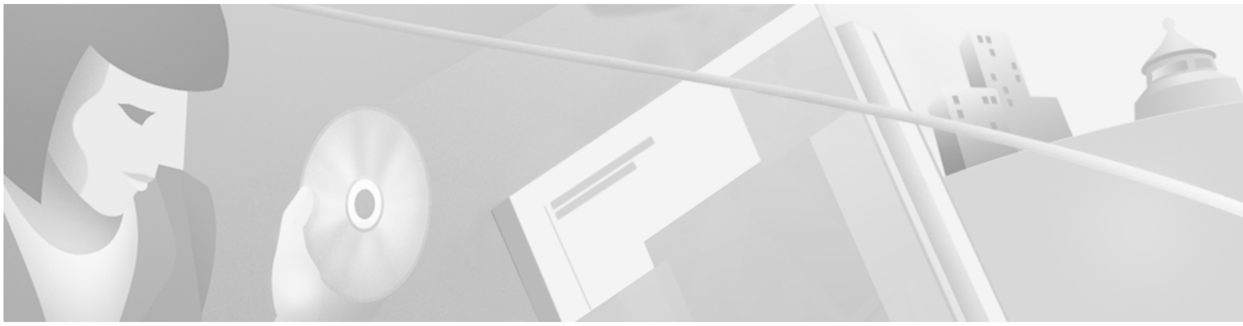
CONTENTS

CHAPTER 3

CHAPTER 4

CHAPTER 6

CHAPTER 7



About This Guide

This preface introduces the *Cisco Subscriber Edge Services Manager and Subscriber Policy Engine Installation and Configuration Guide.* The preface contains the following sections:

- Document Objectives
- Audience
- Document Organization
- Document Conventions
- Related Documentation
- Obtaining Documentation
- Obtaining Technical Assistance

Document Objectives

This guide explains how to install and configure Cisco Subscriber Edge Services Manager (Cisco SESM) applications and related components. Internet service providers (ISPs) and network access providers (NAPs) deploy SESM to provide their end users (subscribers) with a single web interface for accessing multiple Internet services.

Audience

This guide is intended for administrators and others responsible for:

- Installing and running the New World Service Provider (NWSP) sample application in Demo mode, which simulates communication with other network components
- Installing, configuring, and running the New World Service Provider (NWSP) sample application in RADIUS or DESS mode, both of which require communication with other network components
- Deploying a customized SESM application

Document Organization

This guide includes the chapters shown in the following table:

Chapter	Title	Description
Chapter 1	Overview	This chapter describes the features and components of the Cisco SESM.
Chapter 2	Demo Quick Start	This chapter describes procedures for installing and running the NWSP sample application in Demo mode.
Chapter 3	Installing Components	This chapter describes how to install the Cisco SESM software.
Chapter 4	Configuring Components after Installation	This chapter describes all of the configurable attributes in the SESM software components. Use this chapter to change or fine tune attributes after installation.
Chapter 5	Running SESM Components	This chapter describes how to start and stop the SESM software components.
Chapter 6	SESM Applications	This chapter describes how to access the NWSP sample application from a web browser. It also discusses configuration requirements for a customized SESM application.
Chapter 7	Troubleshooting Installation and Configuration	This appendix includes some troubleshooting hints.
Appendix A	Security	This appendix describes the security features in an SESM web application.
Appendix B	Configuring the SSG	This appendix describes how to configure SSG to communicate with an SESM web application.
Appendix C	DTD for MBean Configuration Files	This appendix shows the XML document type definition (DTD) for the MBean configuration files used to configure the SESM software components.
Appendix D	Configuring RADIUS	This appendix describes how to configure a RADIUS server to communicate with: • SSG and SESM web applications running in RADIUS mode. • SSG for accounting purposes, which is appropriate for SESM deployments in both RADIUS and DESS modes.
Appendix E	RDP Packet Handlers	This appendix describes how the RDP processes requests.
Appendix F	Sample MBean Configuration Files	This appendix contains sample configuration files.
Index		

Document Conventions

The following conventions are used in this guide:

- *Italic* font is used for parameters for which you supply a value, emphasis, and to introduce new terms.
- `Computer` font is used for examples.

Note Means reader take note. Notes contain helpful suggestions or references to materials not contained in the manual.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Related Documentation

Documentation for the Cisco SESM includes:

- *Release Notes for the Cisco Subscriber Edge Services Manager, Release 3.1(1)*
- *Cisco Subscriber Edge Services Manager Web Developer Guide*
- *Cisco Distributed Administration Tool Guide*
- *Cisco Subscriber Edge Services Manager and Subscriber Policy Engine Installation and Configuration Guide* (this manual)

The Cisco SSG is a required network component in SESM deployments. Cisco SSG is a feature embedded in the Cisco IOS software running on a Node Route Processor (NRP) in the Cisco 6400 Universal Access Concentrator. Documentation for the Cisco SSG includes:

- *Cisco 6400 Feature Guide*—This guide includes a chapter that documents SSG features. The online link to this guide is:

  http://www.cisco.com/univercd/cc/td/doc/product/dsl_prod/6400/feat_gd/12_1_5/index.htm
- *Cisco 6400 Command Reference*—This guide includes a chapter that documents SSG configuration commands. The online link to this guide is:

  http://www.cisco.com/univercd/cc/td/doc/product/dsl_prod/6400/commandr/index.htm
- *Cisco 6400 NRP—Release Notes for Cisco IOS Release 12.1(5)DC*—The online link to these release notes is:

  http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121relnt/6400/rn121dc5.htm

If you are including the Cisco Access Registrar (a RADIUS server) in your SESM deployement, see the following documents:

- *Cisco Access Registrar 1.6 Release Notes*
- *Cisco Access Registrar User Guide*

Obtaining Documentation

The following sections provide sources for obtaining documentation from Cisco Systems.

World Wide Web

You can access the most current Cisco documentation on the World Wide Web at the following sites:

- http://www.cisco.com
- http://www-china.cisco.com
- http://www-europe.cisco.com

Documentation CD-ROM

Cisco documentation and additional literature are available in a CD-ROM package, which ships with your product. The Documentation CD-ROM is updated monthly and may be more current than printed documentation. The CD-ROM package is available as a single unit or as an annual subscription.

Ordering Documentation

Cisco documentation is available in the following ways:

- Registered Cisco Direct Customers can order Cisco Product documentation from the Networking Products MarketPlace:

  http://www.cisco.com/cgi-bin/order/order_root.pl

- Registered Cisco.com users can order the Documentation CD-ROM through the online Subscription Store:

  http://www.cisco.com/go/subscription

- Nonregistered Cisco.com users can order documentation through a local account representative by calling Cisco corporate headquarters (California, USA) at 408 526-7208 or, in North America, by calling 800 553-NETS (6387).

Documentation Feedback

If you are reading Cisco product documentation on the World Wide Web, you can submit technical comments electronically. Click **Feedback** in the toolbar and select **Documentation**. After you complete the form, click **Submit** to send it to Cisco.

You can e-mail your comments to bug-doc@cisco.com.

To submit your comments by mail, use the response card behind the front cover of your document, or write to the following address:

Attn Document Resource Connection
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-9883

We appreciate your comments.

Obtaining Technical Assistance

Cisco provides Cisco.com as a starting point for all technical assistance. Customers and partners can obtain documentation, troubleshooting tips, and sample configurations from online tools. For Cisco.com registered users, additional troubleshooting tools are available from the TAC website.

Cisco.com

Cisco.com is the foundation of a suite of interactive, networked services that provides immediate, open access to Cisco information and resources at anytime, from anywhere in the world. This highly integrated Internet application is a powerful, easy-to-use tool for doing business with Cisco.

Cisco.com provides a broad range of features and services to help customers and partners streamline business processes and improve productivity. Through Cisco.com, you can find information about Cisco and our networking solutions, services, and programs. In addition, you can resolve technical issues with online technical support, download and test software packages, and order Cisco learning materials and merchandise. Valuable online skill assessment, training, and certification programs are also available.

Customers and partners can self-register on Cisco.com to obtain additional personalized information and services. Registered users can order products, check on the status of an order, access technical support, and view benefits specific to their relationships with Cisco.

To access Cisco.com, go to the following website:

http://www.cisco.com

Technical Assistance Center

The Cisco TAC website is available to all customers who need technical assistance with a Cisco product or technology that is under warranty or covered by a maintenance contract.

Contacting TAC by Using the Cisco TAC Website

If you have a priority level 3 (P3) or priority level 4 (P4) problem, contact TAC by going to the TAC website:

http://www.cisco.com/tac

P3 and P4 level problems are defined as follows:

- P3—Your network performance is degraded. Network functionality is noticeably impaired, but most business operations continue.
- P4—You need information or assistance on Cisco product capabilities, product installation, or basic product configuration.

In each of the above cases, use the Cisco TAC website to quickly find answers to your questions.

To register for Cisco.com, go to the following website:

http://www.cisco.com/register/

If you cannot resolve your technical issue by using the TAC online resources, Cisco.com registered users can open a case online by using the TAC Case Open tool at the following website:

http://www.cisco.com/tac/caseopen

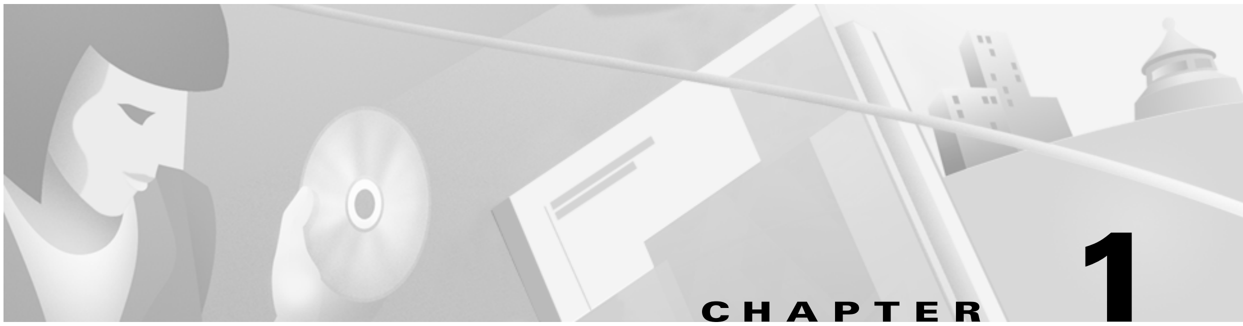
Contacting TAC by Telephone

If you have a priority level 1(P1) or priority level 2 (P2) problem, contact TAC by telephone and immediately open a case. To obtain a directory of toll-free numbers for your country, go to the following website:

http://www.cisco.com/warp/public/687/Directory/DirTAC.shtml

P1 and P2 level problems are defined as follows:

- P1—Your production network is down, causing a critical impact to business operations if service is not restored quickly. No workaround is available.
- P2—Your production network is severely degraded, affecting significant aspects of your business operations. No workaround is available.



CHAPTER 1

Overview

This chapter describes the features and components in the Cisco Subscriber Edge Services Manager (Cisco SESM) Release 3.1(1) and Cisco Subscriber Policy Engine (Cisco SPE) Version 1.0. The chapter includes the following topics:

SESM and SPE Product Descriptions

This section introduces the SESM product. It includes the following topics:

Introduction to SESM

The Cisco Subscriber Edge Services Manager (SESM) works in conjunction with other network components to provide extremely robust, highly scalable connection management to Internet services.

Internet service providers (ISPs) and network access providers (NAPs) deploy SESM to provide their subscribers with a web interface for accessing multiple Internet services. The ISPs and NAPs can customize and brand the content of the web pages and thereby control the user experience for different categories of subscribers.

SESM Core Components

SESM Release 3.1(1) is a solution composed of a number of applications built on a core set of software components. ISPs and NAPs can use these core components to further develop and customize SESM web applications, if required. The *Cisco Subscriber Edge Services Manager Web Developer Guide* describes how to develop SESM applications.

An SESM solution is deployed with the Cisco Service Selection Gateway (SSG), a Cisco IOS feature on the Cisco 6400 Universal Access Concentrator (UAC). Together, SESM and SSG provide subscriber authentication, service selection, and service connection capabilities to subscribers of Internet services.

Subscribers interact with an SESM web application using a standard Internet browser. They do not need to download any software or plug-ins to use the SESM web pages. After a subscriber successfully authenticates, the SESM web application presents a list of services that the subscriber is currently authorized to use. The subscriber can gain access to one or more of those services by selecting them from a web page. Alternatively, an automatic connection feature might provide automatic connection to services.

SESM Release 3.1(1) web applications deployed in Directory Enabled Service Selection/Subscription (DESS) mode incorporate the use of the Cisco Subscriber Policy Engine (SPE) Version 1.0. The SPE allows subscribers to perform account maintenance and self-care activities, such as subscribing to new services, creating subaccounts (for other members of the family, for example), and changing basic account information, such as address, phone number, and e-mail.

For subscribers of Internet services, an SESM web application offers flexibility and convenience, including the ability to access multiple services simultaneously.

For Internet service providers, an SESM web application provides a way to control the subscriber experience and promote customer loyalty. Service providers can change the look and feel of their SESM web application, brand the application, and control the content of the pages displayed to their subscribers.

Note The SESM product was previously called the Cisco Service Selection Dashboard (Cisco SSD).

Cisco Subscriber Policy Engine

The Cisco Subscriber Policy Engine (SPE) Version 1.0 is a policy server specifically customized to provide granular subscriber service policy. SPE combines role-based access control (RBAC) functionality with an open policy server. Service providers can create differentiated subscriber groups. Service and content providers can use the SPE to provide value added and differentiated services to the subscriber population.

SPE is installed when SESM Release 3.1(1) is deployed in DESS mode to provide the following enhanced features and capabilities:

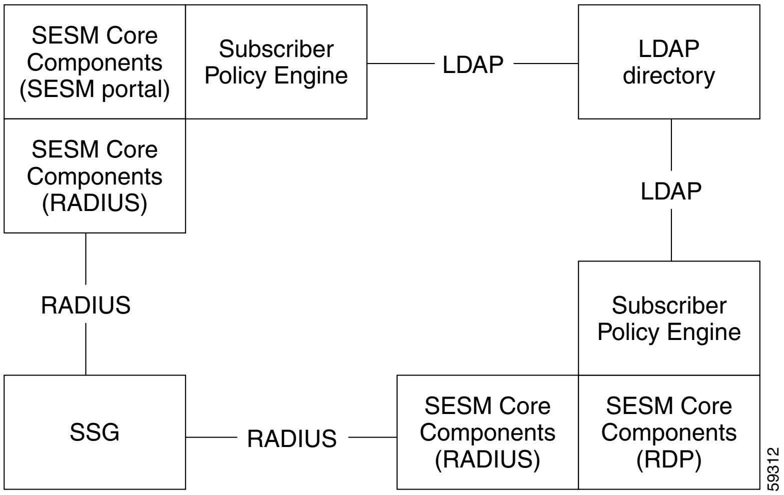
- Use of an LDAP directory to manage subscriber, service profile, and policy information
- Subscriber account self-care

- Subscriber sub-account management
- Subscriber self-subscription to services
- Bulk administration of large subscriber populations
- Delegated administration
- Allow service publishers and business partners access to service creation and management
- Allow service providers and business partners to publish services to targeted subscribers

In the SESM product and its documentation, the SPE components and features are called the DESS components and features.

Figure 1-1 shows the relationship between the SESM and SPE products.

Figure 1-1 SESM in DESS Mode Components


SESM Core Components (SESM portal)
Subscriber Policy Engine
LDAP
LDAP directory
SESM Core Components (RADIUS)
LDAP
RADIUS
Subscriber Policy Engine
SSG
RADIUS
SESM Core Components (RADIUS)
SESM Core Components (RDP)
59312

New World Service Provider Sample Application

The SESM installation package includes a sample SESM web application, called the New World Service Provider (NWSP), that you can configure and subsequently execute as an example of SESM capabilities. You can create the desired look-and-feel and branded aspects of a customized SESM application by altering the sample application or writing your own application using the NWSP as an example.

Captive Portal Sample Application

The SESM installation package includes a captive portal sample application. This application demonstrates how several powerful features in SESM Release 3.1(1) work together to redirect unauthenticated users to an SESM sign-on page immediately after they open a web browser. See the "Key Features" section on page 1-8 for more information about this and other SESM features.

Demo Installation

The SESM installation package provides an option to install the NWSP and captive portal sample applications in Demo mode. Demo mode simulates the actions of an SESM application without requiring additional network components. Demo mode is intended for demonstration purposes only and does not represent SESM performance in a production environment.

SESM Deployment Modes

The SESM Release 3.1(1) solution can be deployed in these modes:

- RADIUS deployment mode—This mode obtains subscriber and service profile information from a RADIUS server.
- DESS deployment mode—The Directory-Enabled Service Selection (DESS) mode integrates the Cisco Subscriber Policy Engine (SPE) Version 1.0 product with the SESM product to provide access to an LDAP compliant directory for subscriber and service profile information. SPE also provides enhanced functionality for SESM web applications and use of the role-based access control (RBAC) model to manage subscriber access.
- Demo mode—This mode demonstrates the capabilities of both RADIUS and DESS modes without requiring additional external components, such as SSG, a RADIUS server, or an LDAP directory server.

The SESM core model implements these modes in a plug-in style. Web developers use the same SESM application programming interface (API) to develop applications intended for either the RADIUS or the DESS modes. Applications intended for DESS mode deployment can include additional features provided by SPE. The *Cisco Subscriber Edge Services Manager Web Developer Guide* describes how to create applications for both RADIUS and DESS mode deployments.

The deployment option affects the following aspects of product installation and configuration:

- The SESM software components that you install and configure—The DESS deployment includes several additional software components to install and configure, all of which are included in the SESM installation package and described in this guide.
- The values of configuration parameters for the SESM software components.
- The network components that you are required to install, configure, and populate with subscriber and service profile information—The RADIUS mode requires SSG and a RADIUS server. The DESS mode requires SSG and an LDAP-compliant directory. Demo mode does not require any additional network components.

SESM Using an External RADIUS Server—RADIUS Mode

In a RADIUS deployment, a RADIUS server stores subscriber and service profiles. RADIUS refers to the Remote Dial-In User Service (RADIUS) database and server that performs authentication, authorization, and accounting (AAA) services for network connections. An SESM deployment works with any RADIUS server that accepts vendor-specific attributes (VSAs).

See the "SESM in RADIUS Mode" section on page 1-13 for more information about the components and data flow in a RADIUS mode deployment.

SESM Integrated with SPE—DESS Mode

In a DESS deployment, a directory stores subscriber and service profile information. The directory must be a Lightweight Directory Access Protocol (LDAP)-compliant directory.

A DESS deployment requires the Cisco Directory Enabled Service Selection/Subscription (DESS) component. You can install the DESS component from the SESM installation package if your SESM purchase license allows it. The DESS component is the Cisco Subscriber Policy Engine (SPE) Version 1.0, packaged for inclusion in the SESM product package.

See the "SESM in DESS Mode" section on page 1-15 for more information about the components and data flow in a DESS mode deployment.

J2EE and JMX Server Requirements

J2EE Server

SESM web applications are J2EE applications, requiring a J2EE-compliant server.

The NWSP sample application, configuration files, and startup scripts are configured to use the Jetty server components from Mort Bay Consulting. You can install the Jetty server using the SESM installation program. If desired, web developers at your site can deploy a J2EE-compliant server other than the Jetty server.

See the "Host Key Feature on SSG" section on page 1-6 before deploying a J2EE server other than the Jetty server.

JMX Server

SESM web applications require the services of a Java Management Extensions (JMX) server.

The installed NWSP sample application, the configuration files, and the startup scripts are set up to use the Sun example JMX server from Sun Microsystems. The SESM installation program installs the JMX server along with the Jetty server. If desired, web developers at your site can deploy a JMX-compliant server other than the Sun example server.

Additional Required Network Software

This section describes the network software that is required in an SESM deployment but is not provided by the SESM installation package.

- Cisco Service Selection Gateway, page 1-5
- Cisco Access Registrar or Third-Party RADIUS Server, page 1-6
- LDAP Directory, page 1-7

Cisco Service Selection Gateway

The Cisco Service Selection Gateway (SSG) is a software feature module embedded in Cisco IOS software running on the Cisco 6400 Universal Access Concentrator (UAC). Each node route processor on the Cisco 6400 UAC can host an SSG. The SSG configured with the Web Selection option works in conjunction with SESM.

SSG performs authentication and service connection tasks on behalf of an SESM application.

Required Cisco IOS Release

SESM Release 3.1(1) requires the SSG feature set embedded in Cisco IOS Release 12.1(5)DC1 or later. For information about this release of SSG, see the following documents.

- *Cisco 6400 Feature Guide*—This guide includes a chapter that documents SSG features.
- *Cisco 6400 Command Reference*—This guide includes a chapter that documents SSG configuration commands.
- *Cisco 6400 NRP—Release Notes for Cisco IOS Release 12.1(5)DC*

The "Related Documentation" section on page xiii provides URLs to the online location of these documents.

Communication Protocol

Regardless of the SESM deployment mode (RADIUS or DESS), SSG and an SESM web application communicate using the RADIUS protocol.

Host Key Feature on SSG

The host key is an important feature on the SSG. It uses a software token (or key) that uniquely identifies each subscriber on the host SSG currently logged on to SESM, even when multiple subscribers are using the same IP address. The host key feature also provides an SSG IP address in the key.

The host key feature provides the following advantages to SESM applications:

- Host key allows SESM applications to robustly handle overlapping IP addresses, nonroutable IP addresses, and dynamically assigned IP addresses.
- Host key eliminates the need to explicitly map subscriber subnets to SSGs.

Note The host key feature is planned for general availability in Cisco IOS Release 12.2(2)B.

When host key is enabled on the SSG, the SSG preserves the port number of the incoming HTTP request. This remote port number becomes the key that uniquely identifies each subscriber. The key is included in the request that is forwarded to the SESM web application.

The SSG makes the port number available, but the J2EE server must access this information and pass it along to the SESM web application. The Jetty server has been extended to allow access to the request handling part of the server API and thus get the remote port number. It does this with its PortBundleHandler. Therefore, only the Jetty server can support the host key feature.

Cisco Access Registrar or Third-Party RADIUS Server

The following scenarios require a RADIUS server:

- An SESM web application deployed in RADIUS mode—This deployment requires user and service profile information in a RADIUS database.
- An SESM web application deployed in DESS mode with an RDP running in Proxy Mode—This deployment requires user profiles in a RADIUS database. In Proxy mode, the RDP proxies authentication requests to a RADIUS database. RDP obtains service authorizations through DESS, based on the information in the directory.

- An SESM web application deployed in either mode when you want to use the SSG accounting features—For any SESM deployment, you can configure the SSG to generate accounting records and send them to a RADIUS server. The SSG accounting features are implemented independently from the SESM web application.

SESM works with any RADIUS server that accepts vendor-specific attributes (VSAs). The VSAs define the subscriber and service profile information required in the SESM deployment. The Cisco Access Registrar is a carrier class RADIUS platform that is fully tested with SESM. See the "Configuring Cisco Access Registrar for SESM Deployments" section on page D-11 for more information about using Cisco Access Registrar in SESM deployments.

Also see the following references for more information about configuring a RADIUS server in an SESM deployment:

- Appendix D, "Configuring RADIUS"—Describes the Cisco VSAs required in an SESM deployment. It also describes how to configure a RADIUS server for an SESM deployment.
- demo.txt file—Contains examples of subscriber and service profiles. This file is a MERIT flat file used by the NWSP sample application when it runs in Demo mode. The demo.txt file is included in your installation directory even if you do not specify demo mode at installation time. You can find demo.txt in the config directory under the nwsp directory (for example, nwsp/config/demo.txt).

LDAP Directory

An SESM web application deployed in DESS mode requires access to an LDAP-compliant directory. SESM is verified and officially supported to work with the Network Directory Service (NDS) eDirectory Version 8.5 from Novell, Inc. Although initial testing with the iPlanet Directory Server Version 5.0 indicates excellent results, Cisco has not fully verified it in an SESM deployment.

An LDAP directory allows interactive updates, a feature that is not supported by a RADIUS server. The DESS mode uses this update capability to offer SESM features that the RADIUS mode cannot provide, such as:

- Subscriber account self care features—Subscribers can change their account information and see those changes take effect immediately.
- Subscriber self subscription—Subscribers can subscribe to new services and have immediate access to the newly subscribed services.
- Sub-account creation—Subscribers can create sub-accounts to their main account and use the sub-accounts immediately.

Key Features

Table 1-1 describes the key features in SESM Release 3.1(1). For information about how to enable and configure these features, see Table 4-9 on page 4-42.

Table 1-1 Features in SESM Release 3.1(1) and SPE 1.0

Feature	Description
Multiple Internet service selection	An SESM web application provides a web portal from which subscribers can: • Authenticate or verify their identity • Select one or more services for connection • See which services are active in their current session and other session status information An SESM web application works in conjunction with SSG to authenticate the subscriber, to obtain the list of services that the subscriber is authorized to use, and to obtain session status information. The SESM application sends service connection requests to SSG, which makes the actual connection.
Java Server Pages (JSPs)	JSPs provide a standard way to integrate Java code with HTML to present interactive, dynamically updated, personalized, and branded web pages to your subscribers.
Walled gardens, open gardens, retail pages, and service advertisements	The following features are implemented through the use of customized JSPs: • Walled Gardens—Service providers can customize the look and feel of the walled garden presentation to subscribers by altering the JSPs. Walled gardens are the services available to a subscriber that require authentication. The specific services available to each subscriber are configured in subscriber profiles and are not affected by the JSPs. • Open Gardens—Service providers can use SESM to offer open gardens, branded offerings of value-added services that do not require authentication and might be specific to the service provider. Links to these services can appear on a pre-authentication page, or you can customize the post authentication pages to include the open gardens. • Retail Pages—Wholesale providers can offer retail pages with a customized look and feel for each Internet service provider. • Service Advertisement—Service providers can use SESM to reach subscribers with targeted messages and thereby increase the acceptance of new services.

Table 1-1 Features in SESM Release 3.1(1) and SPE 1.0 (continued)

Feature	Description
Captive portal	This feature works with the TCP redirect feature on the SSG to redirect HTTP requests for unauthenticated subscribers. • The TCP redirect feature on the SSG redirects incoming TCP packets to a specified SESM web application. With TCP redirect, service providers do not need to provide their subscribers with a URL to the SESM logon page. The subscribers are sent automatically to the logon page when they start a browser session. The TCP redirect feature in Cisco IOS Release 12.1(5)DC1 can redirect packets originating from unauthorized users, which, in effect, redirects packets from subscribers when they first open their Internet browsers and are not yet authenticated by SESM. Future releases will allow redirection based on the packet's source network or destination port. • If the SESM web application is running in captive portal mode, it has an associated captive portal application. The captive portal application: – Captures the original URL in the subscriber's request. For example, subscribers might have a home page setting, or they might open a browser and immediately enter a URL to a specific service or Internet reference page. (Original URLs are lost if you implement TCP redirect without captive portal.) – Redirects the browser to the authentication page of the main SESM application. – Includes the original URL in the redirect request, making this information available to the SESM web application. The NWSP sample application redirects the browser to the originally requested URL after successful authentication, thus honoring home page settings. You could customize your SESM web application to use this information in other ways.
Device and locale awareness	An SESM web application can detect a subscriber's preferred locale, device and browser type, and connection location and respond with web pages appropriate to the subscriber's preferred language, device capabilities, and connection type.
Single sign-on in a point-to-point (PPP) network	This feature offers a streamlined login procedure in a PPP network. A subscriber who logs on using a PPP client can access the SESM without having to re-enter the username and password.
Host key port bundle	This feature on the SSG ensures that each currently logged-on subscriber is uniquely identified, regardless of the IP address being used. This SSG feature allows SESM applications to support the following types of subscribers: • Overlapping IP addresses in PPP and bridged environments—SESM can differentiate between various subscribers using the same IP address. • Nonroutable subscriber IP addresses—SESM supports subscribers at sites using private IP addressing schemes, including subscribers of ISPs using private addressing schemes. • Dynamic IP address assignment—The subscriber session state status within SSG and SESM remains synchronized when a subscriber's IP address changes. This feature also enhances scaling and configuration of large SESM deployments.

Table 1-1 Features in SESM Release 3.1(1) and SPE 1.0 (continued)

Feature	Description
Highly scalable	An SESM web server application is highly scalable in the following ways: • SESM leverages the load-balancing features of J2EE technology. • When the SSG host key feature is enabled, SESM applications are completely stateless regarding subscriber sessions. SSG signals the SESM application whenever state changes occur. Therefore, the SESM applications can be started and stopped without affecting a subscriber. • The SSG host key port bundle feature simplifies large deployments because it eliminates manual mapping of subscriber subnets to SSGs.
The following features are provided by SPE and are available only when SESM is deployed in DESS mode.	
Subscriber account self care	This feature allows subscribers to change their own account details, such as address information and passwords. This subscriber updating capability relieves the service provider from time-consuming maintenance tasks. The NWSP sample application illustrates this feature.
Subscriber service subscription	This feature allows subscribers to subscribe to new services and have immediate access to those services. This feature relieves the service provider from time-consuming service enrollment tasks. It also benefits the subscriber because there is no delay in receiving access to a new service. Subscribers can also unsubscribe from a service. The NWSP sample application illustrates this feature.
Subscriber subaccount creation and management	This feature allows a subscriber with a main account to create subaccounts, with different services and access information in each subaccount. For example, a family might have subaccounts for each family member, with a different set of authorized services within each subaccount. The main account can create and delete subaccounts and subscribe to services for the subaccounts. The NWSP sample application illustrates this feature.
Cisco Distributed Administration Tool (CDAT)	CDAT is a web-based application for administrators to use in creating and maintaining the information on users, services, and access policy that is stored in an LDAP directory. The CDAT application is described in the *Cisco Distributed Administration Tool Guide*.
Role based access control (RBAC)	RBAC is an access model that allows administrators to manage groups of subscribers, rather than individuals. Using the RBAC model, administrators define roles, which have specific privileges, and groups, which have assigned roles. Individual subscribers are then assigned to a group and inherit the roles of that group. The Cisco DESS and AUTH APIs implement the RBAC model. See the *Cisco Distributed Administration Tool Guide* for more information about RBAC.

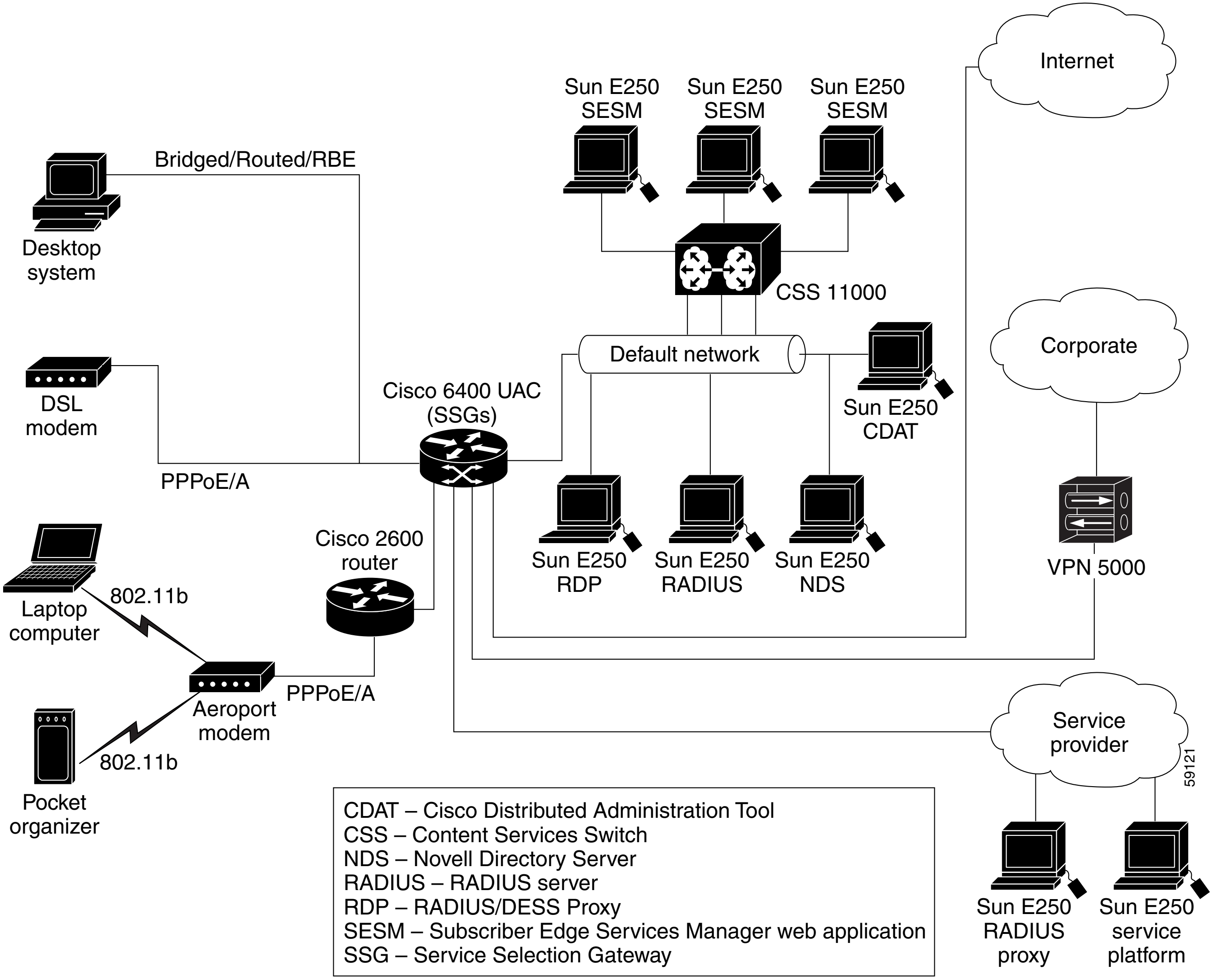
Accounting and Billing Features

The end-to-end solution offered by SESM applications provides support for accounting and billing based on actual services used and the duration of use. Accounting records are produced by a RADIUS server in response to SSG requests. See the SSG documentation for more information.

System Description and Network Diagram

Figure 1-2 shows an SESM deployment in an ISP or NAP communication network.

Figure 1-2 Network Diagram



Regardless of the type of modem or connection layer protocol a subscriber uses, all TCP packets are routed by the SSG when the SSG is enabled. The SSG is a feature in the Cisco IOS running on the node route processors (NRPs) on the Cisco 6400 UAC. Each NRP has an SSG separately enabled. Therefore, a typical production deployment includes multiple SSGs.

Physically, the TCP traffic passes through the SSG on its way to SESM. Logically the HTTP traffic goes directly to an SESM web application running on a default network. The *default network* is an IP address or subnet that TCP packets can access without authentication. The SESM web applications and their associated J2EE web servers run in this default network. The default network is configured on the SSG.

Production deployments might include multiple instances of J2EE web servers and associated SESM web applications on the default network. For production deployments, we recommend using enterprise-class server systems with hot-swappable components and load-balancing across the multiple servers. The Domain Name System (DNS) resolves host names for any of the SESM web applications to the IP address of the load balancer. The Cisco Content Services Switch 11000 (CSS 11000) is preferred for load balancing.

The J2EE web servers receive the HTTP requests for the SESM web application. The SESM web application works with an SSG to establish a session for the user. SESM determines the IP address of the SSG that should handle the session as follows:

- If the host key feature is enabled on the SSG, the SSG's IP address is inserted in the packet. No explicit mapping of a client subnet to an SSG is required.
- If the host key feature is *not* enabled, configuration parameters in the web application's MBean configuration file map client subnets to specific SSGs.

An SESM Release 3.1(1) web application is highly scalable. You can start and stop instances of SESM web applications without affecting subscribers. This is because an SESM application is completely stateless. It does not store any subscriber session information. Rather, the SESM application queries SSG for session state information.

Connection Examples

This section describes how various access methods connect to an SESM web application.

Point-to-Point Protocol Example

This example describes the connection sequence for Point-to-Point Protocol (PPP) access to SESM. For example, consider a DSL subscriber using a PPP client configured on a laptop computer.

1. The subscriber launches the PPP client.
2. The TCP packet travels to the NRP on the Cisco 6400, which has SSG enabled.
3. The SSG on the NRP authenticates the PPP user.
4. The subscriber launches a web browser and sends an HTTP message.
   - If the SSG TCP redirect feature is configured, the subscriber can use any URL in the request, and will be automatically redirected to the SESM web application. If the captive portal feature is also configured, the subscriber could be redirected back to the original URL after being authenticated.
   - If the SSG TCP redirect is not configured, the subscriber must use the URL for the SESM web application.
5. The TCP packet containing the first HTTP request travels through the SSG, to the SSG's default network, to the J2EE web server and the SESM application.
6. If the SESM single sign-on feature for PPP subscribers is enabled, the user is already authenticated and SESM does not request an additional authentication. Rather, SESM queries the SSG for the subscriber's cached profile. A session is established, and SESM returns the subscriber's home page with a list of authorized services.
7. If the SESM single sign-on feature is disabled, or if PPP authentication failed in step 6, SESM returns the SESM logon page. When this request reaches an SESM web application, the application requests authentication services from the SSG. After the subscriber is authenticated, a session is established.

Routed Example

This example describes the connection sequence for a routed access to SESM, which includes the RFC1483 routed access method and the Routed Bridged Encapsulation (RBE) access method.

1. The subscriber launches a web browser and sends an HTTP message.
   - If the SSG TCP redirect feature is configured, the subscriber can use any URL in the request, and will be automatically redirected to the SESM web application. If the captive portal feature is also configured, the subscriber could be redirected back to the original URL after being authenticated.
   - If the SSG TCP redirect feature is not configured, the subscriber must use the URL for the SESM web application.
2. The TCP packet containing the first HTTP request travels through the SSG, to the SSG's default network, to the J2EE web server and the SESM application.
3. SESM returns the SESM logon page.
4. When SESM receives the subscriber's logon information, it requests authentication services from the SSG. After the subscriber is authenticated, a session is established.

SESM in RADIUS Mode

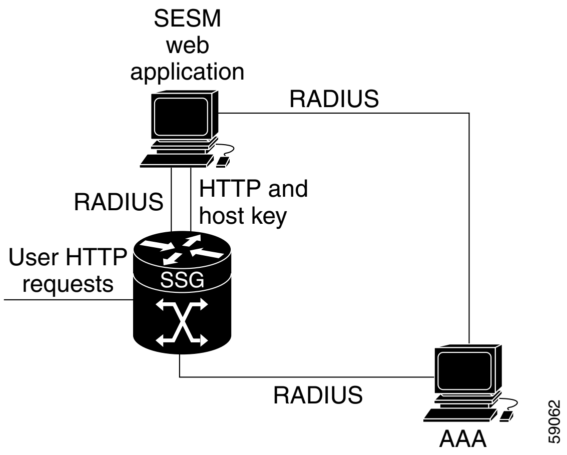
This section describes SESM deployment in RADIUS mode. It includes the following topics:

- Component Diagram for RADIUS Mode, page 1-13
- Processing a Subscriber Request in RADIUS Mode, page 1-14
- Installation and Configuration Requirements for RADIUS Mode, page 1-14

Component Diagram for RADIUS Mode

Figure 1-3 shows a simplified view of SESM deployed in RADIUS mode and the communication mechanisms used between the various software components.

Figure 1-3 SESM Deployed in RADIUS Mode



SSG and the SESM web application work together to process subscriber requests.

- SSG authenticates a subscriber based on a user profile stored in the AAA server.
- The SESM web application obtains the list of authorized services for a subscriber from the user profile in the AAA server.
- After the subscriber selects a service, SSG makes the connection to the service based on information in service profiles stored in the AAA server. In some cases, service preference information might be available in the user profile as well.

Processing a Subscriber Request in RADIUS Mode

Table 1-2 describes the role of SESM applications and SSG in processing typical subscriber actions in a RADIUS deployment.

Table 1-2 Role of Components in the a RADIUS Deployment

Subscriber Action	Software Activity	Components Involved
Subscriber logs on	Authenticate the subscriber in the system.	The SESM application initiates authentication by sending a message to SSG, using the RADIUS protocol. SSG forwards the RADIUS message to the RADIUS server. The RADIUS server authenticates the user and returns a message containing information from the subscriber's user profile. SSG creates an internal host object that represents the subscriber in the current session and forwards the message to SESM.
	Display web interface containing customized content appropriate for the logged on subscriber.	The RADIUS message contains the subscriber's user profile as stored in the RADIUS database. SESM can analyze the user profile and send appropriate content accordingly.
	Display the list of services that the subscriber is currently authorized to access.	The RADIUS message contains the list of services from the subscriber's profile. Authorization is implied for all services in the list. The SESM application obtains a service profile directly from the RADIUS server for each service in the list.
Subscriber selects a service	Access the service.	SESM sends a connection request to SSG. SSG creates a connection object, connecting the host object to the service. When the service is connected, SSG creates a service object. SSG then switches traffic from that subscriber to the requested service.
Subscriber selects a second service	Access a second service, without reauthentication.	The SESM application sends the request to the SSG. SSG creates a second connection object and service object. Both services are concurrently accessed.
Subscriber deselects a service	Stop access to the service.	The SESM application sends the request to the SSG. SSG destroys the appropriate connection object.

Installation and Configuration Requirements for RADIUS Mode

Table 1-3 summarizes the steps required to deploy SESM in RADIUS mode.

Table 1-3 Configuration Requirements for SESM in RADIUS Mode

Activity	Reference
1. Install and configure a RADIUS AAA server.	Appendix D, "Configuring RADIUS" and documentation from the RADIUS server vendor
2. Ensure that all Node Route Processors (NRPs) performing the SSG function are running Cisco IOS Release 12.1(5)DC1 or later.	*Cisco 6400 NRP—Release Notes for Cisco IOS Release 12.1(5)DC* [1]
3. Configure SSG. Use Cisco IOS commands on the SSG host to: – Configure SSG to listen for SESM requests – Enable or disable the host key mechanism – Set up SSG-to-RADIUS communication. – Configure security, routing, and other services provided by SSG.	Appendix B, "Configuring the SSG" *Cisco 6400 Command Reference Guide*[1]
4. Install and configure the SESM, which includes the NWSP sample application and a Jetty web server.	Chapter 3, "Installing Components"
5. Create user and service profiles in the RADIUS database.	Appendix D, "Configuring RADIUS" and documentation from the RADIUS server vendor

[1] See the "Related Documentation" section on page xiii for links to the online versions of these documents.

SESM in DESS Mode

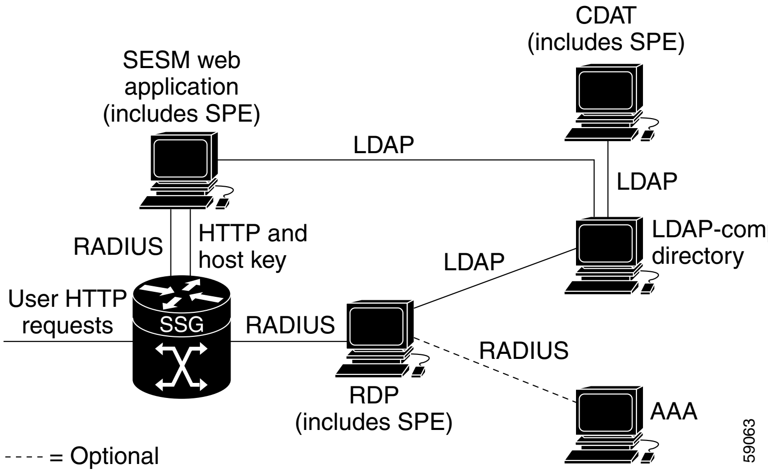
This section describes SESM deployment in DESS mode. It includes the following sections:

- Component Diagram for DESS Mode, page 1-16
- Processing a Subscriber Request in DESS Mode, page 1-16
- Installation and Configuration Requirements for DESS Mode, page 1-17

Component Diagram for DESS Mode

Figure 1-4 shows a simplified view of SESM deployed in DESS mode and the communication mechanisms used between the various software components.

Figure 1-4 SESM Deployed in DESS Mode



The optional AAA server might provide the following services:

- Accounting services
- User authentication services when RDP is configured in Proxy mode

In a DESS mode deployment, the Cisco Subscriber Policy Engine (SPE) Version 1.0 provides services to the SESM web application, CDAT, and RDP. To install SPE services, install the DESS component from the SESM installation package. This guide describes how to install and configure SPE to work with SESM components.

For more information about SPE, including its logical relationship to SESM components, see the "Cisco Subscriber Policy Engine" section on page 1-2.

Processing a Subscriber Request in DESS Mode

Table 1-4 describes the role of SESM applications and SSG in processing typical subscriber actions in a DESS deployment.

Table 1-4 Role of Components in a DESS Deployment

Subscriber Action	Software Activity	Components Involved
Subscriber logs on	Authenticate the user in the system.	The SESM application initiates authentication by sending a RADIUS message to SSG. SSG forwards the RADIUS message to the RDP. The RDP can authenticate using RADIUS or the LDAP directory, depending on how the RDP is configured: • If RDP is configured in proxy mode, it forwards the message to a RADIUS server. • Otherwise, RDP uses the DESS application programming interface (API) to forward the authentication request to the LDAP directory. The response is returned to the SESM application following the same path. SSG creates an internal host object that represents the subscriber in the current session.
	Display appropriate web pages to user.	After the subscriber is authenticated, the SESM application uses the DESS API to retrieve a user profile from the LDAP directory. The SESM application can analyze the profile and display appropriate web pages.
	Display the list of services in the subscriber's profile.	The SESM application uses the DESS API to retrieve service profiles from the LDAP directory for each service in the list.
Subscriber selects a service	Access the service.	SSG sends an authorization request to RDP. Regardless of the RDP mode, RDP always uses the DESS API to send service authorization requests to the LDAP directory. If the service is authorized, SSG creates an internal connection object, connecting the host object to the service. When the service is connected, SSG creates a service object. SSG then switches traffic from that subscriber to the requested service.
Subscriber selects a second service	Access a second service without reauthentication.	SSG sends another authorization request to RDP. Regardless of how it mode, RDP always uses the DESS API to send service authorization requests to the LDAP directory. If the service is authorized, SSG creates a second connection object and service object. Both services are concurrently accessed.
Subscriber updates an e-mail address	Update the LDAP directory.	The SESM application sends the update to the directory using the DESS API.
Subscriber creates a subaccount	Update the LDAP directory.	The SESM application sends the update to the directory using the DESS API.
Subscriber deselects a service	Terminate access to the service.	The SESM application sends the request to the SSG. SSG destroys the appropriate connection object.

Installation and Configuration Requirements for DESS Mode

Table 1-5 summarizes the installation and configuration activities for SESM in DESS mode.

Table 1-5 Configuration Activities Required for SESM in DESS Mode

Activity	Reference
1. (Optional) Install and configure a RADIUS server if: – You want to run RDP in Proxy mode so that it can authenticate subscribers using profiles in a RADIUS server, rather than in the directory. This option allows you to use existing RADIUS user profiles, rather than creating the information on the LDAP directory. (Service authorizations still occur using information in the directory.) – You want to use SSG accounting features.	Appendix D, "Configuring RADIUS" and documentation from the RADIUS server vendor
2. Ensure that all Node Route Processors (NRPs) performing the SSG function are running Cisco IOS Release 12.1(5)DC1 or later.	*Cisco 6400 NRP - Release Notes for Cisco IOS Release 12.1(5)DC* [1]
3. Configure SSG. Use SSG commands on the SSG host to: – Configure SSG to listen for SESM requests. – Set up SSG to RADIUS communication. – Enable the host key mechanism. – Configure security, routing, and other services provided by SSG.	Appendix B, "Configuring the SSG." *Cisco 6400 Command Reference Guide*[1]
4. Install and configure an LDAP directory. **Note** If you are using NDS, you must enable the cleartext password option.	Documentation from the directory vendor
5. Install and configure the SESM software components, which include: the NWSP sample application, Jetty web server, RDP, DESS, and CDAT.	Chapter 3, "Installing Components"
6. Create roles, groups, and user and service profiles in the LDAP directory.	*Cisco Distributed Administration Tool Guide*

[1] See the "Related Documentation" section on page xiii for links to the online versions of SSG documents.

Software Component Descriptions

This section describes the components that you can install from the SESM installation package.

- New World Service Provider, page 1-18
- Jetty Server, page 1-20
- Directory Enabled Service Selection, page 1-20
- RADIUS/DESS Proxy Server, page 1-21
- Cisco Distributed Administration Tool, page 1-21

New World Service Provider

When you install the NWSP component, the installation program prompts you to choose a deployment mode. The installation program then installs and configures software appropriate for that mode. In SESM Release 3.1(1), the modes are Demo mode, RADIUS mode, and DESS mode.

An installation of the NWSP component installs the following items:

- The NWSP sample application

- Captive portal sample application
- Images and JSPs for the NWSP application
- SESM core component class libraries
- API documentation for the SESM libraries
- Configuration and startup files for the NWSP sample application

The New World Service Provider (NWSP) is a sample SESM application. The first step towards developing a customized SESM application is to install and configure the NWSP application in a development environment. You can use this sample application as a starting point for creating a customized and branded SESM application.

See the *Cisco Subscriber Edge Services Manager Web Developer Guide* for information about developing a customized SESM application. Use the configuration information in Chapter 4, "Configuring Components after Installation," to deploy and configure the customized application.

The captive portal sample application demonstrates how several powerful features in this SESM release work together to redirect unauthorized users to an SESM sign-on page immediately after opening a web browser. With this feature, the service provider does not need to provide users with the URL to the SESM sign-on page.

SESM Sample Applications and Demos

This section defines the differences between a sample and a demo application.

SESM Sample Application

An SESM sample application is a fully functioning web application that was built using the SESM development library. It uses the services of the Jetty web server and the JMX management server. Before running the sample application, you need all other solution components installed and configured. For example, you need a fully configured SSG component running on a Cisco 6400 UAC. The RADIUS server (for RADIUS mode) or the LDAP-compliant directory (for the DESS mode) must be installed, configured, running, and populated with user and service information.

Demo Mode

The Demo mode is an SESM application running in a simulated network. The Demo runs without access to other solution components, such as SSG, RADIUS server, or LDAP directory. An SESM application running in standalone Demo mode is *only* intended for demonstration purposes. Demo mode is not in any way representative of Cisco SESM performance in an end-to-end solution with actual network components.

Note If you install the Demo mode, and then later want to perform some development on a customized SESM application, we recommend that you perform another installation. Otherwise, you will need to perform extensive edits to the MBean configuration files.

Demo mode simulates the actions of an SESM deployment in both RADIUS and DESS modes. It uses a local copy of a Merit RADIUS file to obtain profile information. See Chapter 2, "Demo Quick Start," for information about installing and using SESM in Demo mode.

Jetty Server

When you install the jetty component from the SESM installation package, you install the following items:

- Jetty web server—Jetty is a J2EE-compliant server package from Mort Bay Consulting that is released under an open source license. The license puts few restrictions on usage of Jetty. For more information about the Jetty server, see:

  http://jetty.mortbay.com/

- JSP engine—Jetty includes a Java Server Pages (JSP) package, which is currently the Jasper JSP engine from Apache Software Foundation.
- Sun example Java Management Extensions (JMX) server—This is a fully functional JMX server from Sun Microsystems. SESM depends on the JMX server for internal object configuration. For more information about JMX technology and its related JMX MBean standards, see:

  http://java.sun.com/products/JavaManagement

The NWSP application, CDAT, and RDP are installed with configuration files and startup scripts that are ready to run using the Jetty web server and the Sun example JMX server. However, SESM is designed to allow the use of any J2EE web server and any JMX-compliant server. An SESM web application, such as the NWSP sample application, requires an HTTP listener (web server) and a JMX server.

Note For SESM Release 3.1(1), the host key feature works only with a Jetty server.

Directory Enabled Service Selection

When you install the directory-enabled service selection (DESS) component from the SESM installation package, you install the following items:

- Cisco SPE AUTH library—The AUTH library implements a role-based access control (RBAC) authorization model. The RBAC model allows administrators to manage groups of subscribers, rather than individuals. Using the RBAC model, administrators define roles, which have specific privileges, and groups, which have assigned roles. Individual subscribers are then assigned to a group and inherit the roles of that group.
- Cisco SPE DESS library—The DESS library provides the framework for using the RBAC model in an LDAP directory.
- Files containing the directory schema extensions. The install program can optionally apply these extensions to your LDAP directory.
- Files containing sample RBAC data.

See the *Cisco Distributed Administration Tool Guide* for information about the RBAC model, the DESS and AUTH extensions to an LDAP directory, and how to develop subscriber and service profile information in the RBAC model.

RADIUS/DESS Proxy Server

The RADIUS/DESS Proxy (RDP) server is a RADIUS server that can proxy profile requests or use the DESS APIs to query the directory for profiles. RDP acts as the mediator between SSG, which communicates using RADIUS protocol messages, and the LDAP directory schema extensions, which require the DESS API for communication. RDP is a required component in the deployment of SESM in DESS mode.

You can configure the RDP to run in two modes:

- Default mode—In this mode, RDP queries the directory to obtain user authentication and service authorization.
- Proxy mode—In this mode, RDP sends user authentication requests to a specified RADIUS server, rather than to the LDAP directory. This option allows service providers with large RADIUS authentication and accounting services already deployed to continue to use the existing RADIUS database for authenticating users.

  This mode does not affect service authorizations. Regardless of the mode, RDP obtains all service authorizations from information in the LDAP directory.

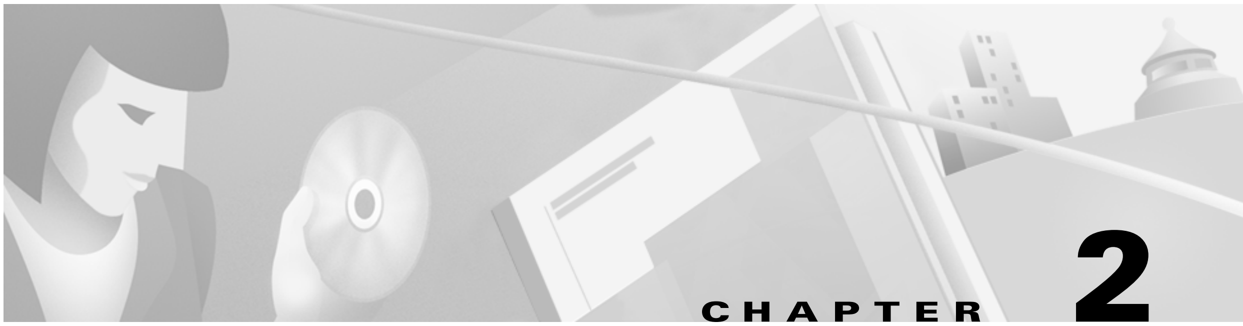
RDP is a Java2 application that uses the services of a JMX server for configuration.

Cisco Distributed Administration Tool

The Cisco Distributed Administration Tool (CDAT) is an administrator's web-based interface for managing data in the DESS and AUTH extensions to the LDAP directory. CDAT provides the means for creating and maintaining users, services, user groups, service groups, roles, and policy rules for the RBAC model.

CDAT, a J2EE application, runs on a J2EE server and uses the services of a JMX server for configuration.

This guide describes how to install and configure CDAT. For information about using CDAT, creating profiles in the RBAC model, and the DESS and AUTH directory extensions, see the *Cisco Distributed Administration Tool Guide*.



Demo Quick Start

This chapter describes procedures for installing and running the Cisco SESM sample applications in Demo mode. The sample applications are the New World Service Provider (NWSP) application and the sample captive portal application. The chapter includes the following topics:

Introduction

The SESM Demo mode has two purposes:

- It lets you demonstrate the capabilities of SESM when other required network components, such as SSG, are not available.
- It is a valuable tool for developers of SESM web applications. Using Demo mode, developers can quickly test the customizations they make to an SESM application. See the *Cisco Subscriber Edge Services Manager Web Developer Guide* for information about using Demo mode during application development.

To prepare for using the Demo mode, you can:

- Install SESM in Demo mode
- Install SESM in DESS or RADIUS mode and switch to Demo mode at run time

The following sections describe the differences between these two approaches.

Installing in Demo Mode

The Demo mode installation is quick and easy and is described in this chapter for your convenience. It requires only a few parameters. In Demo mode, you can demonstrate the features of both RADIUS and DESS deployments.

If you install in Demo mode, you should not expect to switch to DESS or RADIUS modes at run time for the following reasons:

- After installing in Demo mode, the MBean configuration files are not set up properly to support the switch to those other modes. Several manual changes are required in the files.
- The Demo installation might not install all of the components required by the other modes. For example, a Demo installation does not install the DESS component, which is required to run in DESS mode.

Switching to Demo Mode at Run Time

You can install and configure SESM to run in DESS or RADIUS mode, and then easily switch to run the application in Demo mode at run time. The switch from the other modes to Demo mode is supported as follows:

- When you install SESM in DESS or RADIUS mode, the demo data file that supports Demo mode is included in your installation directory.
- The MBean configuration files are set up to point to the demo data file when the application is run in Demo mode.
- The mode attribute in the nwsp.xml file is a Java system property, so that it can be changed at run time.
- The NWSP startup scripts accept a run time argument to change the mode.

To switch to Demo mode at run time, use the following command:

Platform	Command
Solaris	`jetty/bin/startNWSP.sh -mode Demo`
Windows NT	`jetty\bin\startNWSP.cmd Demo`

Installing SESM in Demo Mode

To install SESM in Demo mode, follow these procedures:

Step 1 Log on as a privileged user:

- On Solaris—Run the installation program as root.
- On Windows NT—Run the installation program as a member of the Administrators group.

Make sure you have write privileges to the directory in which you intend to load the demo.

Step 2 Obtain the installation image from the product CD-ROM or from the Cisco web site. The installation image is a tar or zip file, depending on the platform on which you want to install the demo. See the "Obtaining the SESM Installation File and License Number" section on page 3-6 for more information.

Step 3 Uncompress the tar or zip file to a temporary directory. The result includes an executable .bin or .exe file. Table 2-1 shows the names of the compressed and executable files.

Table 2-1 Installation Image File Names

Platform	Compressed File Name	Executable Installation File Name
Solaris	sesm_sol.tar	sesm_sol.bin
Windows NT	sesm_win.zip	sesm_win.exe

Step 4 Execute the installation image as follows:

- On Solaris, change directories to the location of the installation image, and enter the image name. For example:

```
solaris>sesm_sol.bin
```

- On Windows NT, you can double-click the file's icon. Otherwise, open a command prompt window, change directories to the location of the image, and enter the image name. For example:

```
C:\>sesm_sol.exe
```

Step 5 Follow the instructions on the installation windows to install the demo. See Table 2-2 for additional information.

Table 2-2 describes the parameters that you use to install the demo. The Value column provides a place for you to record the values you plan to use during installation.

Table 2-2 Demo Installation and Configuration Parameters

Component	Input Summary	Explanation	Value
General installation parameters	License type	Click the **Evaluation** button. You do not need a license number.	
	License agreement	Read the displayed license agreement to ensure that you agree with the terms of the license. You must accept the agreement to proceed with installation.	

	Installation directory	You can accept the displayed default installation directory, click **Browse** to find a location, or type the directory name in the box. The default installation directories are:
		• On Solaris:
		`/opt/cisco/sesm`
		• On Windows NT:
		`C:\Program Files\cisco\sesm`
		You must have write privileges to the installation directory.	

	Setup type	Click the **Demo** button. The difference between a demo installation and a typical installation is the values that the installation program inserts in the configuration files and the components that are installed. For example, a demo mode installation does not install the DESS component.	
NWSP web application configuration	NWSP port number	Specify the port on which the demo application's web server will listen for HTTP requests. The installation program inserts this value into the application startup script. **Note** The port must be 80 if you plan to choose captive portal mode. The displayed default is port 8080. (Change this to 80 if you choose captive portal mode.) **Note** Each web server running on the same machine must listen on its own unique port. Change this value if another web server, or another instance of the NWSP application, is listening on 8080.	
	Run captive portal	Choose this option to configure the demo application for captive portal mode. In demo mode, you can demonstrate the captive portal feature that captures the original URL supplied by the subscriber. To do this, use the IP address of the host machine as the initial URL. The captive portal feature redirects the request to the captive portal host name.	

Table 2-2 Demo Installation and Configuration Parameters (continued)

Component	Input Summary	Explanation	Value
Note The following section applies only if you choose the captive portal option in the previous section.			
Captive portal configuration (Optional)	Host	Enter the host name or IP address for the host of the captive portal application. This installation program installs the captive portal application on the same machine with the NWSP application.	
	Port	Enter the port number on which the captive portal application will listen. It must be 80.	
	URI	In captive portal mode, SESM sends packets to a welcome page for the captive portal application. Enter the URI of the captive portal application's welcome page. The URI for the sample captive portal application installed with the NWSP sample application is: `/decorate/pages/home.jsp` This value is appended to the host and port entered above to create the URL to which SESM forwards redirected TCP packets. The URI indicates the home page of the captive portal application (that is, the page you want the subscriber to see first). It indicates the directory structure of the application's files within the J2EE container's directory. The URI is location-independent. You can deploy your SESM web application on several host machines, and, although the host and port changes for each host machine, the URI does not change.	

Supported Browsers

You can use the following browsers to demonstrate the NWSP application:

- Netscape Release 4.x and later
- Internet Explorer Release 5.x and later

These browser limitations apply to the NWSP sample application and are mentioned to ensure predictable results during demonstrations. When you develop an SESM application for deployment, you should consider the end users of your deployed application, and design the application to accommodate the media that they commonly use.

Running the SESM Demo

This section includes the following topics:

- Starting the Demo, page 2-6
- Stopping the SESM Demo, page 2-6

Starting the Demo

To start the demo, follow these procedures:

Step 1 Start the NWSP web application in Demo mode.

Platform	SESM Installed Mode	Demo Startup Command
Solaris	Demo mode	`jetty/bin/startNWSP.sh`
	RADIUS or DESS mode	`jetty/bin/startNWSP.sh -mode Demo`
Windows NT	Demo mode	`jetty\bin\startNWSP.cmd`
	RADIUS or DESS mode	`jetty\bin\startNWSP.cmd Demo`

Step 2 Open a web browser.

Step 3 Go to the NWSP URL, which is:

http://*host*:*port*

Where:

host is the IP address or host name of the computer on which you installed the NWSP application. You can enter the value `localhost`, or the IP address 127.0.0.1, to indicate the local computer.

port is the NWSP port number that you specified during the installation.

For example:

```
http://localhost:8080
```

If you selected captive portal mode, the port must be 80, as follows:

```
http://localhost:80
```

Step 4 When the NWSP logon page appears, log on using the following values:

Demo Purpose	User ID	Password
To demonstrate RADIUS mode features	radiususer **Note** Other valid users are user1, user2, and so on, up to user44.	cisco The password is cisco for all of the demo users.
To demonstrate DESS mode features	dessuser1 dessuser2 **Note** The dessuser2 is a subaccount to dessuser1.	cisco cisco

Stopping the SESM Demo

To stop the demo, follow the procedures described in the "Stopping Applications" section on page 5-4.

Demo Data File

This section describes the demo data file. It includes the following topics:

- Demo Data File Name and Location, page 2-7
- File Contents and Format, page 2-7
- Attributes for Demonstrating DESS Features, page 2-8

You might want to examine the demo data file to:

- See the services and features associated with each demo user ID.
- See examples of the vendor specific attributes (VSAs) that SESM and SSG require in a RADIUS database.
- Add new profiles or change existing ones to enhance your demonstration

Demo Data File Name and Location

The subscriber and service profile data that supports the NWSP application running in Demo mode is stored in a RADIUS Merit flat file. The file is located in:

```
nwsp
    config
        demo.txt
```

If you change the name or location of the demo.txt file, you must reflect this change in the demoDataFile attribute in the SSDDemoMode MBean in the nwsp.xml file.

File Contents and Format

The demo.txt file contains example user profiles, service profiles, and service group profiles that support the NWSP application in Demo mode. The format of the file is a Merit RADIUS flat file format, using the RADIUS standard attributes and vendor-specific attributes described in Appendix D, "Configuring RADIUS" and in the "Attributes for Demonstrating DESS Features" section on page 2-8.

The profiles in the demo.txt file can be used as test data for an SESM deployment in RADIUS mode. However, you might want to comment out the following profiles before using demo.txt for testing purposes.

- dessuser1
- dessuser2

These profiles contain attributes that are understood only by an SESM web application running in Demo mode. In RADIUS mode, these attributes are ignored by SSG and RADIUS. Although the NWSP application in RADIUS mode would store the attributes, it would not understand them, and the features that the attributes are intended to demonstrate would not work properly.

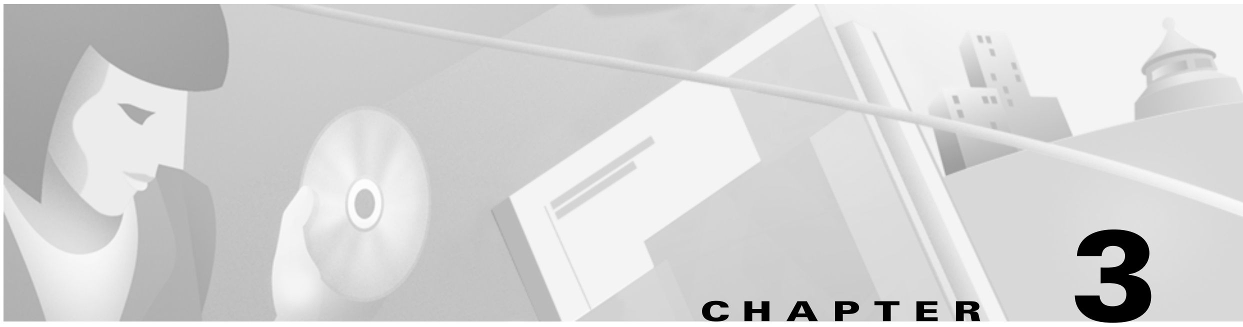
Attributes for Demonstrating DESS Features

Table 2-3 describes the subscriber profile attributes for demonstrating features that are available in DESS mode but not in RADIUS mode. See Appendix D, "Configuring RADIUS" for a description of all other attributes in the demo data file. The attributes in Table 2-3 are for use in Demo mode only. They are not valid VSAs and they should not be added to the RADIUS dictionary. These attributes are not recognized by SSG.

Table 2-3 Attributes for Demonstrating DESS Features

Attribute	Description

Account-Info	Subattributes that can be specified in the demo data file to demonstrate DESS features are:
	• **E***permission*—Sets permissions to perform a task. The value for *permission* must be one of the following strings:
	– Service Selection—The permission to perform service selection is implied and does not have to be explicitly coded in the profile.
	– Self Manage—Use this string to demonstrate the DESS mode feature that allows a subscriber to update their own X.500 user schema information, such as name, address, e-mail, and hobbies.
	– Subaccount Manage—Use this string to demonstrate the DESS mode feature that allows a subscriber to create a subaccount. The Demo mode does not create an actual subaccount; the supporting subaccount profile must be defined in the demo.txt file. Define the subaccount profile and use the **F** attribute.
	– Service Subscription—Use this string to demonstrate the DESS mode feature that allows a subscriber to subscribe to a new service and have immediate access to that service. If you use this string, you must also add a **C** or **L** attribute.
	• **V***name*;*type*;*value*—Use this attribute to specify the initial values that will appear in the fields on the My Account page in the NWSP application running in Demo mode. The Demo allows you to change these values. Use a separate attribute line for each field. The format for each line consists of:
	– *name*—Name of the field on the My Account page in the NWSP application.
	These are X.500 fields. You can add more fields to the demo if you alter the NWSP application to display more fields, as described in the *Cisco Subscriber Edge Services Manager Web Developer Guide*. See the *Cisco Distributed Administrator Tool Guide* for a list of the X.500 names.
	– *type*—Identifies the format of *value*, as follows:
	S or s—Indicates that *value* is a string.
	V or v—Indicates that *value* is a vector of strings in the following format, including the parentheses:
	`{string1;string2;string3}`
	– *value*—Indicates the value to be displayed in the field on the web page display.
	For example:
	`Account-Info = "Vhobbies;V;{science;news;travel}"`
	• **C***serviceName*—Use this attribute to demonstrate the DESS mode self-subscription feature. This feature allows a subscriber to subscribe to a new service and have immediate access to that service. The *serviceName* value must match a service profile name defined elsewhere in the demo flat file (demo.txt).
	• **L***groupName*—Use this attribute to demonstrate the DESS mode self-subscription feature, subscribing to a predefined group of services. The *groupName* value must match a service group profile name defined elsewhere in the demo flat file (demo.txt).
	• **F***parentAccountName*—Use this attribute to indicate that this subscriber profile is a subaccount profile. The *parentAccountName* must match another subscriber profile name defined elsewhere in the demo flat file. (In the installed demo.txt file, dessuser2 is defined as a subaccount to dessuser1.)



CHAPTER 3

Installing Components

This chapter describes how to install the Cisco Subscriber Edge Services Manager (SESM) software and bundled components, including SPE. It includes the following topics:

- Preparing for SESM Installation, page 3-1
- Obtaining the SESM Installation File and License Number, page 3-6
- Performing SESM Installation, page 3-7
- Installation and Configuration Parameters, page 3-10
- Installation Results, page 3-20
- Post-Installation Procedures, page 3-21

Preparing for SESM Installation

This section describes prerequisites to installing SESM. It includes the following topics:

- Installation Platform Requirements, page 3-1
- RAM and Disk Space Requirements, page 3-2
- Java Software Considerations, page 3-2
- SSG and RADIUS Considerations, page 3-4
- LDAP Directory Configuration Requirements, page 3-4

Installation Platform Requirements

This section describes platform requirements for installing the SESM components.

Solaris Platform Requirements

You must have the following hardware and operating system software to install the SESM software on Sun Solaris platforms:

- Sun Ultra10 or Sun E250 (or later version)
- Solaris Version 2.6 (or later version) operating system

Windows NT Platform Requirements

You must have the following hardware and software to install the SESM software on Windows NT platforms:

- Pentium III (or equivalent) processor
- Windows NT Version 4.0, Service Pack 5 (or later version)

RAM and Disk Space Requirements

Table 3-1 shows RAM and disk space requirements for a single instance of each component in SESM. These requirements are approximately the same on the Sun Solaris and the Windows NT platforms.

Table 3-1 RAM and Disk Space Requirements

Component Name	Disk Space (MB)	RAM
Jetty server	1.1	The Jetty server provides the J2EE application environment in which the NWSP and CDAT applications execute. The application memory needs specified for NWSP and CDAT, below, include Jetty server usage.
SESM and the NWSP application	9.1	RAM requirements increase relative to the number of instances running and the specific load. The following numbers are approximations: • In RADIUS mode, the NWSP application requires 17k bytes per subscriber. • In DESS mode, the NWSP application requires more. • In DESS mode, the cache adds to the memory requirements. See the "DESS Attributes" section on page 4-37 for cache size information.
RDP	2.4	32 MB. The RDP memory requirements do not expand based on load. RDP never requires more than 32 MB of RAM.
DESS	1.9	N/A
CDAT	4.9	RAM requirements increase proportionally to the number of objects stored in the directory. For most directory sizes, the 64 MB requirements of the operating system (OS) and other system software should be sufficient for heavily populated directories.

Java Software Considerations

A JRE Version 1.2.2 is bundled in the installation image. The installation process installs this bundled version if it cannot find a suitable version on the installation platform.

This section describes the SESM requirements regarding the Java Runtime Environment (JRE) and the Java Development Kit (JDK). The section includes the following topics:

- Solaris Patch Requirements, page 3-3
- Installing the Bundled JRE, page 3-3
- Specifying an Existing JRE or JDK, page 3-3

- Specifying the JRE or JDK in the Startup Scripts, page 3-3
- Obtaining a JDK for SESM Web Development, page 3-4

Solaris Patch Requirements

On older Solaris platforms, you might need to apply Solaris operating system upgrades (patches). To determine if the machine requires patches, go to the Sun Microsystems Java site and start the process of dowloading the JRE Version 1.2.2. After you log in, you a list of download options appears, including the necessary patches for your operating system version. You should also download the README file, which contains instructions on how to apply the patches.

Installing the Bundled JRE

The installation program determines for itself whether or not to install the bundled JRE Version 1.2.2 by doing the following:

1. It searches for a JDK Version 1.2.2 that is already installed.
2. Failing that, it searches for a JRE Version 1.2.2 or later that is already installed.
3. Failing that, it installs and uses the bundled JRE Version 1.2.2.

To search for an existing JDK or JRE, the installation program looks in the following locations:

- On Windows NT, it looks in the NT Registry for a referenced location.
- On Solaris, it looks in well-known locations. See the "Searching for an Existing JDK or JRE" section on page 7-4 for a list of these locations.

Specifying an Existing JRE or JDK

On Windows NT and Solaris, you can explicitly specify the location of a pre-installed JDK or JRE by starting the installation process on a command line and specifying the javahome parameter, as follows:

```
installImageName -is:javahome location
```

Where:

installImageName is the name of the downloaded SESM image.

location is the path name for the JRE or JDK directory. For example, /usr/java1.2.

Specifying the JRE or JDK in the Startup Scripts

The installation process sets the location of the JDK or JRE in the startup files for NWSP, CDAT, and RDP.

If you change the location of the JDK or JRE after installation, make the corresponding change in the following two startup files:

- Generic startup script—This common script starts the NWSP application, CDAT, and any other customized SESM applications.
- RDP startup script

Table 3-2 shows the path names of the startup scripts that you need to change.

Table 3-2 Startup Script Names

Platform	Generic Startup Script	RDP Startup Script
Solaris	jetty/bin/start.sh	rdp/bin/runrdp.sh
Windows	jetty\bin\start.cmd	rdp\bin\runrdp.cmd

Obtaining a JDK for SESM Web Development

A Java Development Kit (JDK) Version 1.2.2 or later must be installed on any system that will be used by web developers to create or modify the Java Server Pages (JSPs) for a customized SESM application. You can obtain JDK Version 1.2.2 or later from the Sun Java web page:

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

On systems that will be used to customize an SESM application, we recommend that you install the JDK before you install SESM. In that way, the SESM installation program uses the JDK in the application startup scripts, rather than a JRE. The JDK is necessary for recompiling the changed JSPs. See the "Recompiling a Customized JSP" section on page 7-5 for more information.

If you install the JDK after installing SESM, then you must:

- Edit the SESM application start script to use the JDK.
- Ensure that the JDK_HOME environment variable points to the directory into which you installed the JDK.

SSG and RADIUS Considerations

The SESM installation program does not attempt to communicate with SSGs or RADIUS servers. Therefore, SSGs and RADIUS servers do not need to be configured and running for you to install SESM components.

However, you should be prepared to provide correct communication information about those network components during the installation. Otherwise, you must manually edit the configuration files at a later time for the SESM application to work correctly.

The installation program updates configuration files with information that you provide about communicating with SSGs and RADIUS servers. Table 3-5 on page 3-11 describes the configuration information that the installation program prompts you for.

LDAP Directory Configuration Requirements

If you are installing SESM in DESS mode, the installation program establishes communication with your LDAP directory, if possible.

Required Configurations

For communication to occur, perform the following LDAP installation and configuration tasks *before* you run the SESM installation program:

Step 1 Install the LDAP directory.

Step 2 Enable the **Allow Clear Text Passwords** attribute if your LDAP directory is the NDS eDirectory. An SESM deployment in DESS mode does not work on NDS without the cleartext password attribute enabled.

You can enable the cleartext password attribute in NDS by using the freely downloadable ConsoleOne application from Novell.

The clear text password attribute is a property of the LDAP Group object of a server. The LDAP Group object stores the configuration data for a defined LDAP group within the directory tree. The **Allow Clear Text Passwords** attribute allows transmission of bind requests that include passwords over nonencrypted connections. By default, only passwords exchanged over SSL connections are encrypted.

See the NDS documentation for more information about the cleartext password option.

Step 3 Create a container in the LDAP hierarchy for SESM data. The container consists of an LDAP organization and organizational unit. For more information about how SESM data is organized in the LDAP object hierarchy, see the *Cisco Distributed Administration Tool Guide*.

If you intend to load the sample data that comes with CDAT, you might want to name the container to match the contents of the sample data file. Alternatively, you can edit the sample data file before you load it to match the names you use. The sample data uses the following names:

- organization: `cisco`
- organizational unit: `sesm`

Step 4 Create the following administrator accounts. They can be the same accounts, but they do not have to be the same.

- A directory-wide administrator that has permission to extend the directory schema.
- An SESM administrator that has permission to add objects to the SESM container (the organization and organizational unit that you created to hold SESM data).

Advantages to a Running LDAP Directory During SESM Installation

The LDAP directory does not need to be configured and running on the network for you to complete the Cisco SESM installation. However, it is advantageous if the directory is configured and running. If the installation program can communicate with the LDAP directory using the communication parameters that you provide, it can perform the following required tasks:

- Extend the directory schema with DESS extensions. These extensions are the LDAP classes and attributes that will hold the SESM subscriber profiles, service profiles, and policy information.
- Install top-level RBAC objects that are required before administrators can log into CDAT to create additional RBAC objects.

If the installation program does not perform these tasks, you must do them at a later time before running an SESM web application or CDAT.

Dependencies among the SESM Components

You can install all SESM components together on the same machine (a typical installation), or you can install some components separately in a distributed manner (a custom installation). Table 3-3 describes components that must be installed together on the same machine.

Table 3-3 Component Dependencies in a Distributed Installation

SESM Mode	Component Dependencies
RADIUS mode	• NWSP requires a J2EE server (for example, jetty) on the same machine.
DESS mode	• NWSP requires a J2EE server (for example, jetty) and the DESS component on the same machine. • CDAT requires a J2EE server (for example, jetty) and the DESS component on the same machine. • RDP requires the DESS component on the same machine.

Obtaining the SESM Installation File and License Number

The installation images for SESM are available from the product CD-ROM or from the Cisco web site. It includes the following topics:

- Obtaining a License Number, page 3-6
- Downloading from the Cisco Web Site, page 3-7
- Uncompressing the Image, page 3-7

Obtaining a License Number

The SESM installation program installs evaluation and licensed versions of SESM:

- Evaluation—You can install a RADIUS mode evaluation or a DESS mode evaluation. The evaluation options do not require a license number and do not have an expiration period. An evaluation installation provides full software functionality.
- Licensed— You need a license number before deploying SESM in a production environment.

The license number is available on the License Certificate that is shipped with a purchased product. If you have purchased the product and have not yet received the CD-ROM and License Certificate, you can choose the evaluation option during installation. However, be sure to reinstall using your license number when you receive the certificate.

The license number is important when you are requesting technical support for SESM from Cisco. After installation, you can see your license number and the software version in the licensenum.txt file under the installation directory.

Downloading from the Cisco Web Site

If you purchased a contract that allows you to obtain the SESM software from the Cisco web site, follow these procedures:

Step 1 Open a web browser and go to:

http://www.cisco.com

Step 2 Click the **Login** button. Provide your Cisco user ID and password.

To access the Cisco images from the CCO Software Center, you must have a valid Cisco user ID and password. See your Cisco account representative if you need help.

Step 3 Under Service and Support, click **Software Center**.

Step 4 Click **Web Software**.

Step 5 Click **Cisco Subscriber Edge Services Manager**.

Step 6 Download the appropriate image based on the platform you intend to use for hosting the SESM web application.

Uncompressing the Image

Copy and uncompress the tar or zip file to a temporary directory. When you uncompress the file, the results are:

- The installation executable file—A .bin or .exe file, depending on the platform you are using.
- Files used for a silent mode installation—These are .iss and .properties files. See the "Installing Using Silent Mode" section on page 3-9 for information about silent mode.

Table 3-4 shows the names of the compressed and executable files.

Table 3-4 Installation Image File Names

Platform	Compressed File Name	Executable Installation File Name
Solaris	sesm_sol.tar	sesm_sol.bin
Windows NT	sesm_win.zip	sesm_win.exe

Performing SESM Installation

This section describes how to install SESM. It includes the following topics:

- Installation Privileges, page 3-8
- Installation Modes, page 3-8
- Installing Using GUI Mode, page 3-8
- Installing Using Console Mode, page 3-9
- Installing Using Silent Mode, page 3-9

Installation Privileges

You must log on as a privileged user to perform the installation. In addition, you must have write privileges to the directory in which you intend to load the solution components.

The installation program writes to parts of the file system or Windows registry that are only accessible to a privileged user. The outcome of the installation is unpredictable if you are not privileged.

Log on as a privileged user as follows:

- On Solaris—Run the installation program as root.
- On Windows NT—Run the installation program as a member of the Administrators group.

Installation Modes

You can install SESM using the following installation modes:

- GUI mode—An interactive installation method that communicates with you by displaying interactive windows. You use the mouse and the keyboard to provide input during the installation.

  To run the installation in GUI mode, execute the installation image. No special arguments are required.
- Console mode—A text-only, question and answer interactive installation method.

  To run the installation in console mode, use the `-console` argument on the command line when you execute the installation image.
- Silent mode—A text-only noninteractive method. This mode, also known as batch mode, is useful for multiple installs. Before you start the installation process, you prepare files that contain your installation and configuration information. The installation program obtains all input from the response file.

  To run the installation in silent mode, use the `-option` *fileName* argument on the command line when you execute the installation image.

The following sections provide more details about performing an installation in these modes.

Installing Using GUI Mode

GUI mode is the default installation mode. To run in this mode, execute the installation image. No options are required.

- On Solaris, change directories to the location of the installation image, and enter the image name. For example:

```
solaris> sesm_sol.bin
```

- On Windows NT, double-click the installation image file name. Alternatively, open a command prompt window, change directories to the location of the image, and enter the image name. For example:

```
C:\> sesm_win.exe
```

Installing Using Console Mode

To run in console mode, use the `-console` option on the command line.

- On Solaris, change directories to the location of the installation image, and enter the following command:

  ```
  solaris> sesm_sol.bin -console
  ```

- On Windows NT, open a command prompt window, change directories to the location of the image, and enter the following command:

  ```
  C:\> sesm_win.exe -console
  ```

Installing Using Silent Mode

To run in silent mode, you must first prepare the configuration information normally gathered during the installation process in two files:

- InstallShield properties file (.iss file)—This file defines values related to the installation process. It includes the name of the .properties file. This file is specified as an argument on the command line when you start the installation process.
- Java system properties file (.properties file)—This file defines values related to application configuration.

Examples of the .iss and .properties files are included in the installation download. You must modify both files to match your requirements before you start the installation.

To prepare for silent mode:

Step 1 Open the .properties and .iss files in any text editor.

Note Before you begin, you might need to obtain write access to the files.

Step 2 Edit the values for each parameter in the file. Table 3-5 on page 3-11 describes each parameter. Save and close the file.

Step 3 To turn on the installation logging feature for a silent mode installation, open the .iss file in any text editor. Remove the first pound sign (#) from the following line:

# -log # @all

Step 4 Save and close the file.

To run in silent mode, use the `-options` option on the command line, as follows:

```
imageName -options issFileName
```

Where:

imageName is the name of the downloaded installation image.

issFileName is the name of the install shield properties file you prepared.

For example:

- On Solaris, change directories to the location of the installation image, and enter the following command:

  ```
  solaris> sesm_sol.bin -options mysesm.iss
  ```

- On Windows NT, open a command prompt window, change directories to the location of the image, and enter the following command:

  ```
  C:\> sesm_win.exe -options mysesm.iss
  ```

Installation and Configuration Parameters

Table 3-5 describes the installation and configuration parameters that you enter during the installation process. You can use the Value column in the table to record your planned input values.

You can change the value of any configuration parameter later by editing configuration files, as described in Chapter 4. You cannot change the values of the general installation parameters identified in the first part of the table.

Table 3-5 SESM Installation and Configuration Parameters

Category	Input Summary	Explanation	Value
General installation parameters	Installation type and license number	Choose the type of installation: • RADIUS Evaluation—Choose this option to evaluate SESM in a RADIUS deployment. You do not need a license number and there is no expiration time associated with the evaluation. • DESS Evaluation—Choose this option to evaluate SESM in a DESS deployment. You do not need a license number and there is no expiration time associated with the evaluation. • Licensed—If you purchased an SESM license, choose this option and enter the license number provided by Cisco. **Note** Obtain your SESM license number from the License Certificate shipped with the CD-ROM or otherwise provided to you by your Cisco account representative. The installation program interprets the license number you enter and proceeds to install either RADIUS or DESS mode components, whichever matches the license you purchased. A RADIUS mode license will not allow you to install the DESS mode components. The licensenum.txt file in your root installation directory records your license number and the software version number you installed. This information is important when you access Cisco technical support for this product.	
	License agreement	Read the displayed license agreement to ensure that you agree with the terms of the license. You must accept the agreement to proceed with installation.	

	Installation directory	**Note** You must have write privileges to the installation directory.
		To specify the installation directory, you can do any of the following:
		• Accept the displayed default installation directory
		• Click **Browse** to find a location
		• Type the directory name in the box.
		The default installation directories are:
		• On Solaris:
		`/opt/cisco/sesm`
		• On Windows NT:
		`C:\Program Files\cisco\sesm`	

Table 3-5 SESM Installation and Configuration Parameters (continued)

Category	Input Summary	Explanation	Value
General installation parameters *(continued)*	Setup type	Select one of the following: • **Typical**—Installs and configures all components on the same workstation. If you are installing a RADIUS mode deployment, a typical installation includes the following components: – NWSP—Includes the SESM core model. – Jetty If you are installing a DESS mode deployment, a typical installation includes the following components: – NWSP—Includes the SESM core model. – Jetty – DESS – RDP – CDAT See the "Software Component Descriptions" section on page 1-18 for a description of what you are installing with each of these components. • **Custom**—Allows you to choose the components to install and configure from a checklist. Choose this option to: – Install the NWSP application without the Jetty server (because you want to use a different J2EE server) – Reinstall one of the components – Distribute the solution components among different workstations. See the "Dependencies among the SESM Components" section on page 3-6 for a list of components that must be installed on the same workstation. • **Demo**—Installs and configures the NWSP application in DEMO mode. Use this option to demonstrate the capabilities of SESM when other network components, such as SSG, are not available. The difference between a demo installation and a typical installation is the contents of the configuration files. In addition, a demo mode installation does not install the DESS component.	

Table 3-5 SESM Installation and Configuration Parameters (continued)

Category	Input Summary	Explanation	Value

Web server configuration	NWSP port number	Specify the port on which the NWSP application's web server will listen for HTTP requests from subscribers. The installation program updates the application startup script to use this value.
		The displayed default is port 8080.
		Tips Change this value to 80 if you plan to use captive portal mode.
		Tips Each web server running on the same machine must listen on its own unique port. If another web server or another instance of the NWSP application is listening on 8080, change this value.
		The application startup script uses the application port number to derive two other port numbers:
		• A secure socket listener (SSL) port is derived as follows:
		`application port - 80 + 443`
		When the application port is 8080, the SSL port is:
		`8080 - 80 + 443 = 8443`
		• A management console port is derived as follows:
		`application port + 100`
		When the application port is 8080, the management port is:
		`8080 + 100 = 8180`	

	Run captive portal	Choose this option to configure the captive portal application. **Note** If you do not choose this option, the installation program installs the captive portal application but does not configure it. The captive portal application runs on the same web server with the NWSP application. It captures the original URL that was requested by the subscriber and forwards it to the SESM web application along with the redirect. The SESM web application can then honor the subscriber's originally requested URL after authentication occurs.	

Table 3-5 SESM Installation and Configuration Parameters (continued)

Category	Input Summary	Explanation	Value
Note The following section applies only if you choose the captive portal option in the previous section.			
Captive portal configuration	Host	Enter the host name or IP address for the host of the captive portal application. This installation program installs the captive portal application on the same machine with the NWSP application.	
	Port	Enter 80, the port number on which the captive portal application's web server will listen. This installation program configures the captive portal application to run in the same J2EE container with the NWSP application. Therefore, the port number must match the port number used for the NWSP port.	

	URI	Enter the URI of the SESM web application's home page (that is, the page you want the subscriber to see first). The URI is appended to the NWSP host and port entered previously to create the URL to which the captive portal application redirects the subscriber's browser.
		For example, the URI for the NWSP application is:
		`/decorate/pages/home.jsp`
		The leading slash is required.
		Continuing the example, if the NWSP host name is myhost and the NWSP port is 80, the captive portal application would redirect an unauthenticated subscriber to the following URL:
		`myhost:80/decorate/pages/home.jsp`
		The URI indicates the directory structure of the NWSP application's files within the J2EE container's directory. The URI is location-independent. You can deploy your SESM web application on many host machines, and, although the host and port would change for each host machine, the URI would not change.	

Note If you are installing SESM in Demo mode, you are finished with the installation.			

Table 3-5 SESM Installation and Configuration Parameters (continued)

Category	Input Summary	Explanation	Value
SESM to SSG communication **Tips** You can use a **show run** command on the SSG host to determine how SSG is configured.	SSG port number	Specify the port that SSG uses to listen for RADIUS requests from an SESM application. This value must match the value that was configured on the SSG host with the following command: `ssg radius-helper` *`authenticationPort`* The default value is 1812.	
	SSG shared secret	Specify the shared secret used for communication between SSG and an SESM application. This value must match the value that was configured on the SSG host with the following command: `ssg radius-helper key` *`secret`* The default value is `cisco`.	
	SSG port bundle size	Enter the number of bits that SSG uses for port bundling when the host key feature is enabled. This value must match the value that was configured on the SSG host with the following command: `ssg port-map` *`length`* The value must be 0 or 4. A value of 0 indicates that the SSG is not using the host key and port bundle mechanism. **Note** The host key feature is introduced in Cisco IOS Release 12.2(2)B. If you are using an earlier release, use a value of 0 in this field. The default value is 0.	

When the port bundle size is 0, you must map SSGs to client subnets. The following category of parameters lets you map one client subnet for one SSG. You must manually edit the configuration file to:

- Map additional non-host key SSGs,
- Add more client subnets to this SSG, or
- Override the global values you specified in the previous category.

See the "Associating SSGs and Subscriber Requests" section on page 4-27 for more information.

Category	Input Summary	Explanation	Value
One non-host key SSG	SSG address	Enter the host name or IP address of the SSG host.	
	Client subnet	Enter one client subnet address handled by this SSG. For example, 177.52.0.0.	
	Subnet mask	Enter the mask that can be applied to subscriber IP addresses to derive their subnet. For example, 255.255.0.0.	

Note If you are installing SESM in DESS mode, skip the following two categories and continue with the "Directory server information" category.

Table 3-5 SESM Installation and Configuration Parameters (continued)

Category	Input Summary	Explanation	Value
SESM to RADIUS server communication	Primary AAA server IP	Enter the IP address or the host name of the primary RADIUS server.	
	Primary AAA server port	Enter the port number on the primary RADIUS server host that the RADIUS server listens on. The default is 1812.	
	Secondary AAA server IP	Enter the IP address or the host name of the secondary RADIUS server. If you are not using a secondary RADIUS server, enter the same value used for the primary server.	
	Secondary AAA server port	Enter the port number on the secondary RADIUS server host that the RADIUS server listens on. If you are not using a secondary RADIUS server, enter the same value used for the primary server.	
	Shared secret	Enter the shared secret used between the RADIUS server and SESM. If you are using a primary and a secondary server, the shared secret must be the same for both servers. The default value is `cisco`.	
Service Password	RADIUS service password	Enter the password that the SESM application uses to request service and group profiles from RADIUS. This password must match the value that was configured on the SSG host with the following command: `ssg service-password` *`password`* The service-password value must be the same on all of your SSGs. The default value is `servicecisco`.	
Note If you are installing SESM in RADIUS mode, you are finished with the installation.			
Directory server information	Directory address	Enter the IP address or the host name of the system where the directory server is running.	
	Directory port	Enter the port on which the directory server listens.	
	Directory admin user	Enter a user ID that has permissions to extend the directory schema. The default value is cn=admin, ou=sesm, o=cisco.	
	Directory admin password	Enter the password for the directory administrator.	
Note The installation program attempts to access the directory server, using the information you just provided. If access is unsuccessful, the installation program displays a window with the header "Warning—Please confirm these options." You should verify the information you entered and also verify that the directory server is currently running. If the directory is not running, you can continue the installation of DESS components by clicking the **Ignore** button on the warning window. However, if you click **Ignore**, the installation program can not update the directory for SESM use. You must perform the updates at some later time before running SESM web applications or CDAT. See the "Extending the Directory Schema and Installing Initial RBAC Objects" section on page 4-40 for instructions.			

Table 3-5 SESM Installation and Configuration Parameters (continued)

<table>
<tr><th>Category</th><th>Input Summary</th><th>Explanation</th><th>Value</th></tr>
<tr><td>Directory container information</td><td>Directory container</td><td>Enter the organization and organizational unit that will hold the SESM service, subscriber, and policy information. Use the following format:
ou=<i>orgUnit</i>,o=<i>org</i>
For example, the installation program's default values are:
ou=sesm,o=cisco
The above defaults are the values used in the sample data file that comes with CDAT.</td><td></td></tr>
<tr><td></td><td>Directory user ID</td><td>Enter a user ID that has permissions to access and create objects in the organization and organizational unit named above. Use the following format:
cn=<i>userID</i>,ou=<i>orgUnit</i>,o=<i>org</i>
For example, the default values are:
cn=admin,ou=sesm,o=cisco</td><td></td></tr>
<tr><td></td><td>Directory password</td><td>Enter the password associated with the directory user ID.</td><td></td></tr>
<tr><td colspan="4"><b>Note</b> The installation program attempts to access the container using the information you just provided. If it is unsuccessful, a warning message appears, as described in the previous note.</td></tr>
<tr><td>CDAT</td><td>CDAT port number</td><td>Enter the port number on which the CDAT web server will listen.
The default is 8081.</td><td></td></tr>
</table>

Table 3-5 SESM Installation and Configuration Parameters (continued)

Category	Input Summary	Explanation	Value
RDP Configures RDP to SSG communication	IP address	Enter the IP address or host name of the RDP. ⚠ **Caution** This value must be a real IP address to which the NRP can route. You cannot use the values localhost or 127.0.0.1.	
	Port number	Enter the port on which the RDP will listen. The default is 1812.	
	Shared secret	Enter the shared secret to be used for communication between SSG and RDP. It must be a different value from the shared secret used for RDP to RADIUS communication. The default is `cisco`.	
	Service password	Enter the password that RDP uses to request service profiles from the directory. This password must match the value that was configured on the SSG host with the following command: `ssg service-password` *`password`* The service-password value must be the same on all of your SSGs. The default value is `servicecisco`.	
	Next hop password	Enter the password that SSG uses to request next hop tables from RDP. This password must match the value that was configured on the SSG host with the following command: `ssg next-hop download` *`nextHopTableName password`* The service-password value must be the same on all of your SSGs. The default is `nexthopcisco`.	
	Proxy mode	Choose this option to run RDP in proxy mode. RDP has two modes: • Proxy mode—In this mode, RDP forwards authentication requests to a RADIUS server. RDP uses the DESS API to send authorization requests to the directory. • Non-proxy mode—In this mode, RDP performs authentication based on information it obtains from the directory. RDP uses the DESS API to send authorization requests to the directory.	

Table 3-5 SESM Installation and Configuration Parameters (continued)

Category	Input Summary	Explanation	Value
RDP *(continued)*	Add services	Choose this option if you want SSG to perform automatic connections to services when a subscriber's profile includes the autoconnect attribute. When you choose this option, RDP includes the subscriber's service list and related information in replies to SSG. This service information consumes memory on the SSG host—the node route processor (NRP). Do not choose this option if space is a consideration on the NRPs. Instead, you can configure the SESM application to initiate automatic connections. See the "autoConnect" section on page 4-22 for more information.	
If you choose Proxy mode for RDP, then the installation process prompts you for the following RADIUS server information.			
RDP to RADIUS communication	Primary AAA server IP	Enter the IP address or the host name of the primary RADIUS AAA server that you want RDP to communicate with.	
	Primary AAA server port	Enter the port number on the primary RADIUS server host that the RADIUS server listens on.	
	Secondary AAA server IP	Enter the IP address or the host name of the secondary RADIUS server. If you are not using a secondary RADIUS server, enter the same value used for the primary server.	
	Secondary AAA server port	Enter the port number on the secondary RADIUS server host that the RADIUS server listens on. If you are not using a secondary RADIUS server, enter the same value used for the primary server.	
	Shared secret	Enter the shared secret used between RDP and the RADIUS server. The shared secret must be the same for both servers. The default is `cisco`.	

Table 3-5 SESM Installation and Configuration Parameters (continued)

Category	Input Summary	Explanation	Value
The installation program installs the components on your system. When it is finished installing the files, it displays an additional window about modifications to the LDAP directory.			
LDAP directory modifications	Extend schema	Choose this option if you want the installation program to apply the DESS schema extensions to the LDAP directory. These extensions include the dess and auth classes and attributes. For more information about the extensions, see the *Cisco Distributed Administration Tool Guide*. If you do not choose this option, you must extend the directory schema later, before running the SESM application in DESS mode and before logging into CDAT to create objects in the directory. See "Extending the Directory Schema and Installing Initial RBAC Objects" section on page 4-40 for more information. **Note** If you are installing DESS in multiple locations, you only need to extend the schema one time.	
	Install RBAC	Choose this option if you want the installation program to load the top-level RBAC objects. If you do not choose this option, you must install RBAC objects later, before running an SESM application in DESS mode and before logging into CDAT to create objects in the directory. See "Extending the Directory Schema and Installing Initial RBAC Objects" section on page 4-40 for more information. **Note** If you are installing DESS in multiple locations, you only need to extend the schema one time.	

Installation Results

The Cisco SESM installation directory contains the following subdirectories and files:

- _uninst—This subdirectory contains the utility to uninstall the components you just installed. To uninstall, run the executable file in this directory.
- captiveportal—This directory contains the captive portal web application to execute on the Jetty server.
- jetty—This directory contains the following subdirectories:
  - bin—Contains start scripts for Jetty server applications
  - config—Contains configuration files that control Jetty servlets
  - lib—Contains the Jetty server class libraries.
- lib—This directory contains the SESM libraries and the docs subdirectory, which contains the Java application documentation.
- licensenum.txt—This file contains the license number that you used during installation and the version number of the SESM software that you installed.

- nwsp—This directory contains the following subdirectories:
  - config—Contains configuration files for the NWSP application.
  - docroot—Contains the Web application, including libraries, JSPs, images, and a J2EE configuration file.
- redist—This directory contains libraries from other companies that Cisco is redistributing. It includes the Jasper JSP framework, the JMX framework, and the JAXP XML parser framework. It also includes test tools.

When you install SESM in DESS mode, the installation directory contains the following additional directories:

- rdp—This directory contains startup scripts, configuration files, and libraries for the RADIUS/DESS Proxy Server.
- cdat—This directory contains configuration files and libraries for CDAT.
- dess-auth—This directory contains the DESS and AUTH libraries, DESS schema, and sample data.

Post-Installation Procedures

This section outlines the steps to take after you successfully complete an installation.

Step 1 Perform all configuration activities listed in Table 1-3 on page 1-15 (RADIUS mode) or Table 1-5 on page 1-18 (DESS mode).

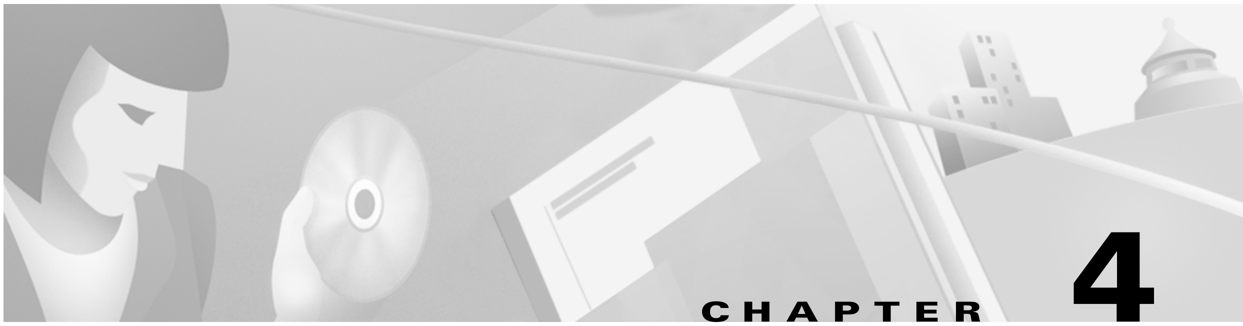
Step 2 Add configuration information for additional SSGs, if the host key feature is not used.

The SESM installation program caters to use of a single SSG or multiple SSGs with the host key feature. For multiple SSG support without the host key feature, you must configure the SSG to client subnet mapping. See the "Associating SSGs and Subscriber Requests" section on page 4-27 for instructions.

Step 3 Start the NWSP web application with the startNWSP script in the jetty bin directory. See Chapter 5, "Running SESM Components" for information about this script.

Step 4 Start a web browser. See the "Supported Browsers" section on page 6-1. Access the NWSP application as described in the "Accessing the NWSP Application" section on page 6-1.

See the "Customizing the NWSP Application" section on page 6-2 for information about customizing the NWSP application.



CHAPTER 4

Configuring Components after Installation

This chapter describes all of the configurable attributes in the Subscriber Edge Services Manager (SESM) software components. Use this chapter to change or fine-tune attributes after installation.

This chapter includes the following topics:

Configuration Overview

This section provides an overview of the configuration files and the configuration technology used by SESM. It includes the following topics:

Changing Configuration Information

You can change any configuration information by manually editing the configuration files. If you change configuration information, you must stop and restart the SESM web application and the Jetty server. If you deployed SESM in DESS mode, you also must stop and restart RDP. See Chapter 5, “Running SESM Components,” for instructions.

Configuration Technology

SESM configuration is based on the Java Management Extensions (JMX) specification and its related JMX MBean standards. For descriptions of these standards, go to:

```
http://java.sun.com/products/JavaManagement
```

The configuration elements involved in SESM are:

- MBeans—MBeans are Java classes that follow a model described in the MBean standards. An MBean represents the management interface for a resource. The management interface is the set of all necessary information and controls that a management application needs to operate on the resource.

  SESM uses MBeans to configure components and the communications connections between those components. For example, an SESM MBean configures the SESM mode; an SSG MBean configures communication between SSG and the SESM web application, an AAA MBean configures communication between RADIUS servers and the SESM web application, and so on. Container-specific parameters are also defined as MBeans. For example, Cisco created a logging MBean for the Jetty server.
- JMX server—The JMX server, sometimes known as the MBean server, is a registry for objects which are exposed to management operations by an agent. Any object that is registered with the JMX server becomes visible to the agent. (For SESM, the agent is the Cisco ConfigAgent.) MBeans are registered by the ConfigAgent or by other MBeans.

  The Jetty component in the SESM installation package includes a JMX server. You can substitute any JMX-compliant server.
- Cisco ConfigAgent—The Cisco ConfigAgent is a JMX-compliant agent provided by Cisco. ConfigAgent configures MBeans by reading and implementing values from MBean configuration files. ConfigAgent is an MBean, started by the SESM web application.
- MBean Configuration Files—The MBean configuration files are XML files in a format defined in xmlconfig.dtd, a Cisco DTD. These files set configurable attributes in SESM. The SESM installation program assigns values for all of the key attributes in these files, using a combination of default values and values you provide during the install. You can change the value of any attribute by editing the appropriate MBean configuration file.

Cisco ConfigAgent

Cisco ConfigAgent performs the following management functions for MBeans.

- Constructs and initializes an MBean—The <Instantiate> tag causes ConfigAgent to construct and initialize an MBean. Most MBeans are initialized by other objects (for example, other MBeans) and not by ConfigAgent.

  After initialization, an MBean registers itself with the JMX server.
- Configures an MBean—The <Configure> tag causes ConfigAgent to configure an MBean.

  When the ConfigAgent detects a newly registered MBean, ConfigAgent configures that MBean if there is a matching entry in the XML files for that MBean.

  The <Set> tag sets attribute values for the MBean.
- Starts an MBean—The <Call> tag causes ConfigAgent to start an MBean.

The contents of the MBean configuration files control ConfigAgent activity.

Configuration Files

Two types of configuration files are used in SESM:

- J2EE configuration files—These are standard J2EE files that conform to Java servlet specifications. Examples are web.xml and webdefaults.xml.
- MBean configuration files—These XML files conform to a format defined by Cisco. These files are named *application*.xml.

J2EE Configuration Files

The J2EE configuration files, such as web. xml and webdefaults.xml, define how the applications run in the J2EE environment. These files conform to Java specifications, as described in the Java Servlet Version 2.3 specifications from Sun Microsystems.

Administrators do not usually need to change the J2EE configuration files. Therefore, the contents of these files are not documented in this guide. However, web developers might require changes to these files. The *Cisco Subscriber Edge Services Manager Web Developer Guide* describes application-specific parameters in the J2EE configuration files. For information about other parameters, see the Java Servlet Version 2.3 specifications. To download these specifications, go to:

http://java.sun.com/aboutJava/communityprocess/first/jsr053

Table 4-1 lists the J2EE configuration files used to configure SESM web applications. The table includes a brief description of each file and shows the installed location of the file.

Table 4-1 Summary of J2EE Configuration Files

Component	File Path and Name	Description

Container (Jetty)	`jetty`
	`config`
	`webdefault.xml`
		This file sets attributes for the Jetty server's handling of HTTP requests and how they map to servlets and JSPs.

SESM web application (NWSP)	`nwsp`
	`docroot`
	`WEB-INF`
	`web.xml`
		This file defines J2EE application parameters, including parameters related to Java Server Pages (JSPs).

SESM captive portal application	`captiveportal`
	`docroot`
	`WEB-INF`
	`web.xml`
		This file defines J2EE application parameters for the captive portal application.

CDAT	`cdat`
	`docroot`
	`WEB-INF`
	`web.xml`
		This file defines J2EE application parameters for CDAT.

MBean Configuration Files

Administrators edit the MBean configuration files to change values of configurable attributes for SESM software components. The installation program assigns initial values for all of the key attributes in these files, using a combination of default values and values you provide during the install. You can change the value of any attribute by editing the appropriate MBean configuration file.

The MBean configuration files conform to xmlconfig.dtd, a Cisco DTD. See the "MBean Configuration File Format" section on page 4-5 for a summary of the MBean configuration file format. See Appendix C, "DTD for MBean Configuration Files" for the complete DTD.

Each software component in an SESM deployment has its own MBean configuration files. Table 4-2 lists all of the MBean configuration files used in an SESM deployment. The table describes the file location relative to the installation directory and a brief description of the file.

Table 4-2 Summary of MBean Configuration Files

Component	File Path and Name	Description

Container (Jetty)	`jetty`
	`config`
	`nwsp.jetty.xml`
	`cdat.jetty.xml`
	`yourapp``.jetty.xml`
		You can configure the Jetty server instance associated with each application differently. These files configure:
		• Logging and debugging for the Jetty server. This log file name is *nnn*.jetty.log.
		• HTTP listener, which configures:
		– The application that is running in the container and the application port.
		– The web server's standard HTTP request log. This log file name is *nnn*.request.log.

SESM web application (NWSP)	`nwsp`
	`config`
	`nwsp.xml`
		This file configures:
		• SESM deployment options
		• Communication between an SESM web application and SSG
		• Communication between an SESM web application and RADIUS servers
		• Attributes for a captive portal application
		• Logging and debugging for the SESM web application. The log file name is *nnn*.application.log.
		• A management port for development and testing purposes

SESM captive portal application	—	Captive portal attributes are included in the MBean configuration file for the SESM web application.

RDP	`rdp`
	`config`
	`rdp.xml`
		This file configures:
		• RDP options and packet handlers
		• RDP communication with SSG
		• Optionally, RDP communication with a RADIUS server
		• Logging and debugging for RDP
		• A management port for development and testing purposes.

Table 4-2 Summary of MBean Configuration Files (continued)

Component	File Path and Name	Description

CDAT	`cdat`
	`    config`
	`        cdat.xml`
		This file configures:
		• System resource usage for the CDAT application
		• Logging and debugging for the CDAT application
		• A management port for development and testing purposes.

DESS	`dess-auth`
	`    config`
	`        config.xml`
		This file configures attributes used by the executing classes in the Dess and Auth application programming interfaces (APIs). The Dess and Auth APIs provide the underlying support for communication between an LDAP directory and the RDP, CDAT, and SESM web applications. If these applications are installed on the same machine, the same config.xml file applies to all of the applications.
		This file contains attributes that control:
		• Directory security
		• Directory connections
		• Caching
		• Logging

For detailed descriptions of all attributes in the MBean configuration files, see the following tables:

- Table 4-3 on page 4-10, “Attributes in the Container MBean Configuration Files”
- Table 4-4 on page 4-18, “Attributes in the Application MBean Configuration File”
- Table 4-6 on page 4-33, “Attributes in the RDP MBean Configuration File”
- Table 4-7 on page 4-36, “Attributes in the CDAT MBean Configuration File”
- Table 4-8 on page 4-38, “Attributes in the Dess-Auth MBean Configuration File”

MBean Configuration File Format

This section summarizes the MBean file format defined in xmlconfig.dtd. The purpose of this summary is to provide enough information for you to easily edit the MBean files. For the full text of the DTD, including extensive comments, see Appendix C, “DTD for MBean Configuration Files.”

Use the following example as a reference while reading the format guidelines that follow. The example configures the debugging MBean for an SESM application.

```
<Instantiate order="1"
    class="com.cisco.aggbu.jmx.LoggerMBean"
    jmxname="com.cisco.aggbu:name=Logger"/>


</Instantiate>
```

```
<!-- ================================================================ -->
<Configure jmxname="com.cisco.aggbu:name=Logger">
    <Set name="debug" type="boolean"><SystemProperty name="nwsp.debug"
        default="false"/></Set>
    <Set name="debugPatterns"></Set>
    <Set name="debugThreads"></Set>
    <Set name="debugVerbosity">LOW</Set>
    <Set name="logDateFormat">yyyyMMdd:HHmmss.SSS</Set>
    <Set name="logFile"><SystemProperty name="application.log"
        default="./logs"/>/yyyy_mm_dd.application.log</Set>
    <Set name="logFrame"  type="boolean">false</Set>
    <Set name="logStack"  type="boolean">false</Set>
    <Set name="logThread" type="boolean">true</Set>
    <Set name="logToErr"  type="boolean"><SystemProperty name="nwsp.logToErr"
        default="false"/></Set>
    <Set name="trace"     type="boolean">true</Set>
    <Set name="warning"   type="boolean">true</Set>
</Configure>
```

The following guidelines explain the basic format of the MBean configuration files.

- The MBean configuration file contains a single <XmlConfig> element containing one or more <Configure> elements.
- Each <Configure> element describes the configuration for either:
  - A single MBean, identified with the name attribute
  - A class of MBeans, identified with the class attribute

  Each <Configure> element must contain one of the above attributes, or both. ConfigAgent matches a registered MBean by both class and name, so that two <Configure> elements might be applied to an MBean.

  The <Instantiate order = *x*> tag causes the ConfigAgent to construct and initialize the named MBean or class of MBeans.

  The value assigned to the order attribute controls the order in which objects are initialized by the ConfigAgent. The lowest value is initialized first and the highest value is initialized last. For example, in the nwsp.xml file, the logger MBean uses the value 1, to ensure that it is initialized first.

  After being initialized, an MBean registers itself with the MBean server. When ConfigAgent detects the newly registered object, it then configures the object.
- The <Set> tag identifies an MBean attribute. The format for the <Set> tag is:

  ```
  <set name="parmName" [type="dataType"]>value</set>
  ```

  Where:

  parmName is the MBean parameter name whose value is being set. Do not change any *parmName*.

  dataType is the required data type of the value you specify. If *dataType* is not explicitly identified, the default *dataType* is string. It is best not to change any *dataType*.

  value is the parameter value. You can edit the value, making sure that the value you provide conforms to the data type specified.
- The <Call> tag calls a method defined within the class or the object’s class. If the method expects arguments, they are specified within the call tag as well.

- The <Arg> tag inside a call tag can be set to any of the following:
  - Literal values.
  - Objects that are created by a New element or returned by a Call element. Call and New elements might contain Set, Put, and Call elements after any Arg elements. These nested elements are applied to the created or returned object.
- A <SystemProperty> tag might appear inside a <Set> or <Call> tag. See the next section ("Java System Properties in the MBean Configuration Files") for more information.

Java System Properties in the MBean Configuration Files

The MBean configuration files use Java system properties as the value for some attributes. The value of a Java system property can be set at application run time. If a value is not specified at run time, the application uses a default value specified in the MBean configuration file.

Note The installed start scripts (START.sh or START.cmd) set some of the system properties. For those system properties, the default values in the MBean configuration files are not used, unless you delete the setting in the startup script.

In the MBean configuration files, the <SystemProperty> tag might appear inside a <Set> or <Call> tag. The format is:

```
<SystemProperty name="propertyName" default="value"/>
```

Where:

propertyName is the Java system property name whose value sets the configurable attribute.

value is the default value used if no value is assigned at run time.

The following lines from the installed start script set system properties. (The -D argument to the JAVA command defines the value of a system property.) For a description of how the start script derives values for the environment variables used in the assignments, see Table 5-1 on page 5-4.

```
$JAVA -Xmx20m \
  -classpath $CLASSPATH \
  -Djetty.home=$JETTYDIR \
  -Dapplication.home=$APPDIR \
  -Dapplication.log=$LOGDIR \
  -Dapplication.portno=$PORTNO \
  -Dmanagement.portno=$MGMTPORTNO \
```

Configuring the J2EE Jetty Container

This section includes the following topics:

- Containers and Applications, page 4-8
- Container Attributes, page 4-9

Also see the "Sample Container MBean Configuration File" section on page F-1.

Containers and Applications

This section defines containers and applications, and describes the relationship between them.

SESM applications and CDAT are J2EE web applications. The J2EE web server is the *container* for the applications that run in it. For example, the Jetty server is the container for the installed NWSP application.

One-to-One Relationship

The SESM core model, the NWSP sample application, and CDAT are designed and configured with the assumption that there is a one-to-one relationship between the web server container and each web application. That is, each application runs in its own web server container. If you are running two instances of the same application, or two different applications, you are running two web servers.

This one-to-one relationship means that you can configure the J2EE server differently for each application. For example, you can turn on logging for one application and turn it off for another.

Configuration File Locations

Each SESM web application (and also CDAT) has two MBean configuration files associated with it. The two files are:

- Application MBean configuration file—Configures the application. For example:

```
nwsp
    config
       nwsp.xml

cdat
    config
       cdat.xml
```

- Container MBean configuration file—Configures the J2EE server for the application. The container's config directory holds an MBean configuration file for *each* application. For example:

```
jetty
    config
       nwsp.jetty.xml
       cdat.jetty.xml
       newapplication.jetty.xml
```

This modular approach has several advantages:

- It makes it easy to switch containers. If you change the J2EE container, you must make changes to the container MBeans, such as changing class or object names, or even adding more MBeans.
- It clearly defines the process that each MBean is configuring. For example, both the container and the application have logging and debugging MBeans.

The RDP and DESS components are not web applications. Therefore, the jetty directory does not contain an MBean configuration file for those components.

Container Attributes

This section describes the attributes in the J2EE container MBean configuration files. These files are located in the container's config directory. For example:

```
jetty
    config
       nwsp.jetty.xml
       cdat.jetty.xml
```

The container MBean configuration files configure the following MBeans:

- Log MBean—Enables or disables the Jetty server logging mechanism and configures the information to appear in the jetty log files.
- Debug MBean—Enables or disables the Jetty server debugging mechanism.
- HTTP Server MBean—Configures the following:
  - The port that the Jetty server listens on for HTTP requests from subscribers and the listener thread pools. Two listeners are used, a main listener and a listener for requests on the Secure Sockets Layer (SSL). Each listener has one pool.
  - The web application to which the requests should be sent. The installed sample files identify two sample applications: the NWSP application and the captive portal application.
  - A request log, which records all HTTP requests.

Table 4-3 describes the attributes in the container MBean configuration files.

Table 4-3 Attributes in the Container MBean Configuration Files

Object Name	Attribute Name	Explanation
LogMBean	append	Indicates if messages overwrite existing contents (false) or are appended to the existing file (true). Installed default: true
	filename	Specifies the log file name and path, as follows: *application.log*/*yyyy_mm_dd*.jetty.log Where: • *application.log*—Is a Java system property. The same system property is used for all log files, so that they are all created in the same directory. Table 5-1 on page 5-4 describes how the start script sets *application.log*. • *yyyy_mm_dd*—Is the year, month, and day that the file was created. • .jetty.log—Is a constant identifying the Jetty log files.
	logTimezone	Installed default: empty
	logDateFormat	Controls the format of the date stamp in the log messages. Installed default: yyyyMMdd:HHmmss.sss
	logLabels	Controls whether or not logging messages include frame details. Installed default: false
	logOneLine	Installed default: false
	logStackSize	Controls whether or not logging messages include an indication of stack depth. Installed default: false
	logStackTrace	Controls whether or not logging messages include trace information. Installed default: false

Table 4-3 Attributes in the Container MBean Configuration Files (continued)

Object Name	Attribute Name	Explanation
DebugMBean	debug	Controls whether or not debugging messages are produced. Installed default: false
	debugPatterns	By specifying one or more patterns, you turn on a filtering mechanism that excludes any message that does not match the pattern. The patterns are file, class, or method names. Pattern matching is based on substring matches. For example, if you specify the pattern RADIUS, the software focuses on RADIUS messages. To specify multiple patterns, separate the patterns using a comma. Installed default: empty
	debugTriggers	Installed default: empty
	verbose	Specifies the level of detail reported in debugging messages. The range of allowed values is 0 (no details) to 255 (all details). Installed default: 0
	suppressStack	Controls whether or not stack information is included in debug messages. Installed default: false
	suppressWarnings	Controls whether or not warning messages are included in debug messages. Installed default: false

Table 4-3 Attributes in the Container MBean Configuration Files (continued)

Object Name	Attribute Name	Explanation
HttpServer MBean—AddListener for HTTP.SocketListener	port	Sets the port number that the web server listens on. The installed value is a Java system property named: `application.portno` **Note** The startup script sets this system property. Unless you alter the startup script, the default value in the MBean configuration file is ignored during application startup. To change the value of `application.portno`, edit the application-specific startup script. The SESM installation program sets `application.portno` in the startup script to the NWSP port that you provided during the installation process. If you are running in captive portal mode, this port value must be 80, whether you explicitly set it here by removing the reference to the Java system property or change the value of `application.portno` in the startup script.
	minThreads	Sets the minimum number of threads that this listener will maintain during periods of low load. This listener will always have system resources allocated for this number of threads. Installed default: 5
	maxThreads	Sets the maximum number of threads that this listener can allocate resources for, even during peak loads. This listener can have up to this number of threads. Installed default: 255
	maxIdleTimeMs	Specifies how long a thread can be idle (not used) before the listener deallocates it. The unit is milliseconds. Installed default: 60000
	maxReadTimeMs	Specifies the time that a read on a request can block. This is how long the web server waits for a request to come from a client after the client opens a socket connection. When maxReadTimeMs is exceeded, the web server closes the socket connection. Installed default: 60000

Table 4-3 Attributes in the Container MBean Configuration Files (continued)

Object Name	Attribute Name	Explanation

HttpServer MBean—AddListener for HTTP.SunJsseListener	port	Sets the port that the secure socket layer (SSL) listener uses. The installed value is a Java system property named:
		`application.ssl.portno`
		Note The startup script sets this system property. Unless you alter the startup script, the default value in the MBean configuration file is ignored during application startup.
		The generic startup script derives a value for `application.ssl.portno` based on the value of `application.portno`, as follows:
		`application.ssl.portno = application.portno - 80 + 443`
		To change the value of `application.ssl.portno`, edit the generic startup script.

	MinThreads	Sets the minimum number of threads that this listener will maintain during periods of low load. This listener will always have system resources allocated for this number of threads. Installed default: 5
	MaxThreads	Sets the maximum number of threads that this listener can allocate resources for, even during peak loads. The listener can allocate up to this number of threads. Installed default: 255
	MaxIdleTimeMs	Specifies the length of time a thread can be idle (not used) before the listener deallocates it. The unit is milliseconds. Installed default: 50000
	Keystore	Sets the pathname of the SSL keystore file. The keystore file is a binary file created by keytool. A sample keystore file is included in the installation. The name and location of the sample is: *jetty.home*/config/nwspkeystore Where: • *jetty.home*—Is a Java system property. The NWSP start script derives the value of *jetty.home* from an expected (installed) directory structure. To change the value of *jetty.home*, edit the start script. Unless you alter the start script, the default value for *jetty.home* specified in this MBean configuration file is ignored at run time. ⚠ **Caution** A keystore file is required for deployments that use HTTPS. HTTPS does not function without a valid keystore file. The nwspkeystore file included with the SESM installation works, but you should replace it with a keystore valid for your specific deployment. See the "HTTPS Description" section on page A-2 for more information
	Password	Must match the value in the keystore file referenced above.
	KeyPassword	Must match the value in the keystore file referenced above.

Table 4-3 Attributes in the Container MBean Configuration Files (continued)

Object Name	Attribute Name	Explanation
HttpServerMBean—LogSink Configures a log file that records the incoming HTTP requests.	This is a positional argument.	The logSink class has one argument, which specifies the name and location of the request log. The installed value is: *application.log*/*yyyy_mm_dd*.request.log Where: • *application.log*—Is a Java system property. whose value is set in the generic startup script. The same system property is used for all log files, so that they are all created in the same directory. See Table 5-1 on page 5-4 for a description of how the start script sets *application.log*. • *yyyy_mm_dd*—Is the year, month, and day that the file was created. The installation program uses the appropriate pathname delimiter for the installation platform. • .request.log—Is a constant identifying an HTTP request file.
	retainDays	Indicates the number of days to keep an old log file before deleting it. Installed default: 90
	append	Indicates whether or not to append messages to an existing file or to create a new file for each application instance. Installed default: true

Table 4-3 Attributes in the Container MBean Configuration Files (continued)

Object Name	Attribute Name	Explanation
HttpServer MBean—<Call AddWebApplication> The first call to this class adds the NWSP application to run on the web server.	These are positional arguments.	AddWebApplication has five positional arguments: 1. The first positional argument specifies the virtual host name for the web server application. 2. The second positional argument specifies the context path for locating the web server application. For example, / or /pathname/*. 3. The third positional argument identifies the location of the application. The XML file sets this value to: *application.home*/docroot Where: *application.home* is a Java system property. 4. The fourth positional argument identifies the location of the webdefault.xml file for this application. The XML file sets this value to: *jetty.home*/config/webdefault.xml Where: *jetty.home* is a Java system property 5. The fifth positional argument specifies whether or not web archive (WAR) files are used. Valid values are TRUE and FALSE. Set this value to FALSE, since NWSP and CDAT are not WAR files. The first three arguments define the location of the web server application. `host/context/application` The NWSP start script derives the values for *application.home* and *jetty.home* from an expected (installed) directory structure. To change the value of *application.home* or *jetty.home*, edit the start script.

Table 4-3 Attributes in the Container MBean Configuration Files (continued)

Object Name	Attribute Name	Explanation
HttpServer MBean—<Call AddWebApplication> The second call to this class adds the captive portal application to run on the web server.	These are positional arguments.	AddWebApplication has five positional arguments: 1. The first positional argument is not used when calling the captive portal application. 2. The second positional argument specifies the context path for locating the web server application. For example, / or /pathname/*. (context) 3. The third positional argument identifies the location of the captive portal application. The XML file sets this value to: *install.root*/captiveportal/docroot Where: *install.root* is a Java system property 4. The fourth positional argument identifies the location of the webdefault.xml file for the captive portal application. The XML file sets this value to: *jetty.home*/config/webdefault.xml Where: *jetty.home* is a Java system property 5. The fifth positional argument specifies whether web archive (WAR) files are used. Valid values are TRUE and FALSE. Set this value to FALSE, since NWSP and CDAT are not WAR files. The first three arguments define the location of the captive portal application. `host/context/application` The NWSP start script derives the values for *install.root* and *jetty.home* from an expected (installed) directory structure. To change the value of *application.home* or *jetty.home*, edit the start script.

Configuring an SESM Web Application

This section describes how to configure an SESM web application, using the NWSP application as an example. The section includes the following topics:

- SESM Application Attributes, page 4-17
- Associating SSGs and Subscriber Requests, page 4-27

Also see the "Sample Application MBean Configuration File" section on page F-3.

SESM Application Attributes

This section describes the SESM application MBean configuration file. This file is located in the application's config directory. For example:

```
nwsp
    config
       nwsp.xml
```

The application MBean configuration file configures the following MBeans:

- Logger—The com.cisco.aggbu.jmx.LoggerMBean configures both logging and debugging tools. The logging tool logs SESM web application activity. The debugging mechanism produces messages useful to developers in debugging applications.
- ManagementConsole—This MBean configures a management console port for development and testing purposes. On this port, you can see the currently set values for all attributes in all of the MBean configuration files.
- SSD—This MBean configures SESM features and options, including the SESM mode.
- SSDDemoMode—This MBean configures SESM in demo mode.
- SSG—The SSG MBean configures communication between SESM web application and SSG. These components communicate using the RADIUS protocol, so this MBean includes RADIUS protocol attributes. The MBean also includes attributes that determine which SSG should handle a subscriber request.
- AAA—The AAA MBean configures communication between SESM web application and the RADIUS servers.
- captiveportal—This MBean configures captive portal information, including the URL that the captive portal redirects to, which should be the SESM web application.
- context parameters— Context parameters are used by an application for any arbitrary reason. The The NWSP application uses context parameters to control web page content based on location.

Table 4-4 explains the configurable attributes in the MBeans listed above.

Table 4-4 Attributes in the Application MBean Configuration File

Object	Attribute Name	Explanation

ManagementConsole	Port	Specifies a port for a management console.
		The management console displays the current settings of all attributes in all of the MBean configuration files. The console is useful in development and testing environments.
		Note The ManagementConsole is the HTML adaptor server included with the Sun example JMX server. However, the HTML adaptor server is not production quality. For example, configuration changes that you make using the management console are not persistent. You should remove the HTML adaptor server from your configuration before transitioning the SESM deployment to public use.
		To remove the JMX HTML adaptor server, comment out the following lines in the configuration files:
		`<Configure jmxname="com.cisco.aggbu:name=ManagementConsole">`
		`<Call name="start"/>`
		`</Configure>`
		The port attribute is set to a Java system property named:
		`management``.portno`
		All of the installed startup scripts set this Java system property to the following value:
		`application.portno + 100`
		For example, if the application.portno is 8080, the management.portno is 8180.
		This runtime setting overrides any value you enter in the configuration file. To change the value of this attribute, edit the start script.

	AuthInfo	AuthInfo provides a level of access control on the Management Console. When a user attempts to access the management console port from a web browser, a logon window appears first. The user must enter a user ID and password that matches the values specified here. AuthInfo requires two positional arguments: **1.** User ID—Enter a user ID that will be required to access the management console. The default value in all of the MBean configuration files is `MgmtUser`. **2.** Password—Enter a password that will be required to access the management console. The default value in all of the MBean configuration files is `MgmtPassword`.

Table 4-4 Attributes in the Application MBean Configuration File (continued)

Object	Attribute Name	Explanation
Logger MBean	debug	Turns debugging on (value is true) or off (value is false). The following parameters control the contents of debug messages that the application generates: logFrame, logStack, logThread, debugPatterns, and debugThreads. When debug is false, the application does not generate debug messages but it can still generate logging messages. The following parameters control the types of logging messages produced: trace and warning. Installed default: false
	debugPatterns	By specifying one or more patterns, you turn on a filtering mechanism that excludes any message that does not match the pattern. The patterns are file, class, or method names. Pattern matching is based on substring matches. For example, if you specify the pattern RADIUS, the software focuses on RADIUS messages. To specify multiple patterns, separate the patterns using a comma. Installed default: empty, which means that you receive all messages.
	debugThreads	Specifies a specific thread name for which to show debugging messages. You can specify multiple thread names, separating them using a comma. By default, no thread name is specified. Because each user interaction with the SESM web application takes place in a thread named for that user, this parameter can be used to focus the logging trace on a specific user activity. (This feature is not implemented in SESM Release 3.1(1).) Installed default: empty
	debugVerbosity	Specifies the level of detail in debugging messages. When the debug attribute is set to false, this attribute is ignored. Values are: • MAX • MED • LOW Installed default: LOW
	logDateFormat	Specifies format of dates in the log file. Installed default: yyyyMMdd:HHmmss.SSS

Table 4-4 Attributes in the Application MBean Configuration File (continued)

Object	Attribute Name	Explanation
LoggerMBean *(continued)*	logFile	Specifies the log file name and location. The installed default is: *application.log*/*yyyy_mm_dd*.application.log Where: • *application.log*—Is a Java system property. The same system property is used for all log files, so that they are all created in the same directory. See Table 5-1 on page 5-4 for a description of how the start script sets *application.log*. • *yyyy_mm_dd* —Is the year, month, and day that the file was created. • application.log—Is a constant identifying the application log files.
	logFrame	Controls whether or not to log the calling member function. Installed default: false
	logStack	Controls whether or not to log stack traces. Installed default: false
	logThread	Controls whether or not to log thread IDs. Installed default: true
	logToErr	Controls whether or not to route log messages to stderr, in addition to the log file. This parameter is useful for monitoring the SESM web application at the command line. Displaying output to stderr is not recommended for production deployments. Installed default: true
	trace	Controls whether or not to log trace messages. These messages indicate entry and exit to code points. Installed default: true
	warning	Controls whether or not to log warning messages (nonfatal exceptions). Installed default: true

Table 4-4 Attributes in the Application MBean Configuration File (continued)

<table>
<tr><th>Object</th><th>Attribute Name</th><th>Explanation</th></tr>
<tr><td>SSD</td><td>mode</td><td>An SESM web application runs in one of the following modes. The SESM installation program sets the mode according to the options you choose during installation.
• RADIUS—In this mode, the SESM web application communicates with SSG and a RADIUS server.
• Demo—SESM runs in this mode when you choose the Demo option during installation. In this mode, the SESM web application does not communicate with other components. Rather, it simulates communication by reading data from a Merit flat file. This mode is intended for demonstrations only, when network components such as SSG, RADIUS, or an LDAP server are not available.
• DESS—In this mode, the SESM web application communicates with SSG and an LDAP directory. The LDAP directory communication relies on a Cisco application programming interface known as directory-enabled service selection (DESS).
The MBean configuration file defines a Java system property for mode:
`ssd.mode`
This system property is different from most of the other system properties used in the MBean configuration files, in that, by default, the startup script does *not* set this system property. Therefore, the application runs in the mode specified in the MBean configuration file unless you explicitly override that value at run time.
To change the mode, you can:
• Reinstall the software.
• Edit the MBean configuration files, changing the mode and other attributes, as appropriate.
• Use the mode option on the SESM application startup script command line. This command line option provides a way to quickly switch between modes for testing purposes. You might need to alter the start script to access a different set of MBean configuration files for each mode, or use some other method to ensure that the attributes match the mode you are using. The syntax is:
– on Solaris: `jetty/bin/startNWSP.sh -mode {Demo | RADIUS | DESS}`
– on Windows: `jetty\bin\startNWSP.cmd {Demo | RADIUS | DESS}`
Note The best way to change the SESM mode is to re-install the software. Several other configuration attributes must be aligned with the mode for SESM to run properly in the selected mode. Also, you might not have all of the appropriate components to run in a mode other than the one you installed. For example, a demo installation does not install the DESS component.</td></tr>
<tr><td></td><td>singleSignOn</td><td>Enables (true) or disables (false) the single sign-on feature.
If single sign-on is enabled, the SESM web application does not ask a PPP subscriber to authenticate (log on). Instead, the SESM web application uses the SSG's PPP authenticated identity. Installed default: false</td></tr>
</table>

Table 4-4 Attributes in the Application MBean Configuration File (continued)

Object	Attribute Name	Explanation
SSD *(continued)*	profileCache Period	Specifies the time in seconds that a service or group object must be idle in the cache before its resources are deallocated from memory. Installed default: 600
	autoConnect	In RADIUS mode, this parameter is ignored. The automatic connection feature is always available, regardless of parameter settings. In RADIUS mode, the SSG always performs automatic service connections for all services marked as auto connect in a subscriber's profile. In DESS mode, the SSG performs automatic connections if it has the service list. If SSG does not have the service list, you can set this autoConnect parameter to allow the SESM application to perform the automatic connections. The Add Services option, which is set during RDP installation, controls whether or not SSG has a service list in DESS mode. The Add Services option configures RDP to either: • Return a service list to SSG—In this case, SSG performs automatic connections for services marked as auto connect in a subscriber's profile. • Not return a service list to SSG—In this case, SSG cannot perform automatic connections. The advantage to this configuration is that it saves memory on the SSG host (the NRP on the Cisco 6400 UAC). If you configure RDP so that it does not return a service list to SSG, change the value of this autoConnect parameter to true to enable automatic connections by the SESM web application.
SSDDemoMode	demoDataFile	Specifies the file that contains data for the demo mode. The installed value is: *application.home*/config/demo.txt Where: *application.home* is a Java system property The NWSP start script derives the value for *application.home* from an expected (installed) directory structure. To change the value of *application.home*, edit the start script.

Table 4-4 Attributes in the Application MBean Configuration File (continued)

Object	Attribute Name	Explanation
SSG—Global attributes The global attributes apply to all SSGs that the SESM web application might communicate with. To determine how an SSG was configured, use the **show run** command on the SSG host.	throttle	The maximum number of simultaneous requests that SESM web applications can send to SSG. This is a RADIUS protocol attribute. The RADIUS protocol queues additional requests and issues them as SSG returns responses or timeout messages for previous requests. You cannot override this global value. Installed default: 20
	PORT	The global value for RADIUS ports on the SSG hosts. This value must match the value that was configured on the SSG host using the following command: `ssg radius-helper` *`authenticationPort`* You can create subnet entries in the MBean configuration file to override this global value for specific SSGs.
	TIMEOUTSECS	The number of seconds the SESM web application waits before timing out RADIUS packets that it sends to SSG. You cannot override this global value. Installed default: 5
	RETRIES	The number of times the SESM web application resends a RADIUS packet to SSG if no response is received. You cannot override this global value. Installed default: 3
	SECRET	The global value for the RADIUS protocol shared secret used for communication between the SESM web application and the SSGs. This value must match the value entered on the SSG host using the **ssg radius-helper key** command. You can create subnet entries in the MBean configuration file to override this global value for specific SSGs.
	MASK	The global value for the mask that the SESM web application applies to incoming subscriber IP addresses to derive an IP address for the SSG. You can create subnet entries in the MBean configuration file to override this global value for specific subnets.
	BUNDLE_ LENGTH	The global value for the port bundle length that SSGs use when the host key feature is enabled. Currently, this value can be either: • 0—A value of 0 indicates that SSGs are not using the host key feature. • 4—The port bundle length is the number of bits that SSG uses to indicate bundled slots. For example, a value of 4 indicates 16 bundled slots. This value must match the value used in the following command on the SSG host: `ssg port-map` *`length`*

Table 4-4 Attributes in the Application MBean Configuration File (continued)

Object	Attribute Name	Explanation
SSG—Subnet entries Use subnet entries to override the global values or to map client subnets to specific SSGs when the host key feature is not being used.	Subnet entries use positional arguments.	The call to setSubnetAttribute has four positional arguments: 1. *subnetAddress* is the subnet for which you are explicitly setting a value, overriding the globally set value. 2. *subnetMask* is the mask that can be applied to the subscriber's IP address to derive the subnet. 3. *argumentName* is the argument that you are explicitly setting. 4. *argumentValue* is the value for *argumentName*. See the "Associating SSGs and Subscriber Requests" section on page 4-27 for more information.
AAA This MBean defines communication between the SESM web application and the RADIUS server, which occurs only when the SESM application is running in RADIUS mode.	Connection	The Configure element in the AAA MBean includes a connection attribute whose value is either: • ServiceProfile—The MBean for this connection type includes the servicePassword attribute. • GroupProfile—The MBean for this connection type includes the groupPassword attribute. The connection name identifies the type of request.
	throttle	The maximum number of simultaneous requests that SESM web applications can send to a RADIUS server. This is a RADIUS protocol attribute. The RADIUS protocol queues additional requests and issues them as the server returns responses or timeout messages for previous requests. Installed default: 256
	timeOut	The number of seconds the SESM web application waits before timing out RADIUS packets that it sends to the AAA server. Installed default: 4
	retryCount	The number of times the SESM web application resends packets to the AAA server if no response is received. Installed default: 3
	primaryIP	The IP address or the host name of the primary AAA server.
	primaryPort	The port number that the primary RADIUS server listens on. Default: 1812
	secret	The shared secret used between the RADIUS server and the SESM web application. The shared secret must be the same for the primary and secondary servers. It must match the secret specified when you configured SESM as a NAS client on the RADIUS server. Default: `cisco`.
	secondaryIP	The IP address or the host name of the secondary AAA server. If you are not using a secondary RADIUS server, enter the same value used for the primary server.

Table 4-4 Attributes in the Application MBean Configuration File (continued)

Object	Attribute Name	Explanation
AAA *(continued)*	secondaryPort	The port number that the secondary RADIUS server listens on. If you are not using a secondary RADIUS server, enter the same value used for the primary server. Default: 1812
	servicePassword	The password that the SESM web application uses to request service and group profiles from the RADIUS server. This password must match the value that was configured on the SSG host with the following command: `ssg service-password` *`password`* The service-password value must be the same on all SSGs. Default: `servicecisco`
	groupPassword	The password that the SESM web application uses to request group profiles from the RADIUS server. Default: `groupcisco`
Captive Portal	captureToURL	The URL of the NWSP application to which the captive portal application redirects the subscriber's browser. The captive portal application captures the original URL that was requested by the subscriber and forwards it to the SESM web application along with the redirect. The SESM web application can then honor the subscriber's originally requested URL after authentication occurs.

Table 4-4 Attributes in the Application MBean Configuration File (continued)

Object	Attribute Name	Explanation
Options context parameters	useIcons	Controls whether the application uses icons or text when it displays, on the web page, the services that a subscriber is authorized to use. Default: TRUE
	confirmAtService Logon	Controls whether or not the application prompts the user for confirmation before it acts on a request to start a service. Default: FALSE
	confirmAtService Logoff	Controls whether or not the application prompts the user for confirmation before it acts on a request to log off. Default: TRUE
	confirmAtAccount Logoff	Controls whether or not the application prompts the user for confirmation before it acts on a request to log off of the SESM application. Default: TRUE
	sessionTimeOut	The number of seconds of inactivity allowed before the application closes a session. This value overrides the timeout value in the nwsp.jetty.xml file. Default: 7200
Arbitrary context parameters	locations and brands	Defines specific locations and brands and the attributes associated with each one. The NWSP application uses the location context parameter to define an initial URL and meaningful symbols (rivers and churches) related to the location. It uses the brand context parameter to define an initial URL and an email address. You can define additional context parameters, for any arbitrary use, by copying the format used in the nwsp.xml file to define the location and brand parameters. See the "Sample Application MBean Configuration File" section on page F-3 section for context parameter examples. To define a context parameter, use separate XML elements to define the following: • The context parameter (for example, location) • The related subcontext parameters (for example, London, Paris, New York) • The attributes that are associated with each subcontext value (for example, URL values, river values, and church values) For new context parameters to be meaningful, the SESM web application must be changed to do something with the new parameters. You can add new subcontext parameters (new locations or new brands) without changing the web application.

Associating SSGs and Subscriber Requests

A typical SESM deployment consists of multiple SSGs. An SESM web application must know which SSG is handling each subscriber request. This section describes how to configure the associations between a subscriber request and its SSG. It includes the following topics:

- Using Host Key with Identical SSG Configurations, page 4-27
- Using Host Key with Varying SSG Configurations, page 4-28
- Specifically Mapping SSGs to Subscriber Subnets, page 4-29
- Format of Global and Subnet Attribute Elements, page 4-30

Using Host Key with Identical SSG Configurations

The easiest way to associate the correct SSG with each subscriber request is to use the host key port bundle feature on all SSGs, and configure certain attributes identically on all of the SSG hosts. We recommend using host key unless you need backward compatibility with SSD Release 2.5(1).

Note To use the host key port bundle feature, the Cisco 6400 NRP must be running Cisco IOS Release 12.2(2)B or later and the SSG host key feature must be configured appropriately.

When the host key feature is enabled on an SSG, the SSG replaces the subscriber IP address in the request with a software token (or key) when it forwards the request to SESM. The SESM application uses this key in its responses to SSG, and the SSG does an internal translation to an actual host object.

The key is a unique combination of an SSG IP address from a range of IP addresses and a port number from a range of port numbers, as follows:

IP_address:*port*

The IP address and port ranges are configured on each SSG. The key uniquely identifies each subscriber currently logged on to SESM, even when multiple subscribers are using the same IP address.

To use the host key feature to associate SSGs, follow these procedures:

1. Enable and configure the host key feature on all of the SSGs, as described in the "Configuring the Host Key Port Bundle Feature on SSG" section on page B-2.
2. Configure the same values on all of the SSG hosts for the following attributes:
   - Port—The SSG port on the SSG host. Specify the port that SSG uses to listen for RADIUS requests from an SESM application. Configure this value on the SSG host with the following command:

     ```
     ssg radius-helper authenticationPort
     ```

   - Shared secret—The shared secret used for communication between SSG and an SESM application. Configure this value on the SSG host with the following command:

     ```
     ssg radius-helper key
     ```

   - Port bundle length—The number of bits that SSG uses for port bundling when the host key feature is enabled. This value must be 0 or 4. Configure this value on the SSG host with the following command:

     ```
     ssg port-map length
     ```

3. Enter these globally configured values when the SESM installation program prompts you for them. These values are reflected in global elements in the <Configure name="SSG"> section of the application MBean configuration file, as the following example illustrates.

Example Using Host Key

When SSG has the host key feature enabled and configured, you can set all parameters globally.

```
<Configure name="com.cisco.aggbu:name=SSG">
<Call name="setGlobalAttribute"><Arg>PORT</Arg><Arg>1812</Arg></Call>
<Call name="setGlobalAttribute"><Arg>SECRET</Arg><Arg>cisco</Arg></Call>
<Call name="setGlobalAttribute"><Arg>MASK</Arg><Arg>255.255.255.255</Arg></Call>
<Call name="setGlobalAttribute"><Arg>BUNDLE_LENGTH</Arg><Arg>4</Arg></Call>
</Configure>
```

In this example, all SSGs are configured to use a port of 1812 and a shared RADIUS secret of `cisco`. The BUNDLE_LENGTH of 4 indicates that host key is configured on all SSGs.

The MASK attribute specifies the mask that SESM applies to the client (source) IP address in a received message to determine the client's subnet, and, from that, the SSG IP address. However, when host key is being used, the client (source) IP address is the SSG IP address. The SESM installation program provides the default mask of 255.255.255.255.

Using Host Key with Varying SSG Configurations

If host key is enabled on all SSGs, but some are configured differently, you can configure the global case and then specifically configure the exceptions. For example, if all but one SSG is assigned the same shared secret, you can configure the shared secret attribute globally, and then add one subnet entry to configure the different secret for the one SSG.

The installation program lets you provide one set of SSG global attribute values and one subnet entry. It records these attribute values in the <Configure name="SSG"> section of the application MBean configuration file, as illustrated in the following example.

Example Using Host Key with One Non-Complying SSG

In this example, host key is enabled on all SSGs. In addition, all SSGs are using the same port, secret, and client IP address mask, except that one SSG uses a different port. In this case, you can set all parameters globally, and then use one subnet entry to define:

- The client subnet being serviced by the SSG that uses the nonconforming port.
- The port value that overrides the globally-set port value.

In the following example, the SSG that services subnet 10.1.1.0 uses port 1245.

```
<Configure name="com.cisco.aggbu:name=SSG">
<Call name="setGlobalAttribute"><Arg>PORT</Arg><Arg>1812</Arg></Call>
<Call name="setGlobalAttribute"><Arg>SECRET</Arg><Arg>cisco</Arg></Call>
<Call name="setGlobalAttribute"><Arg>MASK</Arg><Arg>255.255.255.255</Arg></Call>
<Call name="setGlobalAttribute"><Arg>BUNDLE_LENGTH</Arg><Arg>4</Arg></Call>
<Call name="setSubnetAttribute"><Arg>10.1.1.0</Arg><Arg>255.255.255.0</Arg><Arg>PORT
</Arg><Arg>1245</Arg></Call>
</Configure>
```

Specifically Mapping SSGs to Subscriber Subnets

Each request arriving at an SESM web application contains a source, or client, IP address. SESM uses this client IP address to determine which SSG should handle each request.

- If the configuration file explicitly provides an SSG IP address for a subnet or a specific client IP address, SESM uses that SSG. You code an explicit IP address in a <subnet> element. The MASK value in the subnet element specifies whether the element applies to a subnet or to a specific subscriber IP address. The <IP> parameter in the subnet element specifies the SSG IP address.

  For example, the following subnet entry explicitly sets the SSG IP address to 10.6.7.1 for subnet 10.2.0.0:

  ```
  <Call name="setSubnetAttribute">
  <Arg>10.2.0.0</Arg><Arg>255.255.0.0</Arg><Arg>IP</Arg><Arg>10.6.7.1</Arg></Call>
  ```

- If an explicit IP address for the SSG is not provided, SESM masks the subscriber's IP address to determine the SSG that should handle the request. Use masking as follows:
  - If host key is enabled—The host key feature replaces the original client IP address with the IP address of the SSG. (The port bundle key appended to the address preserves a unique identity for each subscriber). Since the client IP address is the SSG IP address, a global setting for MASK of 255.255.255.255 correctly results in the client IP address being used as the SSG IP address.
  - If the SSG uses the first IP address in a particular set of client subnets—Specify the mask that SESM web application can apply to the client IP address to derive the SSG IP address. For example, if, for all 10.x.0.0 client subnets, the SSG IP address is 10.x.0.1, you would specify a subnet of 10.0.0.0 and a mask of 255.0.0.0.
  - If the SSG IP is the first IP in all client subnets—You can set a global value for mask. For example, for all subscriber addresses x.y.z.n, if the SSG always has an IP address of x.y.0.1, then use a global mask of 255.255.0.0.

Note Set the widest global or subnet mask possible. Each SSG IP address consumes some resources on the machine where the SESM application is running. (Each one uses an open file descriptor.) For example, even when the Cisco 6400 UAC is using host key, a mask of 255.255.255.0 is desirable, so that the SESM uses a single SSG IP address rather than 254 different SSG IP addresses. A mask of 255.255.255.255 is the least efficient, but it is the default setup.

Example Mapping Client Subnets to SSGs

In this example, host key is not being used.In this case, you must explicitly define the mapping from subscriber subnet to the SSG IP address.

```
<Configure name="com.cisco.aggbu:name=SSG">
<Call name="setGlobalAttribute"><Arg>PORT</Arg><Arg>1645</Arg></Call>
<Call name="setGlobalAttribute"><Arg>SECRET</Arg><Arg>cisco</Arg></Call>
<Call name="setGlobalAttribute"><Arg>MASK</Arg><Arg>255.255.255.255</Arg></Call>
<Call name="setGlobalAttribute"><Arg>BUNDLE_LENGTH</Arg><Arg>0</Arg></Call>
<Call name="setSubnetAttribute"><Arg>10.1.1.0</Arg><Arg>255.255.255.0</Arg><Arg>IP
</Arg><Arg>10.21.1.2</Arg></Call>
<Call name="setSubnetAttribute"><Arg>10.1.2.0</Arg><Arg>255.255.255.0</Arg><Arg>IP
</Arg><Arg>10.21.2.2</Arg></Call>
<Call name="setSubnetAttribute"><Arg>10.1.3.0</Arg><Arg>255.255.255.0</Arg><Arg>IP
</Arg><Arg>10.21.3.2</Arg></Call>
<Call name="setSubnetAttribute"><Arg>10.1.4.0</Arg><Arg>255.255.255.0</Arg><Arg>IP
</Arg><Arg>10.21.4.2</Arg></Call>
</Configure>
```

Format of Global and Subnet Attribute Elements

You can set the attributes that associate an SSG with subscriber requests globally, by client subnet, or for a specific client IP address, as follows:

- Global attribute elements—A global setting applies to all SSGs. For example, a global shared secret setting means that all SSGs are configured using the same secret. The global attributes are: PORT, SECRET, MASK, and BUNDLE_LENGTH.
- Subnet attribute elements—The subnet attributes apply to a specific subnet and override the global attribute value. The subnet attributes are optional; if any of them are not specifically coded, the global attribute value is used. Subnet attributes that you can supply are: PORT, SECRET, MASK, BUNDLE_LENGTH, and IP. The IP attribute is the IP address of the SSG for a specified subnet.

  You can also specify some optional session information in a subnet entry, using context parameter values. See Table 4-5.
- A specific client IP address is specified in a subnet element.

The format for the global attribute entries is illustrated in the following examples:

```
<Configure name="com.cisco.aggbu:name=SSG">
<Call name="setGlobalAttribute"><Arg>PORT</Arg><Arg>1645</Arg></Call>
<Call name="setGlobalAttribute"><Arg>SECRET</Arg><Arg>cisco</Arg></Call>
<Call name="setGlobalAttribute"><Arg>MASK</Arg><Arg>255.255.255.0</Arg></Call>
<Call name="setGlobalAttribute"><Arg>BUNDLE_LENGTH</Arg><Arg>0</Arg></Call>
</Configure>
```

The format for subnet entries is:

```
<Call name="setSubnetAttribute">
<Arg>subnetAddress</Arg>
<Arg>subnetMask</Arg>
<Arg>argumentName</Arg>
<Arg>argumentValue</Arg>
</Call>
```

Where:

subnetAddress is the subnet for which you are explicitly setting a value, overriding the globally set value.

subnetMask is the mask that can be applied to the subscriber’s IP address to derive the subnet.

argumentName is the argument that you are explicitly setting. See Table 4-5.

argumentValue is the value for *argumentName*. See Table 4-5.

Table 4-5 Argument Names and Values for Subnet Entries

***argumentName* Value**	***argumentValue* Explanation**
PORT	The SSG port for the specified subnet. Overrides the globally-set SSG port.
MASK	The mask used on the subscriber’s IP address to derive the subnet. Overrides the globally-set mask.
SECRET	The shared secret used between SESM and SSG. Overrides the globally-set shared secret.

Table 4-5 Argument Names and Values for Subnet Entries (continued)

***argumentName* Value**	***argumentValue* Explanation**
BUNDLE_LENGTH	The host key bundle length used on the SSG. Overrides the globally-set bundle length. The bundle length is the number of bits that SSG uses for the port bundle feature. For example, a value of 4 indicates 16 bundled slots. A value of 0 indicates that the SSG is not using the host key and port bundle mechanism. This value must match the value used in the following command on the SSG host: `ssg port-map length` To determine how SSG has configured the port bundle length, use the **show run** command on the SSG host.
IP	Explicitly sets the IP address for the SSG that services the specified *subnetAddress*.
SESSION_LOCATION	The location associated with the specified subnet. Valid values are defined as subcontext parameters under the location context parameter in the nwsp.xml configuration file. The installed file defines the following locations: London, Paris, and New York. For the context parameters to have meaning, the SESM web application must support them. The NWSP application uses the location context parameter to define an initial URL and meaningful symbols related to the location.
SESSION_BRAND	The brand of service associated with the specified subnet. Valid values are defined as subcontext parameters under the brand context parameter in the nwsp.xml configuration file. The installed file defines the following brands: acme, cisco, silver, and gold. For the context parameters to have meaning, the SESM web application must support them. The NWSP application uses the brand context parameter to define an initial URL and an email address.

Configuring RDP

This section describes how to configure the RDP application. The section includes the following topics:

- RDP Modes, page 4-31
- RDP Attributes, page 4-32

Also see the "Sample RDP MBean Configuration File" section on page F-13.

RDP Modes

RDP can run in two modes:

- Non-proxy mode—In this mode, RDP uses the DESS API to obtain authentication and authorization information from the LDAP directory.
- Proxy mode—In this mode, RDP sends authentication requests to a RADIUS server. It uses the DESS API to obtain authorization information from the LDAP directory.

You choose the mode during RDP installation. The content of the rdp.xml file is significantly different depending on the mode. Therefore, to change the mode, we recommend reinstalling the RDP component. (Choose a Custom installation to reinstall a single component.)

RDP Attributes

The MBean configuration file for RDP is located in:

```
rdp
    config
       rdp.xml
```

The rdp.xml file configures the following MBeans:

- Logger—The com.cisco.aggbu.jmx.LoggerMBean configures both logging and debugging tools. The logging tool logs RDP application activity. The debugging mechanism produces messages useful to developers in debugging applications. See the *Cisco Subscriber Edge Services Web Developer Guide* for more information about debugging an application.
- RDPPacketFactory—This MBean creates RDP packets that analyze and process requests from SSG. Each request becomes a series of packets. Each type of packet is handled by a different packet handler.
- RDP—The RDP MBean listens for requests sent through SSG.
- ManagementConsole—This MBean configures a management console port. Administrators can go to this console port on a web browser and see the currently set values for all attributes in all of the MBean configuration files.
- AAA—This MBean applies only when RDP is running in Proxy mode. In that mode, RDP is a RADIUS proxy server. The RDP AAA MBean defines the proxy server attributes.

Table 4-6 explains the configurable attributes in these MBeans.

Table 4-6 Attributes in the RDP MBean Configuration File

MBean	Attribute Name	Explanation
Logger	See the description for Logger MBean in Table 4-4 on page 4-18.	
RDPPacketFactory	The only attributes in this MBean that administrators are expected to change are the password attributes associated with service profile requests. These password attributes are used to identify a service request as one of the following: a single service request, a service group request, or a next hop table request. SSG sets the password in the request; RDP interprets the password. You must configure the values on both sides, as follows: • On SSG, you set the values for these three passwords using IOS commands. • On RDP, you set the values for the three passwords as described here. If the password in a request from SSG does not match one of the three values you set on the RDP side, the request is discarded. You can find the password attributes in this MBean by searching the file for the following string: `<arg>PASSWORD:` **Note** There are no security implications to these attributes. It might be helpful to think of them as identifying keys, rather than passwords. The three password attributes are: • ServiceRequest—Requests containing this password are handled by the ServiceRequest packet handler. The ServiceRequest packet handler uses the DESS API to obtain a list of authorized services for a subscriber. On the SSG side, set this password using the following command: `ssg service-password` *`servicePassword`* • GroupRequest—Requests containing this password are handled by the GroupRequest packet handler. The GroupRequest packet handler forwards requests to a RADIUS server to obtain a list of authorized services for the group of which the subscriber is a member. Group requests are relevant only when RDP is configured in proxy mode. • NextHopRequest—Requests containing this password are handled by the ProxyNextHop packet handler. The Proxy NextHop packet handler passes authentication requests to the AAAMBean when the RDP is configured in proxy mode, or through DESS to the directory when the RDP is not in proxy mode. On the SSG side, set this password using the following command: `ssg next-hop download` *`nextHopTableName password`* See Appendix E, “RDP Packet Handlers,” for more information about how RDP processes requests from SSG.	

Table 4-6 Attributes in the RDP MBean Configuration File (continued)

MBean	Attribute Name	Explanation
RDP	secret	Enter the RADIUS client shared secret to be used for communication between SSG and RDP. It must be a different value from the shared secret used for RDP to RADIUS communication. The installation program's displayed default is cisco.
	localIPAddress	Enter the IP address or host name of the RDP. **Note** This value cannot be localhost (127.0.0.1)
	localPort	Enter the port on which the RDP will listen. The installation program's displayed default is 1812.
	minThreads	Sets the minimum number of threads that RDP will maintain during periods of low load. RDP will always have system resources allocated for this number of threads. Installed default: 10
	maxThreads	The total number of simultaneous requests that the RDP can handle. If the RDP is receiving more requests than the current setting, and the RDP host machine is not processor-bound, then you can increase this number for a potential performance improvement. Installed default: 256
	maxIdleTimeMs	The number of milliseconds that a thread can remain idle before the system deallocates its resources. Installed default: 10000
ManagementConsole	See the description for "ManagementConsole" in Table 4-4 on page 4-18.	

Table 4-6 Attributes in the RDP MBean Configuration File (continued)

MBean	Attribute Name	Explanation
AAA This MBean applies only when RDP is configured in Proxy mode.	Connection	The Configure tag for the AAA MBean includes a connection attribute whose value is either: • NextHop • Proxy The RDP proxy handlers use the connection name to identify the AAA server to proxy the request to.
	throttle	The maximum number of simultaneous requests that RDP can send to a RADIUS server. This is a RADIUS protocol attribute. The RADIUS protocol queues additional requests and issues them as the RADIUS server returns responses or timeout messages for previous requests. Installed default: 256
	timeOut	The number of seconds RDP waits before timing out RADIUS packets that it sends to the AAA server. Installed default: 4
	retryCount	The number of times RDP resends packets to the AAA server if no response is received. Installed default: 1
	primaryIP	Enter the IP address or the host name of the primary RADIUS AAA server that you want RDP to communicate with.
	primaryPort	Enter the port number on the primary RADIUS server host that the RADIUS server listens on.
	AAASecret	Enter the RADIUS client shared secret used between RDP and the RADIUS server. The shared secret must be the same for both servers. The installation program's displayed default value is `cisco`.
	secondaryIP	Enter the IP address or the host name of the secondary RADIUS server. If you are not using a secondary RADIUS server, enter the same value used for the primary server.
	secondaryPort	Enter the port number on the secondary RADIUS server host that the RADIUS server listens on. If you are not using a secondary RADIUS server, enter the same value used for the primary server.

Configuring CDAT

This section describes how to configure the CDAT application. The section includes the following topics:

- Cookies Required, page 4-36
- CDAT Attributes, page 4-36

Also see the "Sample CDAT MBean Configuration File" section on page F-16.

Cookies Required

Make sure that the cookies feature is enabled on the browser where you are running CDAT. If the CDAT application seems to log itself off unexpectedly, check your cookies setting.

CDAT Attributes

The CDAT MBean configuration file is located in:

```
cdat
    config
       cdat.xml
```

The cdat.xml file configures the following MBeans:

- Logger—The Logger MBean configures both logging and debugging tools. The logging tool logs CDAT application activity. The debugging mechanism produces messages useful for debugging.
- ManagementConsole—This MBean configures a management console port. Administrators can go to this console port on a web browser and see the currently set values for all attributes in all of the MBean configuration files.
- CDAT—The CDAT MBean configures resource attributes for the CDAT application.

Table 4-7 explains the configurable attributes in this MBean.

Table 4-7 Attributes in the CDAT MBean Configuration File

MBean Name	Attribute Name	Explanation
Logger	See the description for Logger MBean in Table 4-4 on page 4-18.	
ManagementConsole	See the description for ManagementConsole in Table 4-4 on page 4-18.	
CDAT	sessionTimeout	The maxmimum period of inactivity allowed during a CDAT login, after which the user will be logged out. Values are in seconds. A negative value will prevent the user from ever being logged out. Changes will only take effect for subsequent logins. Default: 600
	maxVariables	The maximum number of page/page instance variables allowed for each CDAT session. This number affects how many pages can be visited before their state is lost, though it is not a one-to-one mapping. If you see many StateTimedOut errors, you should increase this number. Default: 40
	queryMaxResults	The maximum number of results to return from any one directory query. Changes will take immediate effect. A value of zero will remove any limits. Default: 500
	queryTimeout	The timeout (in milliseconds) for directory queries. Changes will take immediate effect. A value of zero will cause an infinite timeout. Default: 0

Configuring DESS

This section describes how to configure the DESS component. The section includes the following topics:

- DESS Attributes, page 4-37
- Extending the Directory Schema and Installing Initial RBAC Objects, page 4-40

Also see the "Sample DESS MBean Configuration File" section on page F-17.

DESS Attributes

The MBean configuration file for DESS is located in:

```
dess-auth
    config
       config.xml
```

This file applies to applications that incorporate the Dess and Auth APIs:

- SESM web applications deployed in DESS mode
- RDP

If these applications are installed on the same machine, the same config.xml file applies to both of them. If the applications are installed on different machines, the DESS component is installed with each of them, and each config.xml file can contain different attribute values.

The config.xml file for DESS contains the following MBean:

- Directory—The Directory MBean configures security, location, logging, and caching attributes for executing classes in the Dess and Auth APIs.

Table 4-8 explains the configurable attributes in this MBean.

Table 4-8 Attributes in the Dess-Auth MBean Configuration File

Object Name	Attribute Name	Explanation
Directory MBean	factory	The full class name of the JNDI connection factory.
	poolSize	The number of active connections allowed to the LDAP server used for authorization.
	URL	The URL of the LDAP server used for authorization.
	principal	The name used when connecting to the LDAP server.
	credentials	The credentials (such as password) used for connecting to the LDAP server.
	context	The default LDAP context used for LDAP operations.
	alwaysGetAllAttributes	If set to true then all the attributes of an LDAP entry are returned for every query.
	traceFileName	The name of the directory log file.
	traceLevel	Should be one of: NONE, ERROR, BRIEF, VERBOSE, or DEBUG.
	printTraceToConsole	If set to true, the application sends trace messages to the console as well as writing them into the log file.
	stackTrace	If set to true, print a stack trace with each trace message.
	cacheMaxObjects	Specifies the maximum number of software objects to hold in the cache. Objects represent subscribers, services, privileges, roles, and so on. When the cache contains cacheMaxObjects, old objects are deleted from cache, regardless of available cache space. Set this value high to allow the available cache space to be the determining factor for cache management. Installed default: 50000

Table 4-8 Attributes in the Dess-Auth MBean Configuration File (continued)

Object Name	Attribute Name	Explanation
Directory MBean	cacheMinFreeMem	Specifies the percentage of Java virtual memory that must remain available (that is, not used by the cache) after the application is loaded into memory. You can calculate the specific amount of memory available for the cache as follows: `cacheSize = (JavaVM - applCodeSize) * (100% - cacheMinFreeMem)` Where: *JavaVM* is the maximum virtual memory size specified at application startup time with the jvm argument. The installed startup scripts use the following values: • The startNWSP script uses 64 MB • The runrdp script uses 20 MB *applCodeSize* is the application size. The NWSP is approximately 18 MB. *cacheMinFreeMem* specifies the percentage of Java virtual memory that must remain available after the application is loaded into memory. The installed default value is 10. If NWSP and RDP applications are installed on the same machine, the same cacheMinFreeMem attribute value applies to both applications. For example, using all of the installed default values, the *cacheSize* for the NWSP application is 90% of 14 MB, or 12.6 MB: *cacheSize* = (32 MB - 18 MB) * (100% - 10%) Installed default: 10
	cacheSessionTimeout	Specifies the timeout of inactive client sessions in seconds. Installed default: 600
	cacheExpireInterval	Specifies the interval in seconds after which the cache attempts to expire objects. **Note** Do not set this attribute to 0. A value of 0 causes *every* request to go to the directory, bypassing caching and any memory storage from a recent request for the same object. A value of 0 would degrade performance substantially. Installed default: 600
	cacheObjectTimeout	Specifies the number of seconds before objects time out. Installed default: 600

Extending the Directory Schema and Installing Initial RBAC Objects

An SESM deployment running in DESS mode requires the following update activities on the LDAP directory:

- Extend the directory schema. These extensions include the dess and auth classes and attributes that will hold the SESM data.. For more information about the extensions, see the *Cisco Distributed Administration Tool Guide*.
- Install initial RBAC objects. Some initial top-level rules and roles must be created in the directory before an administrator can log into CDAT and create additional objects.

The DESS installation process optionally performs these two update activities. If you did not choose these options during the installation, you must do them before running CDAT or an SESM application running in DESS mode.

Note If the SESM components are distributed among different servers, which means that DESS might be installed in more than one location, you only need to perform these update activities one time against the LDAP directory.

To perform these updates after the initial DESS installation, use either of the following procedures:

- Use the installation process to perform the updates by running a custom installation of the DESS component.
- Perform the updates manually using native administration tools and commands.

Using a Custom Installation to Update the Schema and Install RBAC Objects

To use the custom installation process to extend the directory schema and install initial RBAC objects, follow these procedures:

Step 1 Make sure the LDAP directory server is running.

Step 2 Make sure you know the following user IDs and passwords:

- A user ID and password that allows you to update the directory schema
- A user ID and password that allows you to update the container (organization and organizational unit) that you created for SESM data.

Step 3 Execute the SESM installation program on a server that has network access to the LDAP directory.

Step 4 When the installation program prompts for setup type, choose **Custom**.

Step 5 When the installation program prompts for the components to install, choose **DESS**.

Step 6 When the installation program prompts for directory connection information, provide correct information to access the directory. This includes the names of the organization and organizational unit you created to hold the SESM data.

Step 7 When the installation program displays the options, click the **Update schema** and **Install RBAC** check boxes.

Using LDIF Commands to Update the Directory Schema

To use LDIF commands to manually update the directory, follow these procedures:

Step 1 Make sure the LDAP directory server is running.

Step 2 Make sure you have a user ID and password for the directory that allows you to update the schema.

Step 3 Obtain the required updates from the following location under your installation directory. Choose NDS or Netscape, depending on the LDAP directory you are using:

```
dess-auth
    schema
        NDS
        Netscape
```

You need to apply the contents of all of the ldf files found under the NDS or Netscape directories:

```
    authattr.ldf
    authclas.ldf
    dessattr.ldf
    dessclas.ldf
    Policy15.ldf
```

Step 4 Use the **ldapmodify** command to apply all of the preceding files to your directory.

On successful completion, you have applied all of the required updates.

Using Manual Tools to Create Initial RBAC Objects

Some initial RBAC rules and roles must be loaded into the directory before any administrator can log into CDAT to create additional objects. The easiest way to load these top level objects is to allow the installation program to do it. However, you can also obtain them by loading the sample RBAC data files that are installed with DESS or by using your own data generating tool. See the *Cisco Distributed Administration Tool Guide* for information about the initial RBAC objects and loading the sample data.

Configuring Specific Features

Table 4-9 summarizes how to enable or disable some of the major features in an SESM deployment.

Table 4-9 Configuration Requirements for Specific Features

Feature	Configuration Requirements
Single sign-on	**On SESM Host** Edit the following line in the application MBean configuration file (for example, nwsp/config/nwsp.xml): `<Set name="singleSignOn" type="boolean">true</Set>`

Automatic connections	**On SSG Host**
	No action required.
	On SESM Host—RADIUS Mode
	In RADIUS mode, the autoconnect feature is always on, regardless of parameter settings. In RADIUS mode, the SSG always performs automatic service connections for all services marked as auto connect in a subscriber's profile.
	In Subscriber's Profile—RADIUS Mode
	The A attribute in a subscriber's profile marks a service as one that should be automatically connected for the subscriber. For example:
	`user5 Password = "cisco"`
	`Service-Type = Framed-User,`
	`Account-Info = "Ainternet-green"`
	On SESM Host—DESS Mode
	In DESS mode, the SSG performs automatic connections if it has the service list. If SSG does not have the service list, the SESM application can perform the automatic connections. During RDP installation, the Add Services option configures RDP to either:
	• Return a service list to SSG—In this case, RDP includes the subscriber's service list and related information in replies to SSG, and SSG performs automatic connections for services marked for autoconnection in the subscriber's profile.
	The service information consumes memory on the SSG host.
	• Not return a service list to SSG—In this case, SSG cannot perform automatic connections. The advantage to this configuration is that it saves memory on the SSG host.
	In this case, you can configure the SESM application to perform automatic connections. The following line in the application MBean configuration file (for example, nwsp/config/nwsp.xml) controls whether the SESM web application performs automatic connections:
	`<Set name="autoConnect" type="boolean">false</Set>`
	Change the value to `true` to enable automatic connections by the SESM web application.
	To change the setting of the RDP service list option, either reinstall RDP or edit the configuration files to enable the correct set of packet handlers. See Appendix E, "RDP Packet Handlers," for information about the packet handlers that are used in the various configurations.
	In Subscriber's Profile—DESS Mode
	See the *Cisco Distributed Administration Tool Guide* for instructions about marking services for autoconnection in subscriber profiles.

Table 4-9 Configuration Requirements for Specific Features (continued)

<table>
<tr><th>Feature</th><th>Configuration Requirements</th></tr>
<tr><td>Application Interface Options</td><td>

On SESM Host

Edit the following lines in the application MBean configuration file (for example, nwsp/config/nwsp.xml):

```
<Put name="useIcons" type="boolean">TRUE</Put>
<Put name="confirmAtServiceLogon" type="boolean">FALSE</Put>
<Put name="confirmAtServiceLogoff" type="boolean">TRUE</Put>
<Put name="confirmAtAccountLogoff" type="boolean">TRUE</Put>
<Put name="sessionTimeOut" type="String">7200</Put>
```

</td></tr>
<tr><td>Captive portal</td><td>

On SSG Host

Enable the TCP redirect feature using the http-redirect Cisco IOS commands.

```
ssg http-redirect group captive-portal-app1 server 10.1.2.50 80
ssg http-redirect unauthorized-user group captive-portal-app1
```

Note The format of the http-redirect commands might change in the next release of Cisco IOS.

On SESM Host

To enable the captive portal application, choose the Run Captive Portal option when you install the NWSP application. This option sets the captiveportal.home Java system property in the generic startup script.

To disable captive portal, edit the generic startup script (for example, jetty/bin/start.sh) and remove the captiveportal.home system property.

To change the name of the captive portal application being called, edit the third argument in the Call element in the *container* MBean configuration file (for example, jetty/config/nwsp.jetty.xml):

```
<!-- Captive portal web application -->
    <Call name="addWebApplication">
      <Arg></Arg>
      <Arg>/</Arg>
      <Arg><SystemProperty name="install.root" default="."/>/captiveportal/docroot</Arg>
      <Arg><SystemProperty name="jetty.home" default="."/>/config/webdefault.xml</Arg>
      <Arg type="boolean">FALSE</Arg>
    </Call>
```

To configure the SESM web application to which the captive portal application redirects subscribers, edit the following element in the *application* MBean configuration file (for example, nwsp/config/nwsp.xml):

```
<Configure name="com.cisco.aggbu:name=captiveportal">
<Set name="captureToURL">http://localhost:80/decorate/pages/home.jsp</Set>
</Configure>
```

</td></tr>
<tr><td>Walled garden</td><td>

In SSG

Enter Cisco IOS **ssg** commands. For example:

```
>ssg open-garden opengarden-xyz.com
>local-profile opengarden-xyz.com
> attribute 26 9 251 "R10.1.1.0;255.255.255.255"
> attribute 26 9 251 "D10.1.1.10"
> attribute 26 9 251 "Oxyz.com;zap.com"
```

In SESM

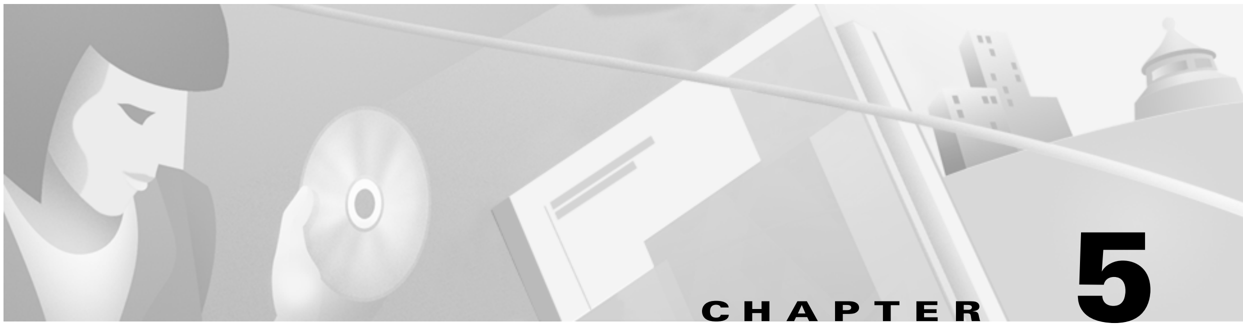
Create JSPs that implement the desired interface. See the *Cisco Subscriber Edge Services Manager Web Developer Guide* for information.

</td></tr>
</table>

Table 4-9 Configuration Requirements for Specific Features (continued)

Feature	Configuration Requirements
Retail pages and service ads	**In SSG** No configuration required. **In SESM** Create JSPs that implement the desired interface. See the *Cisco Subscriber Edge Services Manager Web Developer Guide* for information.

Host Key	**In SSG**
	Enable the SSG host key feature using the Cisco IOS **ssg port-map** commands.
	`ssg port-map enable`
	`ssg port-map source ip loopback 0`
	`ssg port-map destination range` *`lowPort`* `to` *`highPort`* `ip` *`SSDaddress`*
	Disable the host key feature using the following command:
	`ssg port-map disable`
	In SESM
	Edit the BUNDLE_LENGTH attributes in the application MBean configuration file (for example, nwsp/config/nwsp.xml):
	`<Call name="setGlobalAttribute"><Arg>BUNDLE_LENGTH</Arg><Arg>0</Arg></Call>`
	In the SSG MBean, the BUNDLE_LENGTH attributes must match the bundle lengths specified on the SSG side.
	Note A BUNDLE_LENGTH of zero indicates that host key is not being used.



CHAPTER 5

Running SESM Components

This chapter describes how to start and stop Cisco Subscriber Edge Services Manager (SESM) components. The chapter contains the following topics:

Starting the NWSP Application and the Jetty Server

The NWSP application is a J2EE web server application that runs in a Jetty server container. The startup script for NWSP starts both a Jetty server instance and a NWSP application.

Start NWSP using the following script and optional command-line argument:

Platform	Script
Solaris	`jetty/bin/startNWSP.sh [-mode mode]`
Windows NT	`jetty\bin\startNWSP.cmd [mode]`

Valid values for *mode* are Demo, RADIUS, or DESS.

If the mode option is included on the command line, it overrides the default mode specified in the SSD MBean in the nwsp.xml file. If you switch modes using this option, you must make sure that all other configuration parameters are aligned with the mode that you choose. The mode option provides the capability to switch easily between a fully configured deployment (RADIUS or DESS mode) and the demonstration deployment (Demo mode).

Starting RDP

RDP is a Java 2 application that uses the Cisco ConfigAgent and JMX server. It does not use the J2EE HTTP server, and therefore does not have startup files in the Jetty server's bin directory.

Start RDP with the following script:

Platform	Script
Solaris	`rdp/bin/runrdp.sh`
Windows NT	`rdp\bin\runrdp.cmd`

Starting CDAT

CDAT is a J2EE application. The startup script for CDAT is in the Jetty server's bin directory. This startup script calls the same generic startup script used by the SESM web applications.

Start CDAT with the following script:

Platform	Script
Solaris	`jetty/bin/startCDAT.sh`
Windows NT	`jetty\bin\startCDAT.cmd`

Explanation of the NWSP and CDAT Startup Scripts

When you start the NWSP application or CDAT, you are executing two scripts:

- Application-specific startup script—Sets application-specific parameters and calls the generic script
- Generic startup script—Infers additional parameters and starts the SESM web application and the Jetty server.

Both scripts are located in:

```
jetty
    bin
```

You should create an application-specific startup script in this same `bin` directory for customized SESM web applications.

Application-Specific Startup Scripts

The application-specific startup scripts are startNWSP and startCDAT. These scripts set the following variables:

- application name—Identifies the application, either NWSP or CDAT. If you create a customized application, provide the name that identifies your application. See the "SESM Application Names" section on page 6-3 for information about using a new application name value.
- port number— Identifies the port that the application's container (the web server) will listen on.

The installation program updates the application startup script with the port number that you provide during the installation time. To change the port number after installation, edit the startup script. The default values displayed by the installation program are 8080 for NWSP and 8081 for CDAT.

The port number must be unique on the server machine. If multiple SESM applications are running simultaneously on the same server machine, make sure each one listens on a different pertussis caveat applies whether you are running two instances of the same application or two different applications.

Generic Startup Script

The generic startup script derives two other port numbers from the application port number:

- It derives a management console port number as follows.

  ```
  application port + 100
  ```

  For example, if you are using the default application port of 8080 for NWSP, the management console port for NWSP is:

  ```
  8080 + 100 = 8180
  ```

- It derives a secure socket listener (SSL) port as follows:

  ```
  application port - 80 + 443
  ```

  Starting with the default application port value of 8080, the default SSL port is:

  ```
  8080 - 80 + 433 = 8443
  ```

The generic startup script does the following:

- Accepts the variables passed to it from the application startup script
- Sets additional variables, based on the expected (installed) directory structure. For example, it infers the location of the configuration files.
- Starts the SESM web application.

Java System Properties in Startup Scripts

Table 5-1 describes the java system properties that are set by the generic startup script and how the assigned values are derived. The table describes the following lines, which are located at the end of the generic startup script:

```
$JAVA -Xmx64m \
  -classpath $CLASSPATH \
  -Djetty.home=$JETTYDIR \
  -Dapplication.home=$APPDIR \
  -Dapplication.log=$LOGDIR \
  -Dapplication.portno=$PORTNO \
  -Dmanagement.portno=$MGMTPORTNO \
```

Table 5-1 Java System Properties in the Startup Script

System Property and Variable Name	Explanation	Installed Values in the Start Script
jetty.home=$JETTYDIR	jetty.home is the container's directory name. The startup script sets $JETTYDIR to a subdirectory named jetty under the installation directory.	*`installDir`* `jetty`
application.home=$APPDIR	application.home is the application's directory name. The startup script sets $APPDIR to a subdirectory named *applicationName* under the installation directory. The startup script infers the installation directory from the location of the start script itself. The *applicationName* parameter is passed from another script. (startNWSP.sh, for example).	*`installDir`* `nwsp`

application.log=$LOGDIR	application.log is the location of all log files created for this application.
	The startup script sets $LOGDIR differently according to the platform:
	• On Solaris, $LOGDIR is the logs directory under the application directory in the install directory. For example: `installDir/nwsp/logs`
	• On Windows NT, $LOGDIR is *userTemp\application*\logs where *userTemp* is the administrator's temporary directory. For example: `temp\nwsp\logs`
		`installDir`
		`nwsp`
		`logs`

application.portno=$PORTNO	application.portno is the port that the SESM web application (or CDAT) listens on for HTTP requests from subscribers. The startup script sets $PORTNO to the portNo parameter passed from another script (startNWSP.sh, for example).	Specified during installation. The default is 8080 for NWSP and 8081 for CDAT.
management.portno= $MGMTPORTNO	management.portno is the console port that displays the current values for all attributes in all of the MBean configuration files.	The startup script sets $MGMTPORTNO to $PORTNO + 100.

Stopping Applications

This section describes how to stop SESM applications. It includes the following topics:

- Stopping SESM Applications on Solaris, page 5-5

- Stopping SESM Applications on Windows NT, page 5-5

Stopping SESM Applications on Solaris

To stop SESM web applications and their J2EE containers on Solaris, execute the installed stop scripts. None of the scripts take arguments. Table 5-2 lists the script names and locations.

Table 5-2 SESM Stop Scripts on the Solaris Platform

Application	Stop Script Location and Name on Solaris Platforms
NWSP and Jetty	jetty/bin/stopNWSP.sh
CDAT and Jetty	jetty/bin/stopCDAT.sh
RDP	rdp/bin/stoprdp.sh

Stopping SESM Applications on Windows NT

To stop SESM web applications and their J2EE containers on Windows NT platforms, you can:

- Open the Task Manager window, select the appropriate task, and click the **End Task** button. If you are prompted again, click the **End Now** button.
- If you added the application as an NT service, you can use the Services window to stop the service. Open **Control Panel > Services** or **Control Panel > Administrative Tools > Services** and select the service you want to stop. Use the menu commands on the Services window to stop the selected service.

Adding and Removing Services on Windows NT

On a Windows NT platform, you can add your applications to the list of Windows NT services. When the application is a service, it appears in the **Services** window accessed from **Control Panel > Services** or **Control Panel > Administrative Tools > Services** You can start and stop any service from this window. Also, you can optionally configure a service to start automatically when the system reboots.

The SESM installation program provides services scripts with the NWSP, CDAT, and RDP applications. The command usage is the same for all of the services scripts:

- `scriptName -i` installs the application as a service so that it can be managed from the Services window
- `scriptName -h` displays the command usage
- `scriptName -r` removes the application from the Services window

Table 5-3 lists the names and locations of the scripts that add and remove services.

Table 5-3 Scripts for Adding and Removing Services on Windows NT

SESM Application	Services Script Location and Name	Default Service Name
RDP	rdp\bin\rdpsvc.cmd	RDP Application
CDAT	jetty\bin\cdatsvc.cmd	CDAT Web Application
NWSP	jettybin\nwspsvc.cmd	NWSP Web Application

Memory Requirements for the NWSP Application

The total java virtual memory requirements for an SESM web application depends on several factors:

- Number of subscribers concurrently logged in
- Number of subscribed services
- Rate of new logins—The login rate affects transitory memory usage.

Table 5-4 shows SESM memory requirements in various scenarios. The table includes two memory columns for each scenario.

- The Memory Required for Logins is the total memory required for the successful login and authentication of all users, at the indicated login rates.
- The Memory Used After Logins is the actual memory used to support the SESM session with connections to the indicated services after logins are completed.

For SESM Release 3.1(1), Cisco supports a maximum of 10,000 concurrently logged in subscribers in RADIUS mode, and 5,000 concurrently logged in subscribers in DESS mode. We have verified the memory requirements in Table 5-4 for one SESM application instance. It is possible, given more memory, to support larger numbers of users.

Table 5-4 SESM Memory Requirements

	RADIUS Mode				DESS Mode			
	Three Services: 1 Passthrough 1 Proxy 1 Tunnel		3 Services: 1 Passthrough (Auto) 1 Proxy 1 Tunnel		3 Services 1 Passthrough 1 Proxy 1 Tunnel		3 Services: 1 Passthrough (Auto 1 Proxy 1 Tunnel	
Number of Subscribers Logged On[1]	Memory Required for Logins (MB)	Memory Used After Logins (MB)	Memory Required for Logins (MB)	Memory Used After Logins (MB)	Memory Required for Logins (MB)	Memory Used After Logins (MB)	Memory Required for Logins (MB)	Memory Used After Logins (MB)
2000	32	15.3	32	17.7	64	48	64	51
4000	48	28.6	64	30	112	100	112	97
6000	80	42.6	96	40	192	145	208	139
8000	96	52.2	112	57.3				
10000	128	67.4	144	62				

[1] The information in this table was obtained using the following login rates:

.RADIUS mode—20 subscribers per second

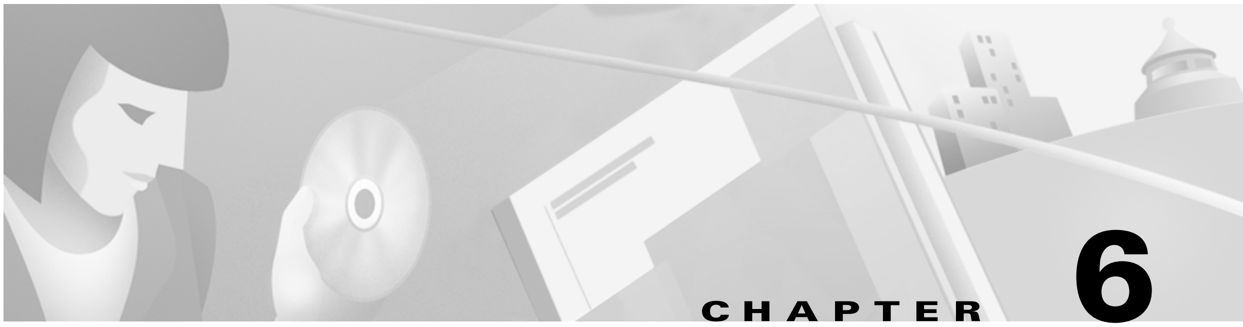
.DESS Mode—10 subscribers per second

The generic startup script sets the amount of Java virtual memory reserved for use by the SESM web application (NWSP). The virtual memory setting is an argument to the java command, which is located at the end of the script, as follows:

```
$JAVA -Xmx64m
```

The installed start script sets the java virtual memory to 64MB. Consider changing the default value in the following circumstances:

- If you are running the Demo exclusively on a machine running other applications, you might want to decrease the memory size.
- Increase the memory if the number of users simultaneously logged on increases. Symptoms of insufficient memory are:
  - Out of memory exceptions
  - Messages stating that the web server is unavailable



CHAPTER 6

SESM Applications

This chapter describes how to access the Cisco Subscriber Edge Services Manager (SESM) application from a web browser. It also describes configuration procedures for a customized application. The chapter contains the following topics:

- Supported Browsers, page 6-1
- Accessing the NWSP Application, page 6-1
- Customizing the NWSP Application, page 6-2

Supported Browsers

Subscribers can access SESM web applications, such as NWSP, with the following browsers:

- Netscape Release 4.x
- Internet Explorer Release 5.x

Accessing the NWSP Application

To access the NWSP application, follow these procedures:

Step 1 Make sure the NWSP application is running.

Step 2 Start a browser.

Step 3 Go to the NWSP URL, which is the host and port number that you specified during the installation:

http://*host*:*port*

Default values are:

```
http://localhost:8080
```

If captive portal is configured, the port value should be 80, as follows:

```
http://localhost:80
```

Note If the captive portal feature is configured for the SESM application and the TCP redirect feature is configured on SSG, subscribers are redirected to the captive portal application without entering an URL.

Step 4 When the NWSP logon page appears, log in using a valid user ID and password. A valid user ID and password is defined in user profiles as follows:

- In RADIUS mode, the user profile must exist in the RADIUS server database. See Appendix D, "Configuring RADIUS," for more information.
- In DESS mode:
  - If RDP is configured in Proxy mode, the user profile must exist in the RADIUS server database that the RDP is proxying to.
  - If RDP is configured in normal (non-Proxy) mode, the user profile must exist in the LDAP directory in the DESS-specified format. See the *Cisco Distributed Administration Toolkit Guide* for more information.

Note See Chapter 2, "Demo Quick Start," for instructions on logging on to the NWSP application running in DEMO mode.

Customizing the NWSP Application

The Cisco Service Selection Dashboard (SSD) Release 3.0(1) is a collection of components for creating specialized Java 2 Platform, Enterprise Edition (J2EE) web server applications. J2EE provides a framework for using various Java-based components to develop multi-tiered applications. The multi-tiered application (as opposed to the two-tiered client server application) provides many opportunities for isolating and controlling functional pieces of a large application. For more information about the J2EE development platform, see:

http://java.sun.com/j2ee/

SESM Application Definition

A Cisco SESM application consists of the following:

- SESM servlets and classes—The SESM API defines the SESM classes, including the configurable MBeans, used to implement the application functionality.
- ConfigAgent—The ConfigAgent is a Cisco developed MBean that configures other MBeans. It configures MBeans that are registered with the JMX server by applying parameter values from .xml files. Because .xml files are easily maintained and changed by system administrators, applications that use ConfigAgent are highly configurable without recompiling. Chapter 4 in this guide explains all of the configurable parameters in all of the MBeans.
- JavaServer Pages (JSPs)—JSPs offer a way to deliver dynamic content in web pages. Web developers at the deployment site can control their subscriber's SESM experience through the JSPs. The *Cisco Subscriber Edge Services Manager Web Developer Guide* provides instructions for defining and compiling JSPs.
- Images—Images are used by the JSPs and control the look and feel and branding aspects of an SESM application. The *Cisco Subscriber Edge Services Manager Web Developer Guide* provides instructions for changing images and incorporating them into the JSPs.

SESM Application Names

The SESM application name that you use for a customized application is arbitrary, but it must match in all of the following places:

- The name of the application-specific subdirectory under the installation directory. For example, the directory that holds all application specific information for the NWSP application is:

  ```
  <installDir>nwsp
  ```

- Application parameter inside the application startup script. In the installed scripts, the application name is hardcoded on the line that calls the generic start script. For example, for the NWSP application on Windows NT, the call line is:

  ```
  call "%SCRIPTDIR%start.cmd" nwsp %PORTNO%
  ```

- Name of the application's configuration file in the `jetty` subdirectory. For example, for the NWSP application, the configuration file name is:

  ```
  nwsp.jetty.xml
  ```

An application name in the startup script tells the ConfigAgent which configuration file to open. The application name is passed to ConfigAgent by the application startup scripts. The application name might also be used in other ways. For example, you can configure the parameter that defines the Jetty Server log file name to incorporate the application name in the log file name.

Configuring Customized SESM Applications

Application developers at your site might make changes to the delivered NWSP sample application, producing a customized application. Customized applications require their own set of configuration files, although the files might be very similar to those provided for the sample application.

To configure a customized SESM application that will run on the Jetty server, follow these steps:

Step 1 Create a configuration file for the new application in the container's config directory. You can copy the `nwsp.jetty.xml` file and appropriately rename it. For example:

```
jetty
    config
        newApplication.jetty.xml
```

Step 2 Edit the new file, enabling and disabling features as described in the "Configuring an SESM Web Application" section on page 4-16.

Step 3 Create a startup script for the new application by copying the `startNWSP` script and appropriately renaming the copy. For example:

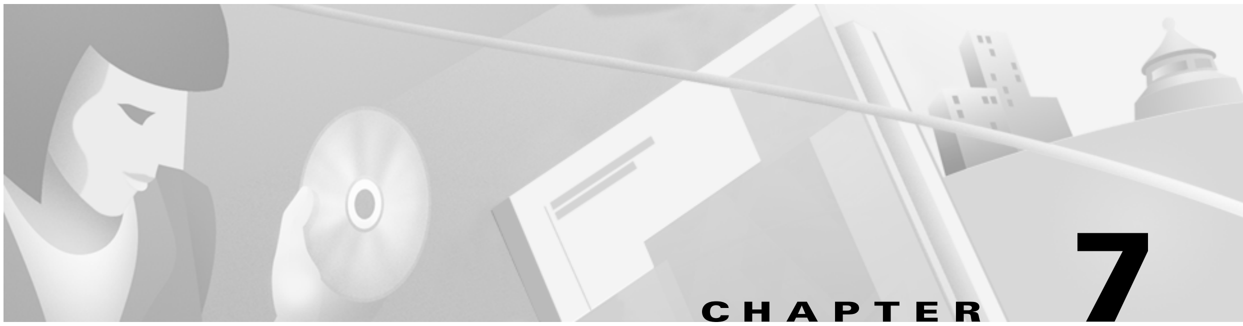
```
jetty
    bin
        startNewApplication
```

Step 4 Edit the new file, changing the application name and the port number parameters. See the "Explanation of the NWSP and CDAT Startup Scripts" section on page 5-2 for more information.

Step 5 Copy the nwsp directory structure, and rename the nwsp objects appropriately. For example, copy:

```
nwsp
    config
        nwsp.xml
    docroot
    docs
```

Step 6 See the *Cisco Subscriber Edge Services Manager Web Developer Guide* for information about customizing the JSPs, images, and other components. That guide also describes how to update the docroot folder, recompile affected components, and edit the web.xml file.



CHAPTER 7

Troubleshooting Installation and Configuration

This chapter details troubleshooting for the Cisco Subscriber Edge Services Manager (SESM). It includes the following topics:

- Troubleshooting Aids, page 7-1
- Troubleshooting Hints, page 7-3

Troubleshooting Aids

This section describes some facilities that might be useful in troubleshooting SESM installation and configuration problems. It includes the following topics:

- Logging and Debugging Mechanisms, page 7-1
- Java Command Line Options, page 7-2
- SESM Management Console, page 7-2
- Obtaining License and Version Information, page 7-3

Logging and Debugging Mechanisms

This section describes the logging and debugging options available to help troubleshoot SESM. The logging and debugging mechanisms are MBeans which are configured in the MBean configuration files. By changing the configuration of the logging and debugging mechanisms, you can change the amount of detail reported and specify message filtering.

Using SESM Application Log Files

You can use the SESM application's log files to troubleshoot problems. Two of the log files have debugging mechanisms that you can configure along with the logging features.

An SESM web application and its Jetty container write to the following log files:

- HTTP request log—This log file records all incoming HTTP requests. You can use this log file to analyze volume and traffic patterns for the web server.

  The default name for this file is *date*.request.log.

  See the HttpServer MBean description in the "Container Attributes" section on page 4-9 for information on configuring this log, including file name and retention period.

- Jetty application log—This log file records logging and debugging messages. The logging messages record the startup of the Jetty server and all ongoing activity, such as errors trapped by the Jetty server and HTTP errors. If the SESM application fails to start, look at this log. Make sure you monitor this log file for illegal HTTP requests that might indicate attempts to subvert the web server. If you enable debugging, the log file also includes more detailed debugging messages.

  The default name for this file is *date*.jetty.log.

  See the Log MBean and Debug MBean descriptions in the "Container Attributes" section on page 4-9 for information on configuring this log, including file name, retention period, and contents.
- SESM application log—This log file records logging and debugging messages. The logging tool logs SESM application activity. The debugging mechanism produces messages useful to developers in debugging applications.

  The default name for this file is *date*.application.log.

  See the Logger MBean in the "SESM Application Attributes" section on page 4-17 for information on configuring this file, including its file name and retention period, whether debugging is turned on or off, and the content of logging and debugging messages.

Log File Locations

For each SESM application, all of these log files are located in the same directory. The *application.log* Java system property is used to specify the directory location. See Table 5-1 on page 5-4, "Java System Properties in the Startup Script", for a description of how the start script sets *application.log*, the default values of *application.log*, and how to change this default.

Java Command Line Options

When you execute a startup script that includes the java command, you can specify any Java option on the command line. To specify Java options, use -jvm as an option on the command line. For example, you can add the following option to the command line when you execute the SESM application startup script:

```
-jvm -Djava.compiler=NONE
```

SESM Management Console

The Sun example JMX server, which is the JMX server installed with the Jetty component in the SESM installation package, includes a JMX HTML adaptor. SESM uses the adaptor to produce a management console that shows the current value of all MBean attributes in all of the MBean configuration files.

Note This JMX HTML adaptor is not production quality. For example, configuration changes made using this console are not persistent. You should remove it from your configuration files before transitioning the SESM application to public use. See the "ManagementConsole" section on page 4-18 for information about configuring and removing this adaptor.

You can access the SESM management console on a web browser at the following URL:

http://*SESMserver*:*managementPortNumber*/

Where:

SESMserver—Host name or IP address of the workstation where SESM is installed.

managementPortNumber—The port configured in the HtmlAdaptorServer MBean in the application configuration file. The default management port number used by the SESM installation program is:

```
applicationPortNumber + 100
```

For example, for NWSP, the installer uses a default application port number of 8080 and a corresponding management port number of 8180.

Management Console User Name and Password

Before you gain access to the management console, you must enter a valid user name and password. Enter the values that match the values in the ManagementConsole MBean in the application's configuration file. See the "ManagementConsole" section on page 4-18 for more information.

Obtaining License and Version Information

If you purchased SESM, your license number is available on the License Certificate shipped with the product. If you have not purchased SESM, you can install an evaluation copy of the software without a license number. An evaluation installation provides full software functionality. Although the evaluation options do not have an expiration period, you must obtain a license before deploying SESM in a production environment.

The installation program records the license number and the software version you installed in the licensenum.txt file under the installation directory.

Troubleshooting Hints

This section contains some hints that might help you identify and fix problems in SESM. The hints are divided into the following topics:

- JRE and JDK Troubleshooting, page 7-3
- Installation Troubleshooting, page 7-5
- Configuration File Location Troubleshooting, page 7-6
- SESM Configuration Troubleshooting, page 7-6
- RADIUS Configuration Troubleshooting, page 7-7
- SSG Configuration Troubleshooting, page 7-7

JRE and JDK Troubleshooting

If the installer does not find an appropriate JRE, it installs the bundled JRE Version 1.2.2.

This section contains the following topics:

- Warning and Error Messages after JRE Installation, page 7-4
- Searching for an Existing JDK or JRE, page 7-4

- Using a Pre-installed JRE or JDK, page 7-4
- Recompiling a Customized JSP, page 7-5

Warning and Error Messages after JRE Installation

The JRE installation might produce warning messages and nonfatal error messages. These messages are expected and normal.

- The warning message states that JSPs will not be compiled. You do not need to recompile JSPs to run the NWSP application.

  If you are a Web developer expecting to write new JSPs or change the NWSP JSPs, you must load the Java Development Kit (JDK). To obtain a recent JDK, go to:

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

- The nonfatal JIT relocation error message is the result of a problem within the bundled JVM obtained from Sun Microsystems. It does not affect SESM operation. You can ignore this message and all supporting information.

Searching for an Existing JDK or JRE

The installer does the following when searching for a valid JDK or JRE:

1. It searches for a JDK Version 1.2.2 that is already installed.
2. Failing that, it searches for a JRE Version 1.2.2 or later that is already installed.
3. Failing that, it installs and uses the bundled JRE Version 1.2.2.

In some cases, even though a JRE is installed, the installer may not find it or finds a different JRE.

On Windows NT, the installer looks in the NT Registry for the location of a JDK or JRE. It uses Java Version 1.2.2 in preference to Version 1.3.

On Solaris, the installer looks in the following well-known locations before installing the bundled JRE:

usr/jre	/opt/jre	/usr/jre1.3	/opt/jre1.3
/usr/jre1.2.2	/opt/jre1.2.2	/usr/jre1.3.0	/opt/jre1.3.0
/usr/java1.2	/opt/java	/usr/java1.3	/opt/java
/usr/java	/opt/java1.2	/usr/java	/opt/java1.3
/usr/java1.2.2	/opt/java1.2.2	/usr/java1.3.0	/opt/java1.3.0
/usr/jdk	/opt/jdk	/usr/jdk	/opt/jdk
/usr/jdk1.2	/opt/jdk1.2	/usr/jdk1.3	/opt/jdk1.3
/usr/jdk1.2.2	/opt/jdk1.2.2	/usr/jdk1.3.0	/opt/jdk1.3.0

Using a Pre-installed JRE or JDK

On either platform, you can specify the location of a pre-installed JRE or JDK by starting the installation process on a command line and specifying the javahome parameter, as follows:

```
installImageName -is:javahome location
```

Where:

installImageName is the name of the SESM downloaded image.

location is the path name for the JRE or JDK.

Recompiling a Customized JSP

If you do not see changes that you make to a JSP, follow these procedures:

Step 1 Install a JDK (Version 1.2.2 or later).

Step 2 Edit the application start script so that it uses the JDK, rather than the JRE. (For example, edit .../jetty/bin/start.sh).

Step 3 Ensure that JDK_HOME points to the directory into which you installed the JDK.

Step 4 Stop the SESM application.

Step 5 Change directories to the application's WEB-INF directory. For example, enter:

```
cd SESM_install/nwsp/docroot/WEB-INF
```

Step 6 In the WEB-INF directory, back up the web.xml file by renaming it. For example, enter:

```
cp web.xml web.xml.bak
```

Step 7 In the WEB-INF directory, copy the web.recompile.xml file over web.xml. For example, enter:

```
cp web.recompile.xml web.xml
```

Step 8 Restart the SESM application.

The installed web.xml file points to precompiled versions of the JSPs. It does *not* reference the JSPs in .../nwsp/docroot. Thus, changing the JSPs in docroot has no effect if you use the installed web.xml file.

The web.recompile.xml file references the JSPs in .../nwsp/docroot, rather than using the precompiled JSPs.

Installation Troubleshooting

This section describes some problems that you might encounter during installation.

No X Server for a Solaris Installation

To install SESM on a Solaris server with no X server, use the Silent or Console installation modes.

Incorrect Permissions

The SESM installation program writes to parts of the file system or Windows registry that are only accessible to a privileged user (that is, root on Solaris, or a member of the Administrators group on Windows NT). An SESM installation must be performed by a privileged user who has access to these resources. Otherwise, the outcome of the installation is unpredictable.

Files Not Found

If you receive Java error messages indicating missing files in system level directories (for example, /var, on Solaris), you do not have correct permissions to perform the installation. See the preceding "Incorrect Permissions" section.

Configuration File Location Troubleshooting

The SESM installation program places the J2EE web server and SESM configuration files in the correct directories as defined in the startup scripts. If the configuration files are moved for any reason, then you must edit the web.xml file to reflect the new locations.

SESM Configuration Troubleshooting

If the SESM software is installed correctly, and all of the configuration files are in the proper location, but the SESM web application does not function, then examine the configuration values in the SESM application's MBean configuration file (for example, nwsp/config/nwsp.xml).

Communication with SSG

If the SSG port number or shared secret specified in the SESM application's MBean configuration file does not match actual SSG configuration (as performed on the SSG host), the SSG cannot see the SESM requests or is unable to decrypt the requests because the shared secret does not match. When the shared secret does not match, the SSG returns an Access Reject message.

For more information on SSG configuration, see Appendix B, "Configuring the SSG."

Communication with RADIUS Server

If incorrect IP addresses or port numbers are specified in the SESM application's MBean configuration file for the primary and secondary RADIUS servers, the RADIUS servers cannot see the SESM requests.

If the IP addresses and port numbers are correct, the RADIUS server returns an Access Reject when either of the following errors is present:

- The shared secret specified for the RADIUS server in the application's *appl*.xml is not correct.
- The SESM web application is not properly configured as a RADIUS client.

For more information on RADIUS configuration, see Appendix D, "Configuring RADIUS."

Out of Memory Exceptions

Out of memory exceptions might indicate that there is not enough Java virtual memory reserved to handle the number of users currently logged on.

The generic startup script sets the Java virtual memory (VM) size to 64MB. To change this value, stop the application, edit the generic start script (start.sh or start.cmd), and restart the application.

Web Server Unavailable

Messages stating that the web server is unavailable might indicate that there is not enough Java virtual memory reserved to handle the number of users currently logged on. Follow the instructions in the "Out of Memory Exceptions" section on page 7-6 to increase Java virtual memory.

RADIUS Configuration Troubleshooting

The RADIUS server must be configured to recognize the following two clients:

- SESM web application
- SSG

If either of these configuration items is incorrect, then the RADIUS server sends Access Reject messages in response to all requests. See the "Configuring NAS Clients" section on page D-2 for information on configuring these RADIUS clients.

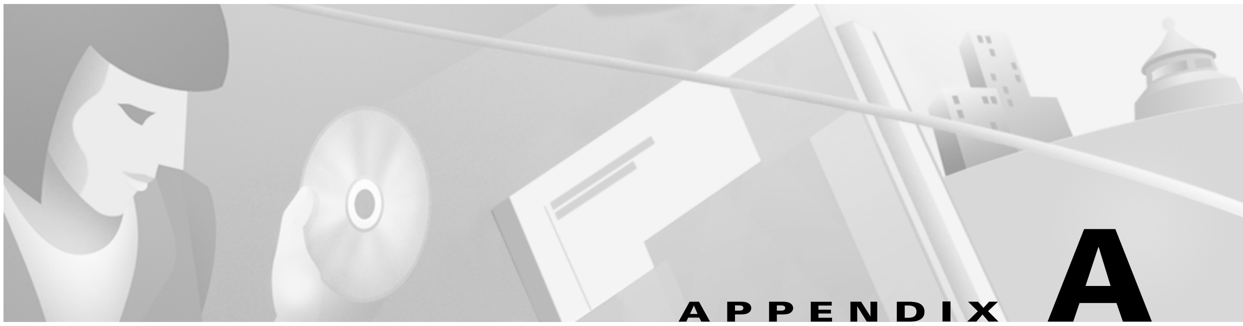
For service profile requests, the password for service and service group profiles must match those defined for the SSG and the SESM application. This password is used in Access Request messages for profiles, where the profile name is the service or service group name and the password is as defined in the following two locations:

- the servicePassword attribute in the AAA section of the SESM application's MBean configuration file
- the service-password parameter for the SSG

SSG Configuration Troubleshooting

The SSG must have a default network location defined, from which the SESM web application is accessible. Otherwise, client requests never reach the SESM application, and the client browser eventually times out.

The SSG must have the radius-helper parameters configured with the correct port numbers and shared secret so that the SSG can see SESM messages and decrypt them. Because the SSG carries out authentication against the RADIUS server, it must also have the correct values defined for the radius-server parameters.



APPENDIX A

Security

This appendix describes the security mechanisms used in a Subscriber Edge Services Manager (SESM) application.

The Cisco SESM:

- Is built using Java technology based on the J2EE specification. As such, it inherits the security features both of the Java language platform and the security framework in J2EE.
- Is a web server-based application, and so must be deployed in a web server that enforces HTTP security.
- Plays a role in authentication for the user, so it must also enforce constraints at this level.

Java Platform Security Description

The following URLs provide a description of Java platform security:

- http://java.sun.com/security/index.html
- For specific Java platforms:
  - http://java.sun.com/products/jdk/1.2/docs/guide/security/
  - http://java.sun.com/products/jdk/1.3/docs/guide/security/
- For training:
  - http://developer.java.sun.com/developer/onlineTraining/Security/Fundamentals/index.html
- For miscellaneous articles:
  - http://developer.java.sun.com/developer/technicalArticles/Security/

HTTP Security Description

HTTP security involves two separate issues:

- Encryption of communications using HTTPS
- Basic and digest access authentication in HTTP1.1 (RFC 2617)

HTTPS Description

HTTPS (Secure Hypertext Transfer Protocol) is HTTP over Secure Sockets Layer (SSL), which are HTTP packets sent as encrypted data. This is the mechanism by which data is securely transmitted over the Internet between a browser client and a server.

SESM implements SSL using the Java Secure Sockets Extension (JSSE). For information about JSSE, go to:

http://java.sun.com/products/jsse/

The J2EE specifications describe an extension framework for the integration of SSL implementations. For implementations other than JSSE, go to:

http://www.phaos.com/e_security/prod_ssl.html

Keytool and Keystore

The SSL part of HTTPS requires a certificate to generate the encryption key. For the Jetty web server bundled with the Cisco SESM, the certificate is named keystore and is found in the /etc directory. The keystore file is created by the keytool utility. For detailed instructions on the use of keytool, go to the following URL:

http://java.sun.com/products//jdk/1.2/docs/guide/security/SecurityToolsSummary.html

The sample keystore functions for nonproduction deployments. However, you must obtain a site-specific certificate for production deployments from VeriSign, Inc. at:

http://www.verisign.com

Though certificates are generally the same in concept, they tend to differ in implementation. Therefore, a degree of certificate manipulation is required to obtain a certificate from a given source to work with a given SSL implementation. For JSSE and the Jetty web server, the required steps are described at:

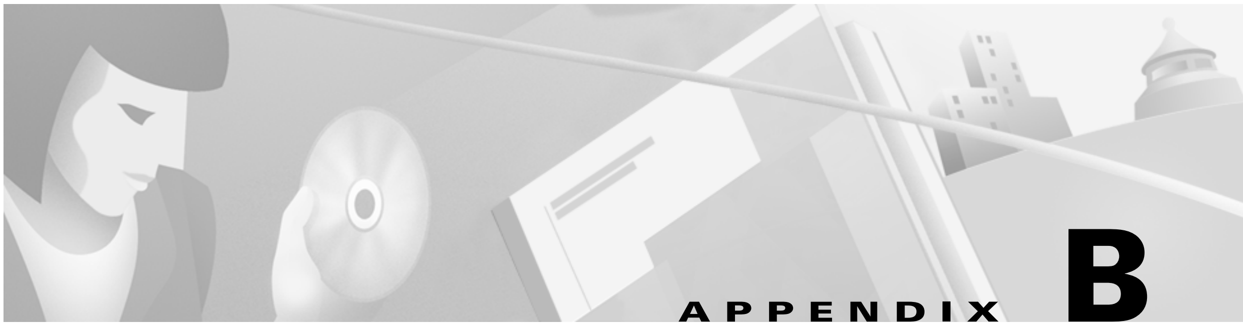
ftp://jetty.mortbay.com/pub/Jetty-dev/webapps/jetty/JsseSSL.html

For other implementations, go to:

http://www.openssl.org

The keystore file is a certificate used for secure sockets layer (SSL) encryption. The SSL implementation shipped with the Cisco SESM is of commercial quality and can use certificates generated by keytool. Keytool resides in the same directory as the JRE.

⚠ **Caution** A keystore is required for deployments that use HTTPS. HTTPS does not function without a valid keystore file. The file included with the installation works, but you should replace it with a keystore valid for your specific deployment.



APPENDIX B

Configuring the SSG

This appendix illustrates some basic steps for configuring the Cisco Service Selection Gateway (SSG) to work with a Subscriber Edge Services Manager (SESM) web application. For a complete description of how to configure SSG, see the following documentation:

- *Cisco 6400 Feature Guide*—This guide includes a chapter that documents SSG features. The online link to this guide is:

  http://www.cisco.com/univercd/cc/td/doc/product/dsl_prod/6400/feat_gd/12_1_5/index.htm

- *Cisco 6400 Command Reference*—This guide includes a chapter that defines SSG configuration commands. The online link to this guide is:

  http://www.cisco.com/univercd/cc/td/doc/product/dsl_prod/6400/commandr/index.htm

- *Cisco 6400 NRP—Release Notes for Cisco IOS Release 12.1(5)DC*—The online link to these release notes is:

  http://www.cisco.com/univercd/cc/td/doc/product/software/ios121/121relnt/6400/rn121dc5.htm

Basic SSG Configuration

This section shows basic procedures for enabling an SSG and configuring it to communicate with a RADIUS server. When following these procedures, replace the sample IP addresses, port numbers, and passwords with values that are appropriate for your configuration.

Step 1 Log on to the NRP.

Step 2 To access enabled mode, enter:

```
en
```

Step 3 To change the configuration, enter:

```
conf t
```

Step 4 To enable the SSG, enter:

```
ssg enable
```

Step 5 To remove a line, enter:

```
no radius-server host 10.3.3.2 auth-port 1647 acctport 1648 0 key cisco
```

Step 6 To add an entry, enter:

```
radius-server host 10.3.3.2 auth-port 1812 acctport 1813 0 key cisco
```

Step 7 To end editing, enter:

```
Ctrl-Z
```

Step 8 To rebuild the configuration, enter:

```
wr t
```

Step 9 To examine the current configuration, enter:

```
show run
```

Step 10 The relevant configuration entries are as follows:

a. To identify the network that the SESM web application is running on, enter:

```
ssg default-network 10.3.3.0 255.255.255.0
```

b. To specify the password to query RADIUS for service profiles, enter:

```
ssg service-password servicecisco
```

c. To configure the RADIUS protocol communication used between SSG and the SESM web application, specify the port on which the SSG is listening as follows:

```
ssg radius-helper auth-port 1812
```

d. To specify the shared secret for password encryption between SSG and the SESM web application, enter:

```
ssg radius-helper key cisco
```

e. To specify the maximum number of concurrent services for a user, enter:

```
ssg maxservice 21
```

f. To configure communication between SSG and the RADIUS server, specify the authentication port, the accounting port, and the secret as follows:

```
radius-server host 10.3.3.2 auth-port 1812 acct-port 1813 key cisco
```

g. To specify the number of RADIUS retries for authentication, enter:

```
radius-server retransmit 3
```

h. To specify the shared secret for password encryption to the RADIUS server, enter:

```
radius-server key cisco
```

Configuring the Host Key Port Bundle Feature on SSG

For the host key port bundle mechanism to operate correctly, the SESM web application must reside in the default network with subscribers (PPP or bridged/routed) connected on downstream interfaces.

Note The host key feature requires Cisco IOS Release 12.2(2)B or later.

To configure the SSG for host key operation, enter the following configuration commands at the terminal configuration prompt on the SSG host:

```
ssg port-map enable
ssg port-map source ip loopback 0
ssg port-map destination range lowPort to highPort ip SESMaddress
```

The **ssg port-map source ip** command configures the IP addresses for use as the IP portion of the host key. Each configured address allows for approximately 4000 host keys, if the default port bundle length of 4 is used. This address becomes the source IP address for all upstream TCP packets from SSG to the SESM web application (and conversely, the destination address for all downstream TCP packets from the SESM web application to the SSG). Although you can explicitly configure these addresses, the safest way to configure them is by using a loopback interface, as shown above, because these IP addresses must be recognized as corresponding to a local interface or loopback.

If you use the interface that is configured to give SSG access to the default network as one of the interfaces in the **ssg port-map source ip** command, that interface cannot also be used as a Telnet interface into the SSG host.

The **ssg port-map destination range** command defines the address and ports of the SESM web application, where:

lowPort is the lowest SESM port

highPort is the highest SESM port

SESMaddress is the IP address of SESM

If there is only one SESM port available, `highPort` should have the value `lowPort + 1`. For example:

```
ssg port-map destination range 10100 to 10101 ip 10.0.3.1
```

Sample SSG Configuration

The following annotated configuration example shows how to implement the following features:

- Enable and configure SSG
- Configure open gardens
- Configure TCP redirection
- Configure communication between SSG and a RADIUS server
- Configure communication between SSG and an SESM web application

Note The following sample SSG configuration is generated on a system running Cisco IOS Release 12.1(5)DC. The syntax for some SSG commands might change in a later Cisco IOS release.

```
#!
#! Last configuration change at 03:16:44 PST Thu May 17 2001 by cisco
#! NVRAM config last updated at 03:02:39 PST Thu May 17 2001 by cisco
#!
#version 12.1
#no service single-slot-reload-enable
#no service pad
#service timestamps debug datetime
#service timestamps log uptime
#no service password-encryption
#!
#hostname agg1-nrp8
```

```
#!
#boot system flash:c6400r-g4p5-mz.121-5.DC.bin
#logging rate-limit console 10 except errors
#no logging console
#aaa new-model
#aaa authentication banner ^C !!! Cisco 6400 NRP8 Service Selection Gateway !!! ^C
#aaa authentication fail-message ^C Unauthorized Access Is Not Permitted ^C
#aaa authentication password-prompt Password:
#aaa authentication username-prompt Username:
#aaa authentication login console local
#aaa authentication ppp default local group radius
#aaa authorization network default local group radius
#aaa accounting update periodic 300
#aaa accounting network default start-stop group radius
#aaa nas port extended
#enable password zeus
#!
#username cisco password 0 cisco
#!
```

The following lines enable and configure SSG to communication with the SESM web application.

```
#ssg enable
#ssg default-network 192.168.2.0 255.255.255.0
#ssg service-password xssg-key
#ssg radius-helper auth-port 1812 acct-port 1813
#ssg radius-helper key cisco
#ssg next-hop download ssg-next-hop xssg-key
#ssg accounting interval 600
#ssg bind service internet FastEthernet0/0/0
#ssg bind service opengarden-aggregation FastEthernet0/0/0
#ssg bind service proxy ATM0/0/0.2
#ssg bind direction downlink ATM0/0/0.301
#ssg bind direction uplink FastEthernet0/0/0
#ssg bind direction uplink ATM0/0/0.2
#ssg bind direction downlink ATM0/0/0.3
```

The following lines illustrate how to configure SSG open gardens:

```
#ssg open-garden opengarden-aggregation
#ssg open-garden opengarden-microweb
#ssg open-garden opengarden-xyz.com
#ssg service-search-order local remote
#!
#local-profile opengarden-microweb
# attribute 26 9 251 "R10.1.1.100;255.255.255.255"
# attribute 26 9 251 "D10.1.2.133"
# attribute 26 9 251 "Ocisco.com"
#!
#local-profile opengarden-xyz.com
# attribute 26 9 251 "R10.1.1.0;255.255.255.255"
# attribute 26 9 251 "D10.1.1.10"
# attribute 26 9 251 "Oxyz.com;zap.com"
#!
#local-profile opengarden-aggregation
# attribute 26 9 251 "D192.1.1.10"
# attribute 26 9 251 "Ocisco.com"
# attribute 26 9 251 "R11.1.1.99;255.255.255.255"
#!
```

The following lines illustrate how to configure the TCP redirect capability. The http-redirect group command specifies an arbitrary name for a captive portal group. You can define multiple captive portal groups, each one directing a set of subscribers to different SESM web applications.

This example defines one captive portal group (captive-portal-1) that is serviced by the SESM web application running on server 10.1.2.50, port 80. The following incoming requests are redirected to that SESM application:

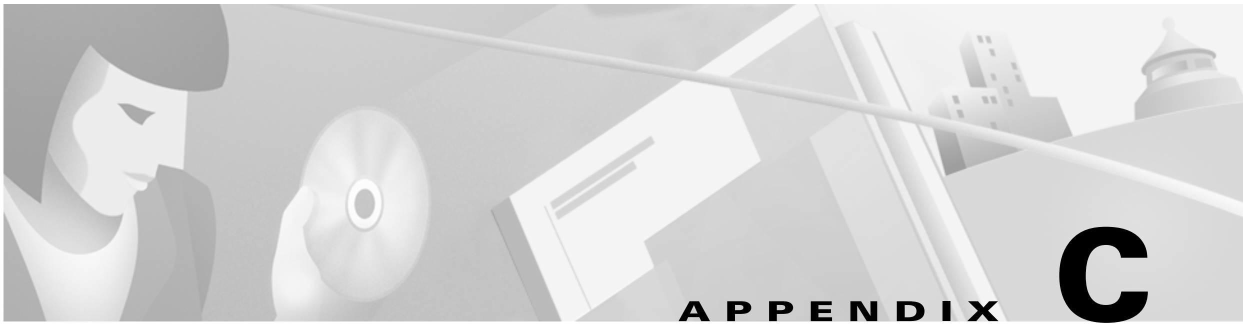
- Subscribers attempting to connect on port 81 get redirected.
- Subscribers attempting to connect from the network defined by IP address 192.168.10.0, mask 255.255.255.0.
- All unauthorized subscribers (that is, subscribers who have just opened their browsers and are not yet logged into SESM are redirected. Regardless of the URL they specify when they open their browsers, they are redirected to the SESM application first.

The lines below imply that the SESM application on server 10.1.2.50, port 80 is configured with the captive portal option turned on and that a captive portal application is running. The SESM application's related captive portal application examines the captured packet and determines an appropriate action. The captive portal application could authenticate the subscriber and display the list of authorized services, or it might display an SESM logon page. Another possibility is that the captive portal application could authenticate the subscriber and then redirect the packet to the original URL the subscriber specified. For example, the captive portal application might honor the home page specified in the subscriber's browser.

```
#ssg http-redirect group captive-portal-1 server 10.1.2.50 80
#ssg http-redirect port 81 group captive-portal-1
#ssg http-redirect network 192.168.10.0 255.255.255.0 group captive-portal-1
#ssg http-redirect unauthorized-user group captive-portal-1
#!
#!

#interface Ethernet0/0/1
# no ip address
#!
#interface Ethernet0/0/0
# description Management LAN
# ip address 192.168.2.48 255.255.255.0
#interface FastEthernet0/0/0
# ip address 192.168.1.48 255.255.255.0
# full-duplex
#!
```

The following lines illustrate how to configure communication between SSG and a RADIUS server.

```
#ip radius source-interface Ethernet0/0/0
#!
#radius-server configure-nas
#radius-server host 192.168.2.50 auth-port 1812 acct-port 1813
#radius-server retransmit 3
#radius-server timeout 60
#radius-server deadtime 2
#radius-server attribute 25 nas-port format d
#radius-server attribute nas-port format d
#radius-server key cisco
#radius-server vsa send accounting
#radius-server vsa send authentication
```



APPENDIX C

DTD for MBean Configuration Files

This appendix shows the full text of the DTD for the MBean configuration files used by ConfigAgent. This DTD name is xmlconfig.dtd.

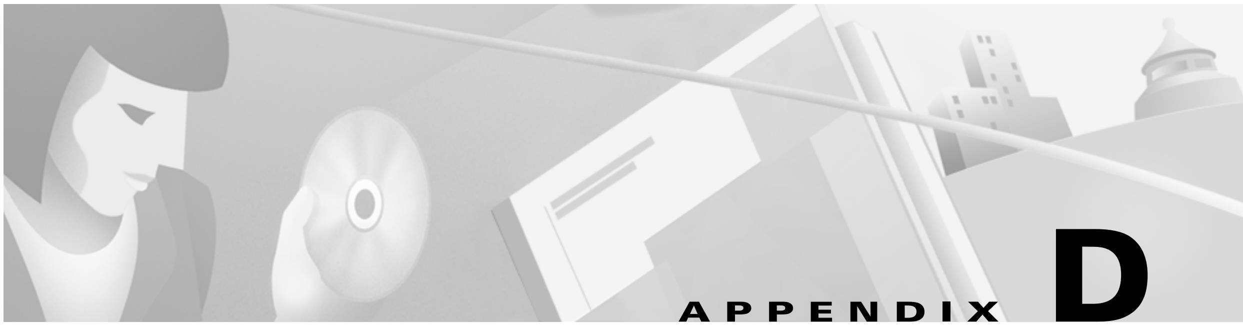
xmlconfig.dtd

```
<!-- Copyright (c) 2001 by Cisco Systems, Inc. All rights reserved.
This is the document type descriptor for the com.cisco.aggbu.jmx.XmlConfig
class, which was copied with permission from the
com.mortbay.Util.XmlConfiguration class.  It allows a MBean object to be
configured by with a sequence of Set, Put and Call elements.

The XML file contains a single <XmlConfig> element containing one or more
<Configure> elements describing the configuration for a single object or
class of object.

Each object or class to be configured is defined in a <Configure> element
section. A Configure element must have either a class or a jmxname attribute
defined. MBeans to be configured are matched by both class and jmxname,
so that two sets of configuration may be applied to an object.
If a Configure element has an init element and a class attribute, then an MBean instance
of that class is initialized and registered by the #newIntances(MBeanServer)
method. If a jmxname attribute is also provided, that is used for registration
with the MBean server.

Configure elements may contain Set, Put and Call elements which are used
in order by the #configure(MBeanServer,ObjectInstance) method.
Examples of these tags and their java equivalents are:

  <Set  name="Test">value</Set>              ~  obj.setTest("value");
  <Put  name="Test">value</Put>              ~  obj.put("Test","value");
  <Call name="test"><Arg>value</Arg></Call>  ~  obj.test("value");
  <Call name="test">
    <Arg>value</Arg>
    <Call name="other"/>
  </Call>                                    ~  obj.test("value").other();

Values may be literals or objects that are created with the
New element or returned from a Call element:

  <Set  name="Test1">
    <New class="com.acme.MyClass"/>
  </Set>

  <Set  name="Test2">
```

```
    <New class="com.acme.MyClass"/>
      <Arg type="int">42</Arg>
      <Set name="something"/>
    </New>
  </Set>

Note that Call and New elements may contain Set, Put and Call elements
after any Arg elements. These nested elements are applied to the
created or returned object.

Untyped balues are matched to arguments on a best effort approach. Primative
types may be specified as element attributes and the value is treated
as a String and converted to that type.
-->


<!ENTITY % CONFIG "Set|Put|Call">
<!ENTITY % TYPE "String|char|short|byte|int|long|boolean|float|double|URL">
<!ENTITY % VALUE "#PCDATA|Call|New|SystemProperty|Array">

<!ENTITY % IDATTR "id ID #IMPLIED" >
<!ENTITY % TYPEATTR "type (%TYPE;) #IMPLIED " >
<!ENTITY % ORDERATTR "order NMTOKEN #REQUIRED" >
<!ENTITY % CLASSATTR "class NMTOKEN #IMPLIED" >
<!ENTITY % NAMEATTR "name NMTOKEN #REQUIRED" >
<!ENTITY % JMXNAMEATTR "jmxname CDATA #IMPLIED" >

<!--
XmlConfig Element.
This is the root element of the configuration file:

    <XmlConfig> <Configure>...</Configure> ... </XmlConfig>

An XmlConfig element can contain Configure elements.
-->
<!ELEMENT XmlConfig ((Instantiate|Configure)*) >
<!ATTLIST XmlConfig %IDATTR; %CLASSATTR;>


<!--
Configure Element.
This is the root element that specifies the class of object that
can be configured:

    <Configure name="domain:n=v"> ... </Configure>

A Configure element can contain an optional Init element followed
by any number of Set, Put or Call elements.
-->
<!ELEMENT Configure (%CONFIG;)* >
<!ATTLIST Configure %IDATTR; %JMXNAMEATTR; %CLASSATTR;>


<!--
Instantiate Element.
This element specifies a set of arguments to an object constructor
and an order attribute specifying when the object is to be constructed
wrt all of the other objects scheduled to be created by the ConfigAgent:

    <Instantiate order="20"> ... </Init>
-->
<!ELEMENT Instantiate (Arg*,(%CONFIG;)*)>
<!ATTLIST Instantiate %IDATTR; %ORDERATTR; %JMXNAMEATTR; class NMTOKEN #REQUIRED>
```

```
<!--
Set Element.
This element maps to a call to a set method on the current object.
The name and optional type attributes are used to select the set
method.
A Set element can contain value text and/or the value elements Call,
New and SystemProperty. If no value type is specified, then white
space is trimmed out of the value. If it contains multiple value
elements they are added as strings before being converted to any
specified type.
-->
<!ELEMENT Set ( %VALUE; )* >
<!ATTLIST Set %IDATTR; %NAMEATTR; %TYPEATTR; >


<!--
Put Element.
This element maps to a call to a put method on the current object,
which must implement the Map interface. The name attribute is used
as the put key and the optional type attribute can force the type
of the value.

A Put element can contain value text and/or the value elements Call,
New and SystemProperty. If no value type is specified, then white
space is trimmed out of the value. If it contains multiple value
elements they are added as strings before being converted to any
specified type.
-->
<!ELEMENT Put ( %VALUE; )* >
<!ATTLIST Put %IDATTR; %NAMEATTR; %TYPEATTR;>


<!--
Call Element.
This element maps to an arbitrary call to amethod on the current object,
The name attribute and Arg elements are used to select the method.

A Call element can contain a sequence of Arg elements followed by
a sequence of Set, Put and/or Call elements which act on any object
returned by the original call:

 <Call name="test"><Arg>value1</Arg><Set name="Test">Value2</Set></Call>

This is equivalent to:

 Object o2 = o1.test("value1");
 o2.setTest("value2");

-->
<!ELEMENT Call (Arg*,(%CONFIG;)*)>
<!ATTLIST Call %IDATTR; %NAMEATTR;>


<!--
Arg Element.
This element defines a positional argument for the Call element.
The optional type attribute can force the type of the value.

An Arg element can contain value text and/or the value elements Call,
New and SystemProperty. If no value type is specified, then white
space is trimmed out of the value. If it contains multiple value
elements they are added as strings before being converted to any
specified type.
-->
```

```
<!ELEMENT Arg ( %VALUE; )* >
<!ATTLIST Arg %IDATTR; %TYPEATTR; >



<!--
New Element.
This element allows the creation of a new object as part of a
value of a Set, Put or Arg element. The class attribute determines
the type of the new object and the contained Arg elements
are used to select the constructor for the new object.

A New element can contain a sequence of Arg elements followed by
a sequence of Set, Put and/or Call elements which act on the new object:

 <New class="com.acme.MyClass">
   <Arg>value1</Arg><Set name="Test">Value2</Set>
 </New>

This is equivalent to:

 Object o = new com.acme.MyClass("value1");
 o.setTest("value2");

-->
<!ELEMENT New (Arg*,(%CONFIG;)*)>
<!ATTLIST New %IDATTR; %CLASSATTR; >



<!--
System Property Element.
This element allows JVM System properties to be retrieved as
part of the value of a Set, Put or Arg element.
The name attribute specifies the property name and the optional
default argument provides a default value.

 <SystemProperty name="Test" default="value"/>

This is equivalent to:

 System.getProperty("Test","value");

-->
<!ELEMENT SystemProperty EMPTY>
<!ATTLIST SystemProperty %IDATTR; %NAMEATTR; default CDATA #IMPLIED>


<!--
Array element
Can have a class attribute to specify the base type of each object
in the array.
-->
<!ELEMENT Array (Item)* >
<!ATTLIST Array %IDATTR; %CLASSATTR;>


<!--
Item element
Only occurs inside Arrays and is identical to Arg in every way except its name
-->
<!ELEMENT Item ( %VALUE; )* >
<!ATTLIST Item %IDATTR; %TYPEATTR; >
```



APPENDIX D

Configuring RADIUS

This appendix describes the configuration steps required to include a RADIUS server in an CiscoSubscriber Edge Services Manager (SESM) deployment. This appendix includes the following topics:

Configuring SSG to Communicate with the RADIUS Server

You must configure SSG to communicate with the RADIUS server. To do so, use the **radius-server host** Cisco IOS command on the SSG host. Different ports are used for handling authentication and accounting packets. For example:

```
radius-server host 10.3.3.2 auth-port 1812 acct-port 1813 key cisco
```

To use different RADIUS servers for authentication and accounting, use two commands as follows:

```
radius-server host 10.3.3.2 auth-port 1812 acct-port 0 key cisco
radius-server host 10.3.3.3 auth-port 0 acct-port 1813 key cisco
```

Configuring NAS Clients

The RADIUS protocol is based on a client server model. The RADIUS server is the server. Multiple dial-in Network Access Server (NAS) devices are the clients. Before communication can occur, every client must be configured on the server.

An SESM deployment requires that you configure the following NAS clients on the RADIUS server:

- The SSG host—This is the node route processors (NRP) on the Cisco 6400 UAC. The RADIUS server must recognize each SSG host as a client.
- The SESM web application—This is the NWSP application, or your customized SESM web application. SESM web applications query the RADIUS server directly for service information. The RADIUS server must recognize the SESM web application as a client.

Table D-1 summarizes the information that might be required to define a NAS client on the RADIUS server. See your RADIUS server vendor documentation for more specific requirements, syntax, and procedures.

Table D-1 NAS Client Configuration

Property	Description
Name or IP Address	Identifies the client. Use either IP address or hostname.
Shared Secret	Must match a shared secret value configured on the client. If the shared secrets do not match, the RADIUS server issues an access-reject message. A shared secret is a value that is configured on both the client and the server. It is never sent over the network. The shared secret is used for MD5 encryption of the profile password.
Type	For SSG—Cisco:NAS For SESM—RAD_RFC+ACCT_RFC

The following sample entries show a Merit RADIUS format defining SESM web applications and an SSG host as RADIUS clients. The examples use the value `cisco` as the shared secret on all of the clients.

```
#Entries for SESM-Server clients
10.3.3.2       cisco     type=RAD_RFC+ACCT_RFC
10.3.3.101     cisco     type=RAD_RFC+ACCT_RFC
10.3.3.102     cisco     type=RAD_RFC+ACCT_RFC

#Entries for 6400 NRP (SSG host)
192.168.1.6     cisco     type=Cisco:NAS
```

Adding Cisco SSG VSAs to the Attribute Dictionary

An attribute dictionary defines attributes to the RADIUS server. The attribute dictionary contains:

- Standard RADIUS attributes as defined by RFC 2138.
- Vendor-specific attributes (VSAs) that extend the standard attributes. VSAs add new capabilities, supported by specific vendors, to the RADIUS server. The value of a VSA can be one or more subattributes whose meanings depend on the vendor’s definition.

An SESM deployment requires that you add Cisco VSAs to your RADIUS attribute dictionary. See your RADIUS server vendor's documentation for instructions and syntax. The Cisco Access Registrar ships with all of the Cisco SESM VSAs preconfigured.

Table D-2 shows the Cisco VSAs required in an SESM deployment that uses a RADIUS server, which includes:

- SESM running in RADIUS mode. In this deployment, the RADIUS server supports authorization, authentication, and accounting features.
- SESM running in DESS mode and using SSG accounting features. In this deployment, the RADIUS server supports accounting features.
- SESM running in DESS mode and using the RADIUS DESS Proxy (RDP) in proxy mode. In this deployment, the RADIUS server supports authentication features.

Table D-2 Cisco SSG VSAs

RADIUS Attribute	Vendor ID	Subattribute	Name	Type
26	9	1	Cisco-Avpair	String
26	9	250	Account-Info	String
26	9	251	Service-Info	String
26	9	252	Command-Code	String
26	9	253	Control-Info	String

Configuring Service Profiles

Service profiles define the services that subscribers can access using the SESM web pages.

In an SESM deployment, you must configure a service profile for each service that will be accessible through the SESM web application.

Table D-3 briefly describes the attributes in a RADIUS service profile. Use the following references for more information.

- If you are using the Cisco Access Registrar, see the "Configuring Cisco Access Registrar for SESM Deployments" section on page D-11 for service profile examples and syntax. Otherwise, see your RADIUS server vendor documentation for the syntax of a service profile
- For sample SESM service profiles, see the demo.txt file located in the NWSP config directory (for example, nwsp/config/demo.txt). This file is installed whether or not you choose the demo option. It shows service and user profiles in Merit RADIUS format.
- The *Cisco 6400 Feature Guide*, Chapter 4, describes service profile attributes and provides examples of their use.

Table D-3 Attributes in Service Profiles

Attribute	Description
Service profile name	An identifying name for a service profile. Each profile name must be unique. Service profile names are used in the user profiles to indicate that a subscriber is subscribed to the service.
Password	Must match the service password that was configured on the SSG host and in SESM. On the SSG host (the Cisco 6400 NRP), configure a service password using the following Cisco IOS command: `ssg service password` *`password`* In SESM, configure the service password in the following line from the AAAMBean in the nwsp/config/nwsp.xml file: `<Set name=”servicePassword”>servicecisco</Set>`
Service-Type	Standard RADIUS attribute number 6. The value must be “outbound.”
Session-Timeout	Standard RADIUS attribute number 27. Specifies the maximum length of time, in seconds, that this service (the service object on SSG) can remain active in a session at any one time. When the time expires, SSG deletes the service object, which disconnects the subscriber from the service. If the host key feature is enabled on the SSG, the SSG signals the state change to the SESM web application. **Note** The NWSP application does not relay this state change to the subscriber. If Session-Timeout is not set, there is no limit on how long the subscriber can use the service. In a dial-up networking or bridged (non-PPP) network environment, a subscriber can disconnect from the NAS and release the IP address without logging out from the SSG. If this happens, the SSG continues to allow traffic to pass from that IP address, which can be a problem if the IP address is obtained by another user. You can use the Session-Timeout and the Idle-Timeout attributes to prevent this problem.
Idle-Timeout	Standard RADIUS attribute number 28. Specifies the maximum length of time, in seconds, that a service connection can remain idle before it is disconnected. See the explanation of the Session-Timeout attribute, above, for more information about setting this attribute.

Table D-3 Attributes in Service Profiles (continued)

Attribute	Description
Service-Info	A vendor-specific attribute (attribute number 26), vendor 9, subattribute 251. Valid values for Service-Info attributes are: • **I***description*—Service description. Optional. Describes the service. • **T***type*—Type of service. Optional. Valid values for *type* are: – P—Passthrough. This is the default. – T—Tunnel – X—Proxy. Indicates that the SSG performs proxy service. • **M***mode*—Service mode. Optional. Valid values for *mode* are: – S—Sequential mode. Prevents the subscriber from accessing any other services while connected to this service. – C—Concurrent mode. This is the default. Allows the subscriber to simultaneously log onto this service while connected to other services. • **R***ip_address;mask*—Service route (destination). Required. Specifies the network or the host where the service resides. Multiple instances of this attribute can exist within a single service profile, to specify multiple service destinations. An Internet service is typically specified as “`R0.0.0.0;0.0.0.0`”. • **D***ip_address_1[;ip_address_2]*—DNS Server Address. Optional. Specifies the IP addresses for the primary and secondary DNS servers to use for the domains that are defined using the O option. • **O***name1[name2]...[;nameX]*—Domain names. Optional. • **S***RadiusServerAddress;authPort;acctPort;secret*—Remote server information. Required when type of service (T) is Proxy (X); not applicable for other service types. Specifies the remote RADIUS server that will perform authentication, authorization, and accounting for this service.

Table D-3 Attributes in Service Profiles (continued)

Attribute	Description
Service-Info (*continued*)	• **G***key*—Service next hop gateway. Specifies the next hop key for this service. Each SSG uses its own next hop gateway table that associates this key with a valid IP address. See the "Configuring Optional Profile Features" section on page D-10 for information about creating a next hop gateway table. • **U***url* or **H***url*—These attributes specify the URL that is displayed in the HTTP address field when the service opens. If the SESM web application is designed to use HTML frames, then these options also specify whether the service is displayed in a new browser window or in a frame in the current (SESM) window, as follows: – **U***url*—URL for a service displayed in its own browser window. – **H***url*—URL for a service displayed in a frame in the SESM browser window. **Note** In a frameless application, both U and H cause a new browser window to open for the service. The NWSP application is a frameless application. • **X**—Indicates that the RADIUS authentication and accounting requests use the full username (for example, user@service). • **V***string*—Service-defined cookie. Optional. Specifies any information that you wish to include in RADIUS authentication and accounting requests. SSG does not parse or interpret *string*. You must configure the proxy RADIUS server to interpret this attribute. SSG supports only one service-defined cookie per service profile. Use this attribute to add fields to accounting records.

Table D-3 Attributes in Service Profiles (continued)

Attribute	Description
Cisco-AVpair	A vendor-specific attribute (attribute number 26), vendor 9, subattribute 1. Valid values for the Cisco-AVpair attribute in a service profile are: • **“ip:inacl**[#*number*]={*standardACL* | *extendedACL*}**”**—Upstream access control list (ACL). Specifies either a Cisco IOS standard ACL or an extended ACL to be applied to upstream traffic coming from the subscriber. • **“ip:outacl**[#*number*]={*standardACL* | *extendedACL*}**”**—Downstream ACL. Specifies either a Cisco IOS standard ACL or an extended ACL to be applied to downstream traffic going to the subscriber. – *number*—Identifies the access list. If a profile includes multiple inacl or outacl attributes, the attributes are downloaded and executed according to the order implied by *number*. – *standardACL*—A Cisco IOS standard ACL. – *extendedACL*—A Cisco IOS extended ACL. **Note** A profile can include multiple instances of inacl attributes and multiple instances of outacl attributes. Use one attribute for each ACL statement. Multiple attributes can be used for the same ACL. • **“vpdn:ip-addresses=***address1*[<*delimiter*>*address2*][<*delimiter*>*address3*]...**”**—Virtual private dial-up network (VPDN) IP address. Specifies the IP addresses of the home gateways (LNSs) to receive the L2TP connections. – *address*—IP address of the home gateway. – <*delimiter*>—A comma (,) or a space ( ) indicates that the SSG selects load sharing among IP addresses. A slash (/) indicates that the SSG considers IP addresses on the left side of the slash a higher priority than those on the right side of the slash. • **“vpdn:tunnel-id=***name***”**—VPDN tunnel ID. Specifies the name of the tunnel that must match the tunnel ID specified in the LNS VPDN group. See the *Cisco 6400 Feature Guide* for information about configuring LNS. • **“vpdn:tunnel-password=***secret***”**—L2TP tunnel password. Specifies the secret (password) used for L2TP tunnel authentication.

Example Service Profiles

The service configuration examples in this section use a Merit RADIUS format.

Example Service Profile for Passthrough Service

```
internet Password = "servicecisco", Service-Type = Outbound
    Service-Info = "IInternet",
    Service-Info = "R153.153.153.0;255.255.255.0",
    Service-Info = "MC",
    Service-Info = "TP"
```

Example Service Profile for Proxy Service

```
corporate Password = "servicecisco", Service-Type = Outbound
    Service-Info = "ICorporate Intranet (proxy)",
    Service-Info = "R154.154.154.0;255.255.255.0",
    Service-Info = "S10.3.3.101;1812;1813;cisco",
    Service-Info = "MC",
    Service-Info = "TX"
```

Example Service Profile Using Timeout Values

```
iptv Password = "servicecisco", Service-Type = Outbound
    Service-Info = "IIP/TV",
    Service-Info = "R160.160.160.0;255.255.255.0",
    Service-Info = "MC",
    Service-Info = "TP"
    Idle-Timeout = 60,
    Session-Timeout = 60
```

Configuring User Profiles

User profiles define SESM logon names and passwords, access control lists associated with each logon, and subscribed services for each logon.

In an SESM deployment, you must define a user profile for each user ID and password combination that will sign onto the SESM application from a web browser.

Table D-4 briefly describes the attributes in a RADIUS user profile. Use the following references for more information about:

- If you are using the Cisco Access Registrar, see the "Configuring Cisco Access Registrar for SESM Deployments" section on page D-11 for user profile examples and syntax. Otherwise, see your RADIUS server vendor documentation for the syntax of a user profile
- For sample SESM user profiles, see the demo.txt file located in the NWSP config directory (for example, nwsp/config/demo.txt). This file is installed whether or not you choose the demo option. It shows service and user profiles in Merit RADIUS format.
- The *Cisco 6400 Feature Guide*, Chapter 4, describes user profile attributes and provides examples of their use.

Table D-4 Attributes in User Profiles

Attribute	Description
Profile name	Identifies the profile. Each profile name must be unique.
Password	The user's password.
Session-Timeout	Standard RADIUS attribute number 27. Specifies the maximum length of time, in seconds, that this user session (the host object on SSG) can remain active at any one time. When the time expires, SSG deletes the host object, which ends the session. If the host key feature is enabled on the SSG, the SSG signals the state change to the SESM web application. **Note** The NWSP application does not relay this state change to the subscriber. If Session-Timeout is not set, there is no limit on how long the session lasts. In a dial-up networking or bridged (non-PPP) network environment, a subscriber can disconnect from the NAS and release the IP address without logging out from the SSG. If this happens, the SSG continues to allow traffic to pass from that IP address, which can be a problem if the IP address is obtained by another user. You can use the Session-Timeout and the Idle-Timeout attributes to prevent this problem.
Idle-Timeout	Standard RADIUS attribute number 28. Specifies the maximum length of time, in seconds, that a user session can remain idle before it is disconnected. See the explanation of the Session-Timeout attribute, above, for more information about setting this attribute.
Account-Info	A vendor-specific attribute (attribute number 26), vendor 9, subattribute 250. Valid values for Account-Info attributes are: • **“N***serviceName***”**—Service name. Subscribes the user to the specified service and includes the service in the service list for that user obtained by the SESM web application. The *serviceProfileName* must be defined in a service profile. There can be multiple instances of this attribute within a user profile. • **“G***serviceGroupProfileName***”**—Service group. Creates a folder for the service group on the subscriber's SESM web page. The *serviceGroupProfileName* must be defined in a service group profile. There can be multiple instances of this attribute within a user profile. • **“A***autoConnectServiceName***”**—Automatic connection. Subscribes the user to the specified service and indicates that the user should be automatically connected to this service after successful logon. **Note** The service list displayed by SESM does not include A entries. It only shows N entries. For more information, see the “Example User Profile for Auto Services” section on page D-10. • **U***url* or **H***url*—These attributes specify the URL for the user's preferred Internet home page. If the SESM web server application is designed to use HTML frames, then these options also specify whether the home page is displayed in a new browser window or in a frame in the current (SESM) window, as follows: – **U***url*—URL for the home page displayed in its own browser window. – **H***url*—URL for the home page displayed in a frame in the SESM browser window. **Note** In a frameless application, both U and H cause a new browser window to open for the home page. The NWSP application is a frameless application.

Table D-4 Attributes in User Profiles (continued)

Attribute	Description
Cisco-AVpair	A vendor-specific attribute (attribute number 26), vendor 9, subattribute 1. Valid values for the Cisco-AVpair attribute in a user profile are: • **“ip:inacl**[#*number*]={*standardACL* | *extendedACL*}**”**—Upstream access control list (ACL). Specifies either a Cisco IOS standard ACL or an extended ACL to be applied to upstream traffic coming from the subscriber. • **“ip:outacl**[#*number*]={*standardACL* | *extendedACL*}**”**—Downstream ACL. Specifies either a Cisco IOS standard ACL or an extended ACL to be applied to downstream traffic going to the subscriber. – *number*—Identifies the access list. If a profile includes multiple inacl or outacl attributes, the attributes are downloaded and executed according to the order implied by *number*. – *standardACL*—A Cisco IOS standard ACL. – *extendedACL*—A Cisco IOS extended ACL. **Note** A profile can include multiple instances of inacl attributes and multiple instances of outacl attributes. Use one attribute for each ACL statement. Multiple attributes can be used for the same ACL.

Example User Profiles

The user profile example in this section is in a Merit RADIUS format.

Example User Profile for Auto Services

```
user1 Password = "cisco"
    Service-Type = Framed-User,
    Account-Info = "Ainternet”,       (hidden on the subscriber’s web page)
    Account-Info = ”Ninternet"        (makes it visible)
```

Note The first Account-Info line specifies automatic connection to the service. If you do not include the second line, the autoconnection service does not appear on the SESM web page. To display the service on the SESM web page, you must include both entries as shown in the example.

Configuring Optional Profile Features

SSG supports the following additional optional features. See the *Cisco 6400 Feature Guide* for information about these features.

- Service group profiles—Use these profiles to create service groups, which are groups of services that can be subscribed to as a unit.
- Pseudo-service profiles—Use these profiles to define variable length tables or lists of information in the form of services.
  - Transparent Passthrough Filter—Is no longer supported.
  - Next Hop Gateway—Associates next hop gateway keys with IP addresses. Because multiple SSGs might access services from different networks, service profiles can specify next hop keys. (See the service-info **G** attribute in Table D-3 on page D-4.) If this is the case, you must configure a next hop gateway pseudo-service profile to resolve the keys to valid IP addresses.

An example next hop gateway pseudo-service profile follows:

```
ssg-next-hop Password = "xssg-key"
Control-Info = "Gl2tp-net7;192.168.1.101",
Control-Info = "Gl2tp-net40;192.168.1.102",
Control-Info = "Gweb-key;192.168.1.101",
Control-Info = "Gproxy-radius-key;192.168.1.101",
Control-Info = "Gxint-24;192.168.1.101"
```

Configuring the RADIUS Accounting Feature

If you configure a RADIUS accounting port, SSG generates accounting records and forwards them to the RADIUS server. To configure a RADIUS server for accounting only, you must perform the following configuration steps.

- Configure the NAS clients as described in the "Configuring NAS Clients" section on page D-2.
- Add the Cisco VSAs to the RADIUS server attribute dictionary, as described in the "Adding Cisco SSG VSAs to the Attribute Dictionary" section on page D-2.
- Configure an accounting port, as described in the "Configuring SSG to Communicate with the RADIUS Server" section on page D-1.

Note You do not need to provide service and user profiles if you are using the RADIUS server solely for accounting purposes.

The subscriber actions that cause SSG to generate a RADIUS accounting record are:

- Subscriber logs in
- Subscriber logs off
- Subscriber accesses a service
- Subscriber terminates a service

Use the following references for more information:

- Chapter 4 in the *Cisco 6400 Feature Guide*—Descrubes the attributes contained in the accounting records
- RADIUS server vendor documentation—Describes RADIUS accounting capabilities

Configuring Cisco Access Registrar for SESM Deployments

This section describes how to configure the Cisco Access Registrar (Cisco AR) for an SESM deployment. The section includes profile examples in Cisco AR format.

Configuring the RADIUS Ports

By default, Cisco Access Registrar listens on ports 1645 and 1646 for any type of RADIUS request. You can configure Cisco Access Registrar to listen on ports 1812 and 1813 instead by entering the following commands:

```
add /Radius/Advanced/Ports/1812
add /Radius/Advanced/Ports/1813
```

These commands cause Cisco Access Registrar to listen on the explicitly defined ports, 1812 and 1813, for all types of RADIUS requests. It no longer listens on the default ports.

Cisco SSG VSAs in Cisco Access Registrar's Dictionary

Cisco Access Registrar is installed with the following Cisco VSAs already defined in its attribute dictionary:

- Cisco-AVPair
- Cisco-SSG-Account-Info
- Cisco-SSG-Service-Info
- Cisco-SSG-Command-Code
- Cisco-SSG-Control-Info

Configuring NAS Clients in Cisco Access Registrar

Use the following commands to configure the NAS clients required by an SESM deployment:

```
add /Radius/Clients/SESM1 "" 10.3.3.2 cisco
add /Radius/Clients/SESM2 "" 10.3.3.101 cisco
add /Radius/Clients/SESM1 "" 10.3.3.102 cisco
```

Configuring Attribute Profiles in Cisco Access Registrar

This section shows commands for creating sample profiles in Cisco Access Registrar format.

Internet Service Profile

```
add /Radius/Profiles/internet-profile
set /Radius/Profiles/internet-profile/Attributes/Cisco-SSG-Service-Info IInternet
    R153.153.153.0;255.255.255.0 MC TP
```

Corporate Service Profile

```
add /Radius/Profiles/corporate-profile
set /Radius/Profiles/corporate-profile/Attributes/Cisco-SSG-Service-Info "ICorporate
    Intranet(proxy)" R154.154.154.0;255.255.255.0 S10.3.3.101;1812;1813;cisco MC TX
```

IPTV Profile

```
add /Radius/Profiles/iptv-profile
set /Radius/Profiles/iptv-profile/Attributes/Cisco-SSG-Service-Info IIP/TV
    R160.160.160.0;255.255.255.0 MC TP
set /Radius/Profiles/iptv-profile/Attributes/Idle-Timeout 60
set /Radius/Profiles/iptv-profile/Attributes/Session-Timeout 60
```

Standard user profile

```
add /Radius/Profiles/std-user-profile
set /Radius/Profiles/std-user-profile/Attributes/Service-Type Framed
set /Radius/Profiles/std-user-profile/Attributes/Cisco-SSG-Account-Info Ainternet
    Ninternet
```

Pseudo-service profile:

```
add /Radius/Profiles/pseudo-service-profile
set /Radius/Profiles/pseudo-service-profile/Attributes/Cisco-SSG-Control-Info
    Gl2tp-net7;192.168.1.101 Gl2tp-net40;192.168.1.102 Gweb-key;192.168.1.101
    Gproxy-radius-key;192.168.1.101 Gxint-24;192.168.1.101
```

Configuring Cisco Access Registrar Userlists and AA Services

This section describes how to configure userlists and authentication and authorization services on Cisco Access Registrar.

Configuring Userlist for SESM Services

The following commands configure userlists containing SESM services and corresponding attribute profiles.

```
add /Radius/Userlists/SESMservices
add /Radius/Userlists/SESMservices/internet "" servicecisco TRUE "" internet-profile
add /Radius/Userlists/SESMservices/corporate "" servicecisco TRUE "" corporate-profile
add /Radius/Userlists/SESMservices/iptv "" servicecisco TRUE "" iptv-profile
```

Configuring Userlist for SESM Users

The following commands configure userlists containing SESM users and corresponding attribute profiles.

```
add /Radius/Userlists/SESMusers
add /Radius/Userlists/SESMusers/user1 "" cisco TRUE "" std-user-profile
add /Radius/Userlists/SESMusers/ssg-next-hop "" xssg-key TRUE "" pseudo-service-profile
```

Configuring AA Services

The following commands configure Cisco Access Register AA services. The first command configures services for the SESM services userlist. The second command configures services for SESM users userlist.

```
add /Radius/Services/Outbound "" local "" "" RejectAll "" SESMservices
add /Radius/Services/SESMdefault "" local "" "" RejectAll "" SESMusers
```

Checking the Service-Type Attribute

The following commands configure Cisco Access Registrar to check the Service-Type attribute in the request. If Service-Type is set to Outbound, then the Outbound AA service is used; otherwise, the SESMdefault AA service is used.

```
set /Radius/DefaultAuthenticationService ${q|Service-Type}{SESMdefault}
set /Radius/DefaultAuthorizationService ${q|Service-Type}{SESMdefault}
```

Configuring Accounting on Cisco Access Registrar

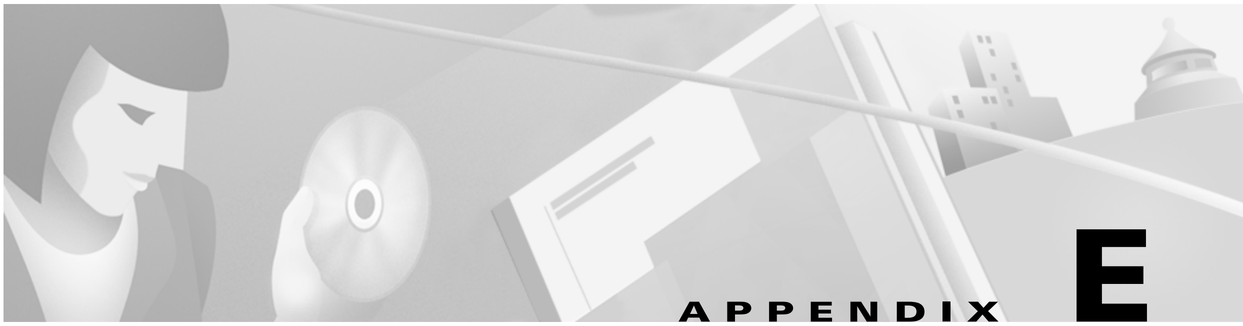
To configure accounting services, use the following commands:

```
add /Radius/Services/SESMaccounting "" file
set /Radius/DefaultAccountingService SESMaccounting
```

Saving the Configuration and Reloading the Server

To save the configuration and reload the Cisco Access Registrar server, use the following commands:

```
save
reload
```



RDP Packet Handlers

RDP is a flexible and extensible application. This appendix describes the programming methodology in RDP that processes requests received from SSG. It includes the following topics:

- Packet Handlers, page E-1
- RDPPacket Class Description, page E-2
- Processing Requests in Proxy Mode, page E-5

Packet Handlers

This section describes the RDP packet handler class. It includes the following topics:

- Overview of Packet Handlers, page E-1
- Configuring the Packet Handlers, page E-2
- Adding Additional Packet Handlers, page E-2

Overview of Packet Handlers

The RDP application is very flexible in the way it handles requests that it receives from SSG. This flexibility is implemented with a number of different packet handlers, each handling a request in a different way. Developers at your site can extend the RDP application with additional packet handlers to provide even more flexibility.

RDP cycles a request from SSG through several levels of packet handlers, each one working to narrow down the type of packet, until a response is generated. The request is initially untyped and is processed by the packet handler for untyped packets. As the request gets processed by various packet handlers, it gets typed several times, each time with a more specific type. RDP creates a new packet object to process each newly assigned packet type.

Configuring the Packet Handlers

The RDPPacketFactoryMBean is the configurable class that specifies the packet handlers to use for each packet type. The rdp.xml file includes the following entries for each packet handler:

```
<Call name="addType">
      <Arg>packetType</Arg>
      <Arg>class</Arg>
    </Call>
```

Each <Call name="addType"> element takes two arguments: a packet type and a class that will handle that packet type. The *packetType* is a string. The *class* is a string specifying an RDPPacket derived class. Class parameters follow the class and are separated from it by a semicolon.

The RDPPacketFactoryMBean also accepts entries that set attributes. The attribute entries are used as parameters to the ProfileRequestPacket packet handler to narrow down the packet type.

```
<Call name="setAttribute">
      <Arg>PASSWORD:password</Arg>
      <Arg>packetType</Arg>
    </Call>
```

Each <Call name="setAttribute"> element takes two arguments: a *password* and a *packetType*.

There must be a corresponding <Call name="addType"> element for *packetType*, to specify the packet handler class for that packet type.

Adding Additional Packet Handlers

This packet handling mechanism is extensible. Web developers can write customized or additional packet handlers and map them to specific packet types by making changes or additions in the rdp.xml file.

RDPPacket Class Description

When RDP receives a request, it creates an RDPPacket. The packet handlers in the RDPPacket class have two public methods:

- getType method
- handle method

An RDPPacket derived class either overrides the getType method, in which case it narrows down the type of the packet, or it overrides the handle method, in which case it generates a response. An object calls the handle method first. If the handle method can process the request, it does so, generating the response. Otherwise, the default RDPPacket handle method calls the getType method.

The getType method determines some information about the type of packet. The default handle method uses the returned type to create a new RDPPacket derived packet. The handle method is then called on the new packet, as described in the previous paragraph.

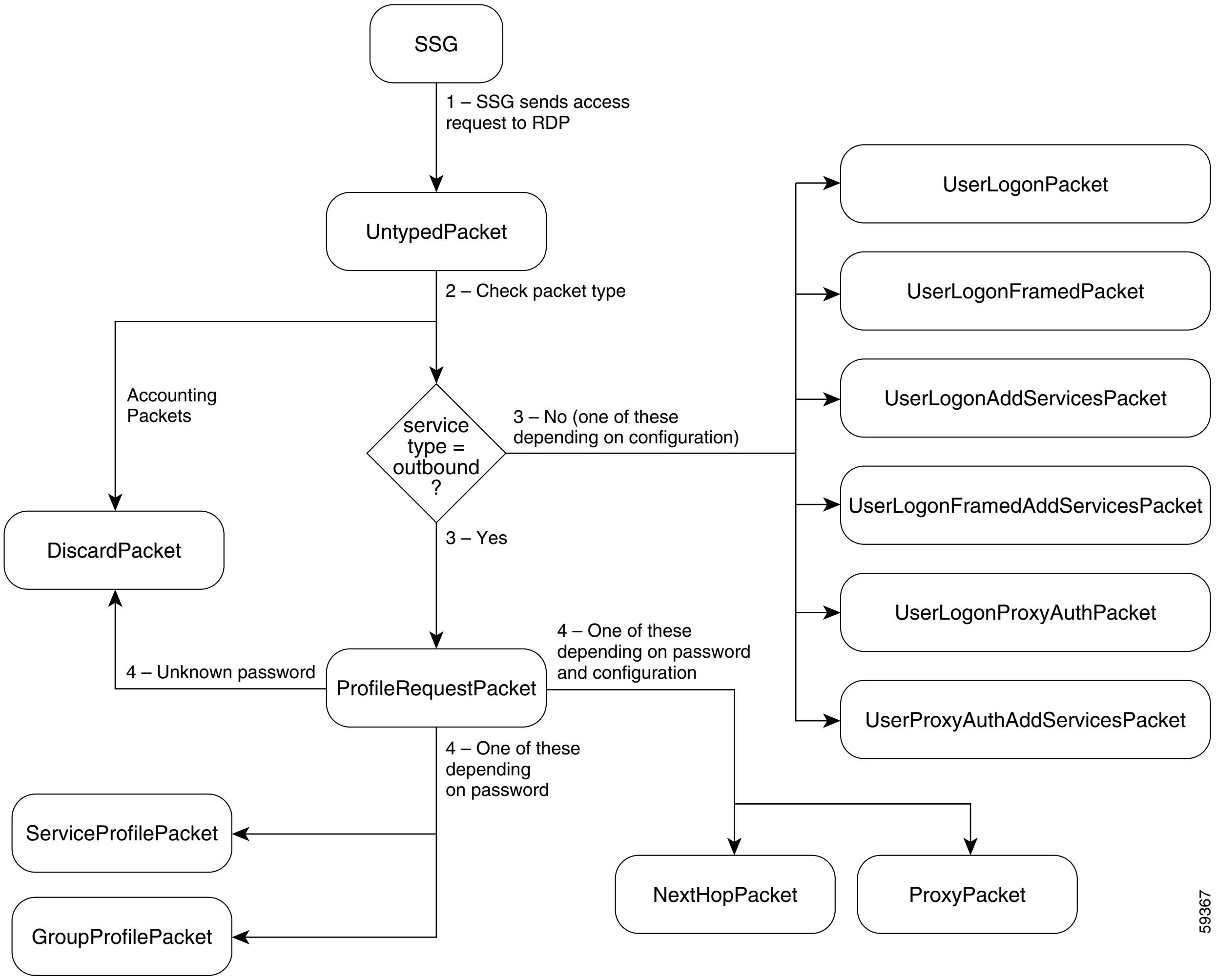
Table E-1 describes the RDPPacket classes included with the installed RDP application.

Table E-1 RDPacket Classes and Methods

Class	Methods
RDPPacket	getType—If the request is an Access Request, this method prompts you with Untyped. Otherwise, the method prompts you with Unknown.
DiscardPacket	handle—Returns null. (That is, it silently discards the request.)
RejectPacket	handle—Returns an Access Reject message.
UntypedPacket	getType—If the request contains the AV Service-Type with the value `Outbound`, then the method ProfileRequest appears. Otherwise, this method prompts you with UserLogon.
ProfileRequestPacket	getType—If the request contains a password that matches a password defined by the `PASSWORD:` attribute, this method displays the attribute's value. Otherwise, this method prompts you with Unknown.
ProxyPacket	handle—Proxies the request to an AAA server. Requires a parameter to define the name of the AAA MBean.
ServiceProfilePacket	handle—Uses the DESS API to create a service profile response.
GroupProfilePacket	handle—Uses the DESS API to create a group profile response.
NextHopPacket	handle—Uses the DESS API to create a next hop gateway response.
UserLogonAddServices Packet	handle—Uses the DESS API to authenticate and authorize a subscriber. All services and groups the subscriber is subscribed to appear.
UserLogonPacket	handle—Uses the DESS API to authenticate a subscriber. If the subscriber is using PPP, the subscriber's auto-logon services appear.
UserProxyAuthAdd ServicePacket	handle —Proxies the request to a AAA server, but uses DESS to add authorization information. Requires a parameter to define the name of an AAA MBean.
UserProxyAuthPacket	handle—Proxies the request to an AAA server, but uses DESS to add authorization information for auto-logon services if the user is a PPP user. Requires a parameter to define the name of an AAA MBean.

Figure E-1 shows how RDP processes a request from SSG. A detailed explanation follows the figure.

Figure E-1 RDP Request Processing


SSG
1 – SSG sends access request to RDP
UntypedPacket
2 – Check packet type
Accounting Packets
service type = outbound ?
3 – No (one of these depending on configuration)
UserLogonPacket
UserLogonFramedPacket
UserLogonAddServicesPacket
UserLogonFramedAddServicesPacket
UserLogonProxyAuthPacket
UserProxyAuthAddServicesPacket
DiscardPacket
3 – Yes
4 – Unknown password
ProfileRequestPacket
4 – One of these depending on password and configuration
4 – One of these depending on password
ServiceProfilePacket
GroupProfilePacket
NextHopPacket
ProxyPacket
59367

A request from SSG is processed in the following way:

1. The initial packet is handled by the base class. The getType method returns Untyped.
2. An Untyped packet is handled by the UntypedPacket class.
3. The getType method returns one of the following types:
   - If the packet contains the AV pair service-type = Outbound, getType returns ProfileRequest packet.
   - Otherwise, getType returns one of the UserLogon request packets, depending on values in the MBean configuration file.

4. A ProfileRequest packet is handled by the ProfileRequestPacket class. This class narrows the type again using the PASSWORD: attributes set in the rdp.xml file. If the password in the request (prepended with the string PASSWORD:) matches any of the password attributes set in the rdp.xml file, the getType method returns the packet type associated with the password in the corresponding <Call name="setAttribute"> element. Password attributes identify the following types of requests:
   - ServiceRequest—The ServiceRequest packet handler uses the DESS API to retrieve a list of services that this subscriber is authorized to access.
   - GroupRequest —The GroupRequest packet handler uses the DESS API to retrieve a list of services that this subscriber is authorized to access through group membership.
   - ProxyNextHop—The ProxyNextHop packet handler passes the request to the RADIUS server identified in the AAA MBean in the rdp.xml file.
   - If the password does not match any of the above, getType returns Unknown. An Unknown packet is handled by the RejectPacket packet handler.

   See the "RDPPacketFactory" section in Table 4-6 on page 4-33 for information about how to set these password values.

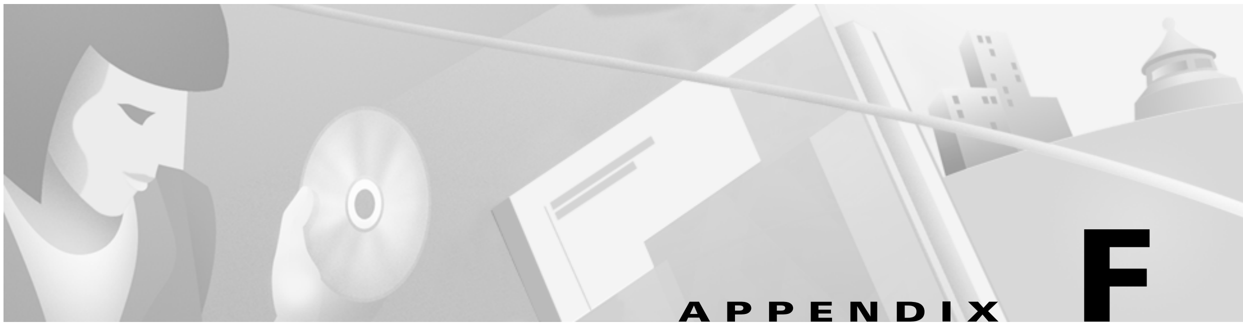
Processing Requests in Proxy Mode

When RDP is running in Proxy mode, profile requests are forwarded to a RADIUS server. This section describes the configuration entries in rdp.xml that make this happen. The section discusses the following entries from the installed rdp.xml file.

```
<Call name="setAttribute">
      <Arg>PASSWORD:nexthopcisco</Arg>
      <Arg>ProxyNextHop</Arg>
    </Call>

<Call name="addType">
      <Arg>ProxyNextHop</Arg>
      <Arg>com.cisco.aggbu.rdp.ProxyPacket;NextHop</Arg>
    </Call>

<Configure name="com.cisco.aggbu:name=AAA,connection=NextHop">
```

If a ProfileRequestPacket has the password nexthopcisco (this is an example; your password value might be different), it is typed ProxyNextHop. The `<Call name="addType">` element for ProxyNextHop maps the packet to the ProxyPacket class.

The ProxyPacket class accepts a string in its constructor which identifies the connection object that will handle the request. The string after the class name and semicolon in the `<Call name="addType">` element is passed to the ProxyPacket class constructor. This connection object name matches the connection object configured by the AAA MBean.



APPENDIX F

Sample MBean Configuration Files

This appendix contains sample MBean configuration files. It includes the following sections:

- Sample Container MBean Configuration File, page F-1
- Sample Application MBean Configuration File, page F-3
- Sample RDP MBean Configuration File, page F-13
- Sample CDAT MBean Configuration File, page F-16
- Sample DESS MBean Configuration File, page F-17

Sample Container MBean Configuration File

An example jetty/config/nwsp.jetty.xml file follows.

```
<?xml version="1.0"  encoding="ISO-8859-1"?>
<!DOCTYPE XmlConfig PUBLIC "-//Cisco Systems//DTD XmlConfig 1.1//EN"
"http://www.cisco.com/aggbu/xmlconfig_1_1.dtd">
<!-- Copyright (c) 2001 by Cisco Systems, Inc. All rights reserved. -->
<!-- This is the container specific configuration for the NWSP web application.
     Container independant configuration can be found at:
        $INSTALLROOT/nwsp/config/nwsp.xml
-->

<XmlConfig>

  <!-- ================================================================ -->
  <Instantiate order="10" class="com.mortbay.Jetty.JMX.LogMBean"/>
  <Instantiate order="11" class="com.mortbay.Jetty.JMX.DebugMBean"/>
  <Instantiate order="12"
             class="com.mortbay.Jetty.JMX.HttpServerMBean"
             jmxname="com.mortbay.Jetty:name=Jetty,Server=0"/>

  <!-- ================================================================ -->
  <Configure jmxname="com.mortbay.Jetty:name=Log,WriterLogSink=0">
    <Set name="append" type="boolean">true</Set>
    <Set name="filename"><SystemProperty name="application.log"
default="./logs"/>/yyyy_mm_dd.jetty.log</Set>
    <Set name="logTimezone"></Set>
    <Set name="logDateFormat">yyyyMMdd:HHmmss.SSS' '</Set>
    <Set name="logLabels"     type="boolean">false</Set>
    <Set name="logOneLine"    type="boolean">false</Set>
    <Set name="logStackSize"  type="boolean">false</Set>
    <Set name="logStackTrace" type="boolean">false</Set>
    <Set name="logTags"       type="boolean">true</Set>
    <Set name="logTimeStamps" type="boolean">true</Set>
    <Set name="retainDays"    type="int">31</Set>
  </Configure>

  <Configure class="com.mortbay.Jetty.JMX.DebugMBean" >
    <Set name="debug" type="boolean">false</Set>
    <Set name="debugPatterns"></Set>
    <Set name="debugTriggers"></Set>
    <Set name="verbose" type="int">0</Set>
    <Set name="suppressStack" type="boolean">false</Set>
    <Set name="suppressWarnings" type="boolean">false</Set>
  </Configure>

  <!-- ================================================================ -->
  <Configure jmxname="com.mortbay.Jetty:name=Jetty,Server=0">
    <Call name="addListener">
      <Arg>
        <New class="com.mortbay.HTTP.SocketListener">
          <Set name="port"><SystemProperty name="application.portno"
default="8080"/></Set>
          <Set name="minThreads">5</Set>
          <Set name="maxThreads">255</Set>
          <Set name="maxIdleTimeMs">60000</Set>
          <Set name="maxReadTimeMs">60000</Set>
        </New>
      </Arg>
    </Call>

    <Call name="addListener">
      <Arg>
        <New class="com.mortbay.HTTP.SunJsseListener">
          <Set name="port"><SystemProperty name="application.ssl.portno"
default="8130"/></Set>
          <Set name="MinThreads">5</Set>
          <Set name="MaxThreads">255</Set>
          <Set name="MaxIdleTimeMs">50000</Set>
          <Set name="Keystore"><SystemProperty name="jetty.home"
default="."/>/config/nwspkeystore</Set>
      <Set name="Password">OBF:1vny1zlo1x8e1vnw1vn61x8g1zlu1vn4</Set>
      <Set name="KeyPassword">OBF:1u2u1wml1z7s1z7a1wnl1u2g</Set>
        </New>
      </Arg>
    </Call>

    <Set name="logSink">
      <New class="com.mortbay.Util.WriterLogSink">
        <Arg><SystemProperty name="application.log"
default="./logs"/>/yyyy_mm_dd.request.log</Arg>
        <Set name="retainDays">90</Set>
        <Set name="append">true</Set>
      </New>
    </Set>

    <!-- NWSP web application -->
    <Call name="addWebApplication">
      <Arg>localhost</Arg>
      <Arg>/</Arg>
      <Arg><SystemProperty name="application.home" default="."/>/docroot</Arg>
      <Arg><SystemProperty name="jetty.home" default="."/>/config/webdefault.xml</Arg>
      <Arg type="boolean">FALSE</Arg>
      <Call name="addHandler">
           <Arg type="int">0</Arg>
           <Arg><New class="com.cisco.aggbu.jetty.PortBundleHandler"/></Arg>
      </Call>
    </Call>

    <!-- Captive portal web application -->
    <Call name="addWebApplication">
      <Arg></Arg>
      <Arg>/</Arg>
      <Arg><SystemProperty name="install.root" default="."/>/captiveportal/docroot</Arg>
      <Arg><SystemProperty name="jetty.home" default="."/>/config/webdefault.xml</Arg>
      <Arg type="boolean">FALSE</Arg>
    </Call>

    <Call name="start"/>

  </Configure>

</XmlConfig>
```

Sample Application MBean Configuration File

This section contains two sample files:

- RADIUS Mode Deployment, page F-3
- DESS Mode Deployment, page F-8

RADIUS Mode Deployment

The following nwsp/config/nwsp.xml file shows a RADIUS mode deployment with the captive portal feature enabled.

```
<?xml version="1.0"  encoding="ISO-8859-1"?>
<!DOCTYPE XmlConfig PUBLIC "-//Cisco Systems//DTD XmlConfig 1.1//EN"
"http://www.cisco.com/aggbu/xmlconfig_1_1.dtd">
<!-- Copyright (c) 2001 by Cisco Systems, Inc. All rights reserved. -->
<!-- This is the container independent configuration for the NWSP web application.
     Container specific configuration can be found at:
         $INSTALLROOT/$CONTAINER/config/nwsp.xml
-->

<XmlConfig>
  <!-- ================================================================ -->
  <Instantiate order="1"
             class="com.cisco.aggbu.jmx.LoggerMBean"
             jmxname="com.cisco.aggbu:name=Logger"/>

  <Instantiate order="99"
               class="com.sun.jdmk.comm.HtmlAdaptorServer"
               jmxname="com.cisco.aggbu:name=ManagementConsole">
      <Arg type="int">
        <SystemProperty name="management.portno"/>
      </Arg>
      <Arg>
<Array class="com.sun.jdmk.comm.AuthInfo">
          <Item>
            <New class="com.sun.jdmk.comm.AuthInfo">
              <Arg>MgmtUser</Arg>
              <Arg>MgmtPassword</Arg>
            </New>
          </Item>
</Array>
      </Arg>
  </Instantiate>

 <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=Logger">
    <Set name="debug" type="boolean"><SystemProperty name="nwsp.debug" 
default="false"/></Set>
    <Set name="debugPatterns"></Set>
    <Set name="debugThreads"></Set>
    <Set name="debugVerbosity">LOW</Set>
    <Set name="logDateFormat">yyyyMMdd:HHmmss.SSS</Set>
    <Set name="logFile"><SystemProperty name="application.log" 
default="./logs"/>/yyyy_mm_dd.application.log</Set>
    <Set name="logFrame"  type="boolean">false</Set>
    <Set name="logStack"  type="boolean">false</Set>
    <Set name="logThread" type="boolean">true</Set>
    <Set name="logToErr"  type="boolean"><SystemProperty name="nwsp.logToErr" 
default="false"/></Set>
    <Set name="trace"     type="boolean">true</Set>
    <Set name="warning"   type="boolean">true</Set>
  </Configure>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=ManagementConsole">
    <Call name="start"/>
  </Configure>

  <!-- ================================================================ -->
  <Configure class="com.cisco.aggbu.ssd.core.model.SSDMBean"
             jmxname="com.cisco.aggbu:name=SSD">
    <Call name="defineMode">
      <Arg>Demo</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.demo.DemoAuthenticationService</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.demo.DemoAuthorizationService</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.demo.DemoConnectionService</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.demo.DemoServiceProfileService</Arg>
    </Call>
    <Call name="defineMode">
      <Arg>RADIUS</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSAuthentication</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSAuthorization</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSConnection</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSServiceProfile</Arg>
    </Call>
    <Call name="defineMode">
      <Arg>DESS</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSAuthentication</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.dess.DESSAuthorizationService</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSConnection</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.dess.DESSServiceProfileService</Arg>
    </Call>
    <Set name="mode"><SystemProperty name="ssd.mode" default="RADIUS"/></Set>
    <Set name="singleSignOn" type="boolean">false</Set>
    <Set name="autoConnect" type="boolean">false</Set>
    <Set name="profileCachePeriod" type="int">600</Set>
  </Configure>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=SSDDemoMode">
    <!--
      - This is the demo data file. It is in the format of a Merit
      - dictionary with special extensions for this software.
      -->
    <Set name="demoDataFile"><SystemProperty
name="application.home"/>/config/demo.txt</Set>
     <!--
       - This is is an example of using brands in Demo mode.
       - See the definitions for the brands below. The same example
       - can be used in SSG configuration to tie subnets to brands.
       -->
     <Call name="setSubnetAttribute">
       <Arg>127.0.0.0</Arg>
       <Arg>255.0.0.0</Arg>
       <Arg>SESSION_BRAND</Arg>
       <Arg>gold</Arg>
    </Call>
  </Configure>

  <!-- =============================================================== -->
  <!-- Settings for the DESS SPI. -->
  <Configure jmxname="com.cisco.aggbu:name=DESSMode">
    <!-- The time in minutes between checking the authorization tokens. -->
    <Set name="tokenCheckInterval" type="int">5</Set>
    <!-- The age of a token (time since last used) for it to be removed from cache. -->
    <Set name="tokenMaxAge" type="int">10</Set>
  </Configure>

  <!-- =============================================================== -->
  <Configure jmxname="com.cisco.aggbu:name=SSG">
<!--
  - Maxmimum number of simultaneous requests allowed to each SSG. Extra
  - requests will be placed on a queue and issued as responses are received
  - or timeout.
  -->
    <Set name="throttle" type="int">20</Set>

    <!--
      - Here we define attributes for RADIUS communication with the SSG If
      - we are running with Port Bundle Host key then we need only define
      - the global attributes for all of the SSGs.
      -->
    <Call name="setGlobalAttribute"><Arg>PORT</Arg><Arg>1812</Arg></Call>
    <Call name="setGlobalAttribute"><Arg>TIMEOUTSECS</Arg><Arg>10</Arg></Call>
    <Call name="setGlobalAttribute"><Arg>RETRIES</Arg><Arg>3</Arg></Call>
    <Call name="setGlobalAttribute"><Arg>SECRET</Arg><Arg>cisco</Arg></Call>
    <Call name="setGlobalAttribute"><Arg>MASK</Arg><Arg>255.255.255.255</Arg></Call>
    <!--
      - A non zero value here, the default should be 4, will turn Port
      - Bundle Host Key on.
      -->
    <Call name="setGlobalAttribute"><Arg>BUNDLE_LENGTH</Arg><Arg>0</Arg></Call>
    <!-- The following line configures a single non-hostkey SSG              -->
    <!-- Additional SSGs can be configured by adding further 'Call' elements -->
    <!-- Remove the following call if the bundle size is ever set to > 0     -->
    <!-- Arg list: <client subnet>, <subnet mask>, IP, <SSG IP address>      -->
    <Call
name="setSubnetAttribute"><Arg>10.25.0.0</Arg><Arg>255.255.0.0</Arg><Arg>IP</Arg><Arg>10.5
.5.1</Arg></Call>
    <Call
name="setGlobalAttribute"><Arg>PORT_BUNDLE_HOST_KEY_SWITCH</Arg><Arg>false</Arg></Call>
    <!--
      - This value may be true or false. True is implied by a non zero
      - BUNDLE_LENGTH. If the BUNDLE_LENGTH is non zero, then this value
      - will be ignored. As a BUNDLE_LENGTH of 0 is a legal value, however,
      - the Port Bundle Host Key feature can can also be turned on here
      - when the BUNDLE_LENGTH is 0, which it would be for persistent - connections.
```

```
<Callname="setGlobalAttribute"><Arg>PORT_BUNDLE_HOST_KEY_SWITCH</Arg><Arg>true</Arg></Call
>

      -->

    <!--
      - If we need to map from a client IP address to an SSG explicitly,
      - then we could have an entry like this:

    <Call
name="setSubnetAttribute"><Arg>213.0.0.0</Arg><Arg>255.0.0.0</Arg><Arg>IP</Arg><Arg>195.24
5.182.2</Arg></Call>

      - which would map the client subnet 213.0.0.0 to the SSG at
      - 195.245.182.2 with the global parameters defined above for
      - the RADIUS protocol.
      -->
    <!-- If we need to define a location for a subnet, say London, then we
       - could do this:

    <Call
name="setSubnetAttribute"><Arg>213.0.0.0</Arg><Arg>255.0.0.0</Arg><Arg>SESSION_LOCATION</A
rg><Arg>London</Arg></Call>

      - See the location definitions below for illustrations of how
      - attributes can be associated with locations.
      -->
  </Configure>

  <!-- =============================================================== -->
  <!--
    - Here we define attributes for RADIUS communication with the RADIUS
    - servers for service and group profiles in RADIUS mode.
    -->
  <Configure jmxname="com.cisco.aggbu:name=AAA,connection=ServiceProfile">
    <Set name="throttle" type="int">256</Set>
    <Set name="timeOut" type="int">4</Set>
    <Set name="retryCount" type="int">3</Set>
    <Set name="primaryIP">127.0.0.2</Set>
    <Set name="primaryPort" type="int">1812</Set>
    <Set name="secret">cisco</Set>
    <Set name="secondaryIP">127.0.0.3</Set>
    <Set name="secondaryPort" type="int">1812</Set>
    <Set name="servicePassword">servicecisco</Set>
    <Call name="open"/>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:name=AAA,connection=GroupProfile">
    <Set name="throttle" type="int">256</Set>
    <Set name="timeOut" type="int">4</Set>
    <Set name="retryCount" type="int">3</Set>
    <Set name="primaryIP">127.0.0.2</Set>
    <Set name="primaryPort" type="int">1812</Set>
    <Set name="secret">cisco</Set>
    <Set name="secondaryIP">127.0.0.3</Set>
    <Set name="secondaryPort" type="int">1812</Set>
    <Set name="groupPassword">groupcisco</Set>
    <Call name="open"/>
  </Configure>

  <!-- =============================================================== -->
  <Configure jmxname="com.cisco.aggbu:name=captiveportal">
    <!--
      - This is the URL that the Captive Portal application will redirect
      - to after it has copied the original request URL. It should point
      - to the NWSP application.
      -->
    <Set name="captureToURL">http://localhost:80/decorate/pages/home.jsp</Set>
  </Configure>

  <!-- ================================================================ -->
  <!--
    - These are examples of how arbitrary contetxt properties can be used
    - in the SESM applications.
    -->
  <Configure jmxname="com.cisco.aggbu:context=ssd">
    <!--
      - This section defines sub contexts of the SSD context within which
      - further attributes can be defined below.
      -->
    <Call name="createSubContext"><Arg>options</Arg></Call>
    <Call name="createSubContext"><Arg>location</Arg></Call>
    <Call name="createSubContext"><Arg>brand</Arg></Call>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=options">
    <!--
      - These options control different aspects of the NWSP applications
      - behaviours. These settings are used by the NWSP application to
      - control different aspects of its behaviour.
      -->
    <!-- Use Icons in the service list instead of text. -->
    <Put name="useIcons" type="boolean">TRUE</Put>
    <!-- Confirm that you want to logon onto a service as opposed
       -  to single click logon. -->
    <Put name="confirmAtServiceLogon" type="boolean">FALSE</Put>
    <!-- Confirm that you want to logoff a service as opposed
       - to single click logoff. -->
    <Put name="confirmAtServiceLogoff" type="boolean">TRUE</Put>
    <!-- Confirm that you want to logoff from the application as opposed
       - to single click logoff. -->
    <Put name="confirmAtAccountLogoff" type="boolean">TRUE</Put>
    <!-- This overrides the setting in the Jetty nwsp.xml. -->
    <Put name="sessionTimeOut" type="String">7200</Put>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=location">
    <!-- Here we are defining separate contexts for locations. -->
    <Call name="createSubContext"><Arg>London</Arg></Call>
    <Call name="createSubContext"><Arg>Paris</Arg></Call>
    <Call name="createSubContext"><Arg>NewYork</Arg></Call>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=location,1=London">
    <!-- Here we define attributes for the London location. -->
    <Put name="url">http://www.london.com</Put>
    <Put name="river">Thames</Put>
    <Put name="church">St Pauls</Put>
    <Put name="brand">silver</Put>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=location,1=Paris">
    <!-- Here we define attributes for the Paris location. -->
    <Put name="url">http://www.paris-france.org/</Put>
    <Put name="river">Seine</Put>
    <Put name="church">Notre Dame</Put>
  </Configure>
```

```
  <Configure jmxname="com.cisco.aggbu:context=ssd,0=location,1=NewYork">
    <!-- Here we define attributes for the Hudson location. -->
    <Put name="url">http://www.usa.net/newyork</Put>
    <Put name="river">Hudson</Put>
    <Put name="church">Wall Street</Put>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=brand">
    <!-- Here we are defining separate contexts for brands. -->
    <Call name="createSubContext"><Arg>acme</Arg></Call>
    <Call name="createSubContext"><Arg>cisco</Arg></Call>
    <!-- Silver and gold don't need additional attributes, but we
       - define them here for completeness. -->
    <Call name="createSubContext"><Arg>silver</Arg></Call>
    <Call name="createSubContext"><Arg>gold</Arg></Call>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=brand,1=acme">
    <!-- Here we define attributes for the acme brand. -->
    <Put name="url">http://www.acme.com</Put>
    <Put name="email">support@acme.com</Put>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=brand,1=cisco">
    <!-- Here we define attributes for the cisco brand. -->
    <Put name="url">http://www.cisco.com</Put>
    <Put name="email">support@cisco.com</Put>
  </Configure>

</XmlConfig>
```

DESS Mode Deployment

The following nwsp/config/nwsp.xml file shows a DESS mode deployment with the captive portal feature enabled. RDP was installed in normal (non-proxy) mode, with the Add Services option checked.

```
<?xml version="1.0"  encoding="ISO-8859-1"?>
<!DOCTYPE XmlConfig PUBLIC "-//Cisco Systems//DTD XmlConfig 1.1//EN"
"http://www.cisco.com/aggbu/xmlconfig_1_1.dtd">
<!-- Copyright (c) 2001 by Cisco Systems, Inc. All rights reserved. -->
<!-- This is the container independent configuration for the NWSP web application.
     Container specific configuration can be found at:
         $INSTALLROOT/$CONTAINER/config/nwsp.xml
-->

<XmlConfig>
  <!-- ============================================================= -->
  <Instantiate order="1"
             class="com.cisco.aggbu.jmx.LoggerMBean"
             jmxname="com.cisco.aggbu:name=Logger"/>

  <Instantiate order="99"
               class="com.sun.jdmk.comm.HtmlAdaptorServer"
               jmxname="com.cisco.aggbu:name=ManagementConsole">
      <Arg type="int">
        <SystemProperty name="management.portno"/>
      </Arg>
      <Arg>
<Array class="com.sun.jdmk.comm.AuthInfo">
          <Item>
            <New class="com.sun.jdmk.comm.AuthInfo">
              <Arg>MgmtUser</Arg>
              <Arg>MgmtPassword</Arg>
            </New>
          </Item>
</Array>
      </Arg>
  </Instantiate>

 <!-- =============================================================== -->
  <Configure jmxname="com.cisco.aggbu:name=Logger">
    <Set name="debug" type="boolean"><SystemProperty name="nwsp.debug"
default="false"/></Set>
    <Set name="debugPatterns"></Set>
    <Set name="debugThreads"></Set>
    <Set name="debugVerbosity">LOW</Set>
    <Set name="logDateFormat">yyyyMMdd:HHmmss.SSS</Set>
    <Set name="logFile"><SystemProperty name="application.log"
default="./logs"/>/yyyy_mm_dd.application.log</Set>
    <Set name="logFrame"  type="boolean">false</Set>
    <Set name="logStack"  type="boolean">false</Set>
    <Set name="logThread" type="boolean">true</Set>
    <Set name="logToErr"  type="boolean"><SystemProperty name="nwsp.logToErr"
default="false"/></Set>
    <Set name="trace"     type="boolean">true</Set>
    <Set name="warning"   type="boolean">true</Set>
  </Configure>


  <!-- =============================================================== -->
  <Configure jmxname="com.cisco.aggbu:name=ManagementConsole">
    <Call name="start"/>
  </Configure>

  <!-- =============================================================== -->
  <Configure class="com.cisco.aggbu.ssd.core.model.SSDMBean"
             jmxname="com.cisco.aggbu:name=SSD">
    <Call name="defineMode">
      <Arg>Demo</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.demo.DemoAuthenticationService</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.demo.DemoAuthorizationService</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.demo.DemoConnectionService</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.demo.DemoServiceProfileService</Arg>
    </Call>
    <Call name="defineMode">
      <Arg>RADIUS</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSAuthentication</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSAuthorization</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSConnection</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSServiceProfile</Arg>
    </Call>
    <Call name="defineMode">
      <Arg>DESS</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSAuthentication</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.dess.DESSAuthorizationService</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.radius.RADIUSConnection</Arg>
      <Arg>com.cisco.aggbu.ssd.spis.dess.DESSServiceProfileService</Arg>
    </Call>
    <Set name="mode"><SystemProperty name="ssd.mode" default="DESS"/></Set>
    <Set name="singleSignOn" type="boolean">false</Set>
    <Set name="autoConnect" type="boolean">false</Set>
    <Set name="profileCachePeriod" type="int">600</Set>
  </Configure>

  <!-- =============================================================== -->
  <Configure jmxname="com.cisco.aggbu:name=SSDDemoMode">
```

```
    <!--
      - This is the demo data file. It is in the format of a Merit
      - dictionary with special extensions for this software.
      -->
    <Set name="demoDataFile"><SystemProperty
name="application.home"/>/config/demo.txt</Set>
     <!--
       - This is is an example of using brands in Demo mode.
       - See the definitions for the brands below. The same example
       - can be used in SSG configuration to tie subnets to brands.
       -->
     <Call name="setSubnetAttribute">
       <Arg>127.0.0.0</Arg>
       <Arg>255.0.0.0</Arg>
       <Arg>SESSION_BRAND</Arg>
       <Arg>gold</Arg>
    </Call>
  </Configure>

  <!-- ================================================================ -->
  <!-- Settings for the DESS SPI. -->
  <Configure jmxname="com.cisco.aggbu:name=DESSMode">
    <!-- The time in minutes between checking the authorization tokens. -->
    <Set name="tokenCheckInterval" type="int">5</Set>
    <!-- The age of a token (time since last used) for it to be removed from cache. -->
    <Set name="tokenMaxAge" type="int">10</Set>
  </Configure>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=SSG">
<!--
  - Maxmimum number of simultaneous requests allowed to each SSG. Extra
  - requests will be placed on a queue and issued as responses are received
  - or timeout.
  -->
    <Set name="throttle" type="int">20</Set>

    <!--
      - Here we define attributes for RADIUS communication with the SSG If
      - we are running with Port Bundle Host key then we need only define
      - the global attributes for all of the SSGs.
      -->
    <Call name="setGlobalAttribute"><Arg>PORT</Arg><Arg>1812</Arg></Call>
    <Call name="setGlobalAttribute"><Arg>TIMEOUTSECS</Arg><Arg>10</Arg></Call>
    <Call name="setGlobalAttribute"><Arg>RETRIES</Arg><Arg>3</Arg></Call>
    <Call name="setGlobalAttribute"><Arg>SECRET</Arg><Arg>cisco</Arg></Call>
    <Call name="setGlobalAttribute"><Arg>MASK</Arg><Arg>255.255.255.255</Arg></Call>
    <!--
      - A non zero value here, the default should be 4, will turn Port
      - Bundle Host Key on.
      -->
    <Call name="setGlobalAttribute"><Arg>BUNDLE_LENGTH</Arg><Arg>0</Arg></Call>
    <!-- The following line configures a single non-hostkey SSG              -->
    <!-- Additional SSGs can be configured by adding further 'Call' elements -->
    <!-- Remove the following call if the bundle size is ever set to > 0     -->
    <!-- Arg list: <client subnet>, <subnet mask>, IP, <SSG IP address>      -->
    <Call
name="setSubnetAttribute"><Arg>10.25.0.0</Arg><Arg>255.255.0.0</Arg><Arg>IP</Arg><Arg>10.5
.5.1</Arg></Call>
    <Call
name="setGlobalAttribute"><Arg>PORT_BUNDLE_HOST_KEY_SWITCH</Arg><Arg>false</Arg></Call>
    <!--
      - This value may be true or false. True is implied by a non zero
      - BUNDLE_LENGTH. If the BUNDLE_LENGTH is non zero, then this value
      - will be ignored. As a BUNDLE_LENGTH of 0 is a legal value, however,
      - the Port Bundle Host Key feature can can also be turned on here
      - when the BUNDLE_LENGTH is 0, which it would be for persistent
      - connections.

    <Call
name="setGlobalAttribute"><Arg>PORT_BUNDLE_HOST_KEY_SWITCH</Arg><Arg>true</Arg></Call>

      -->

    <!--
      - If we need to map from a client IP address to an SSG explicitly,
      - then we could have an entry like this:

    <Call
name="setSubnetAttribute"><Arg>213.0.0.0</Arg><Arg>255.0.0.0</Arg><Arg>IP</Arg><Arg>195.24
5.182.2</Arg></Call>

      - which would map the client subnet 213.0.0.0 to the SSG at
      - 195.245.182.2 with the global parameters defined above for
      - the RADIUS protocol.
      -->
    <!-- If we need to define a location for a subnet, say London, then we
       - could do this:

    <Call
name="setSubnetAttribute"><Arg>213.0.0.0</Arg><Arg>255.0.0.0</Arg><Arg>SESSION_LOCATION</A
rg><Arg>London</Arg></Call>

      - See the location definitions below for illustrations of how
      - attributes can be associated with locations.
      -->
  </Configure>

  <!-- =============================================================== -->
  <!--
    - Here we define attributes for RADIUS communication with the RADIUS
    - servers for service and group profiles in RADIUS mode.
    -->
  <!-- Uncomment and modify this element when run in RADIUS mode
  <Configure jmxname="com.cisco.aggbu:name=AAA,connection=ServiceProfile">
    <Set name="throttle" type="int">256</Set>
    <Set name="timeOut" type="int">4</Set>
    <Set name="retryCount" type="int">3</Set>
    <Set name="primaryIP">127.0.0.1</Set>
    <Set name="primaryPort" type="int">1812</Set>
    <Set name="secret">cisco</Set>
    <Set name="secondaryIP">127.0.0.2</Set>
    <Set name="secondaryPort" type="int">1812</Set>
    <Set name="servicePassword">servicecisco</Set>
    <Call name="open"/>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:name=AAA,connection=GroupProfile">
    <Set name="throttle" type="int">256</Set>
    <Set name="timeOut" type="int">4</Set>
    <Set name="retryCount" type="int">3</Set>
    <Set name="primaryIP">127.0.0.1</Set>
    <Set name="primaryPort" type="int">1812</Set>
    <Set name="secret">cisco</Set>
    <Set name="secondaryIP">127.0.0.2</Set>
    <Set name="secondaryPort" type="int">1812</Set>
    <Set name="groupPassword">groupcisco</Set>
    <Call name="open"/>
```

```
</Configure>
  -->

<!-- ================================================================ -->
<Configure jmxname="com.cisco.aggbu:name=captiveportal">
  <!--
    - This is the URL that the Captive Portal application will redirect
    - to after it has copied the original request URL. It should point
    - to the NWSP application.
    -->
  <Set name="captureToURL">http://localhost:80/decorate/pages/home.jsp</Set>
</Configure>

<!-- ================================================================ -->
<!--
  - These are examples of how arbitrary contetxt properties can be used
  - in the SESM applications.
  -->
<Configure jmxname="com.cisco.aggbu:context=ssd">
  <!--
    - This section defines sub contexts of the SSD context within which
    - further attributes can be defined below.
    -->
  <Call name="createSubContext"><Arg>options</Arg></Call>
  <Call name="createSubContext"><Arg>location</Arg></Call>
  <Call name="createSubContext"><Arg>brand</Arg></Call>
</Configure>

<Configure jmxname="com.cisco.aggbu:context=ssd,0=options">
  <!--
    - These options control different aspects of the NWSP applications
    - behaviours. These settings are used by the NWSP application to
    - control different aspects of its behaviour.
    -->
  <!-- Use Icons in the service list instead of text. -->
  <Put name="useIcons" type="boolean">TRUE</Put>
  <!-- Confirm that you want to logon onto a service as opposed
     -  to single click logon. -->
  <Put name="confirmAtServiceLogon" type="boolean">FALSE</Put>
  <!-- Confirm that you want to logoff a service as opposed
     - to single click logoff. -->
  <Put name="confirmAtServiceLogoff" type="boolean">TRUE</Put>
  <!-- Confirm that you want to logoff from the application as opposed
     - to single click logoff. -->
  <Put name="confirmAtAccountLogoff" type="boolean">TRUE</Put>
  <!-- This overrides the setting in the Jetty nwsp.xml. -->
  <Put name="sessionTimeOut" type="String">7200</Put>
</Configure>

<Configure jmxname="com.cisco.aggbu:context=ssd,0=location">
  <!-- Here we are defining separate contexts for locations. -->
  <Call name="createSubContext"><Arg>London</Arg></Call>
  <Call name="createSubContext"><Arg>Paris</Arg></Call>
  <Call name="createSubContext"><Arg>NewYork</Arg></Call>
</Configure>

<Configure jmxname="com.cisco.aggbu:context=ssd,0=location,1=London">
  <!-- Here we define attributes for the London location. -->
  <Put name="url">http://www.london.com</Put>
  <Put name="river">Thames</Put>
  <Put name="church">St Pauls</Put>
  <Put name="brand">silver</Put>
</Configure>
```

```
  <Configure jmxname="com.cisco.aggbu:context=ssd,0=location,1=Paris">
    <!-- Here we define attributes for the Paris location. -->
    <Put name="url">http://www.paris-france.org/</Put>
    <Put name="river">Seine</Put>
    <Put name="church">Notre Dame</Put>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=location,1=NewYork">
    <!-- Here we define attributes for the Hudson location. -->
    <Put name="url">http://www.usa.net/newyork</Put>
    <Put name="river">Hudson</Put>
    <Put name="church">Wall Street</Put>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=brand">
    <!-- Here we are defining separate contexts for brands. -->
    <Call name="createSubContext"><Arg>acme</Arg></Call>
    <Call name="createSubContext"><Arg>cisco</Arg></Call>
    <!-- Silver and gold don't need additional attributes, but we
       - define them here for completeness. -->
    <Call name="createSubContext"><Arg>silver</Arg></Call>
    <Call name="createSubContext"><Arg>gold</Arg></Call>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=brand,1=acme">
    <!-- Here we define attributes for the acme brand. -->
    <Put name="url">http://www.acme.com</Put>
    <Put name="email">support@acme.com</Put>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:context=ssd,0=brand,1=cisco">
    <!-- Here we define attributes for the cisco brand. -->
    <Put name="url">http://www.cisco.com</Put>
    <Put name="email">support@cisco.com</Put>
  </Configure>

</XmlConfig>
```

Sample RDP MBean Configuration File

An example rdp.xml file follows. See Appendix E, “RDP Packet Handlers,” for more information about this MBean and the possibilities for extending RDP functionality with customized packet handlers.

Note The contents of this MBean is different depending on the options you checked during RDP installation. (The packet handlers are different.) The following file shows RDP installed in normal (non-proxy) mode, with the Add Services option checked.

```
<?xml version="1.0"  encoding="ISO-8859-1"?>
<!DOCTYPE XmlConfig PUBLIC "-//Cisco Systems//DTD XmlConfig 1.1//EN"
"http://www.cisco.com/aggbu/xmlconfig_1_1.dtd">
<!-- Copyright (c) 2001 by Cisco Systems, Inc. All rights reserved. -->
<!-- This is the container independent configuration for the RDP application.
     Container specific configuration can be found at:
         $INSTALLROOT/$CONTAINER/config/rdp.xml
-->

<XmlConfig>
  <!-- ================================================================= -->
  <Instantiate order="1"
               class="com.cisco.aggbu.jmx.LoggerMBean"
               jmxname="com.cisco.aggbu:name=Logger" />

  <Instantiate order="97"
               class="com.cisco.aggbu.rdp.RDPPacketFactoryMBean"
               jmxname="com.cisco.aggbu:name=RDPPacketFactory" />

  <Instantiate order="98"
               class="com.cisco.aggbu.rdp.RDPMBean"
               jmxname="com.cisco.aggbu:name=RDP" />

  <Instantiate order="96"
               class="com.sun.jdmk.comm.HtmlAdaptorServer"
               jmxname="com.cisco.aggbu:name=ManagementConsole">
      <Arg type="int">
        <SystemProperty name="management.portno"/>
      </Arg>
      <Arg>
<Array class="com.sun.jdmk.comm.AuthInfo">
          <Item>
            <New class="com.sun.jdmk.comm.AuthInfo">
              <Arg>MgmtUser</Arg>
              <Arg>MgmtPassword</Arg>
            </New>
          </Item>
</Array>
      </Arg>
  </Instantiate>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=Logger">
    <Set name="debug" type="boolean"><SystemProperty name="rdp.debug"
default="false"/></Set>
    <Set name="debugPatterns"></Set>
    <Set name="debugThreads"></Set>
    <Set name="debugVerbosity">LOW</Set>
    <Set name="logDateFormat">yyyyMMdd:HHmmss.SSS</Set>
    <Set name="logFile"><SystemProperty name="application.log"
default="./logs"/>/yyyy_mm_dd.application.log</Set>
    <Set name="logFrame"  type="boolean">false</Set>
    <Set name="logStack"  type="boolean">false</Set>
    <Set name="logThread" type="boolean">true</Set>
    <Set name="logToErr"  type="boolean"><SystemProperty name="rdp.logToErr"
default="false"/></Set>
    <Set name="trace"     type="boolean">true</Set>
    <Set name="warning"   type="boolean">true</Set>
  </Configure>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=ManagementConsole">
    <Call name="start"/>
  </Configure>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=RDPPacketFactory">
    <Call name="addType">
      <!-- The untyped handler looks for the service type AV in the packer to
         - determine whether the request is for a service profile (service
         - type == outbound) or a user profile (no service type)-->
      <Arg>Untyped</Arg>
      <Arg>com.cisco.aggbu.rdp.UntypedPacket</Arg>
    </Call>
    <Call name="addType">
```

```
      <!-- There are six user logon handlers; userLogonPacket (authenticates),
         - UserLogonFramedPacket (authenticates and adds a Service-type=2
         - (Framed user) ), UserLogonFramedAddServicesPacket (authenticates
         - and adds a Service-type=2 and services, i.e. authorizes),
         - UserLogonAddServices (authenticates and authorizes),
         - UserProxyAuthPacket (authenticates via a proxy) and
         - UserProxyAuthAddServicePacket (authenticates via a proxy and
         - authorizes) -->
      <Arg>UserLogon</Arg>
      <Arg>com.cisco.aggbu.rdp.UserLogonFramedPacket</Arg>
    </Call>
    <Call name="addType">
      <Arg>ProfileRequest</Arg>
      <!-- Attempts to match the password to the PASSWORD: attribute and
         - return the matching value -->
      <Arg>com.cisco.aggbu.rdp.ProfileRequestPacket</Arg>
    </Call>
    <!-- Following attribute and type handle service profiles -->
    <Call name="setAttribute">
      <Arg>PASSWORD:servicecisco</Arg>
      <Arg>ServiceRequest</Arg>
    </Call>
    <Call name="addType">
      <Arg>ServiceRequest</Arg>
      <Arg>com.cisco.aggbu.rdp.ServiceProfilePacket</Arg>
    </Call>
    <!-- Following attribute and type handle group profiles -->
    <Call name="setAttribute">
      <Arg>PASSWORD:groupcisco</Arg>
      <Arg>GroupRequest</Arg>
    </Call>
    <Call name="addType">
      <Arg>GroupRequest</Arg>
      <Arg>com.cisco.aggbu.rdp.GroupProfilePacket</Arg>
    </Call>
    <!-- Following attribute and type handle next hop profiles -->
    <Call name="setAttribute">
      <Arg>PASSWORD:nexthopcisco</Arg>
      <Arg>NextHopRequest</Arg>
    </Call>
    <Call name="addType">
      <Arg>NextHopRequest</Arg>
      <Arg>com.cisco.aggbu.rdp.NextHopPacket</Arg>
    </Call>
    <Call name="addType">
      <Arg>Unknown</Arg>
      <!-- Does not respond to the request -->
      <Arg>com.cisco.aggbu.rdp.DiscardPacket</Arg>
    </Call>
    <!-- Example use of a Proxy handler.
         String after ';' is name of AAA connection (see AAAMBean below)
    <Call name="addType">
      <Arg>ProxyNextHop</Arg>
      <Arg>com.cisco.aggbu.rdp.ProxyPacket;Proxy</Arg>
    </Call>
    -->
  </Configure>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=RDP">
<Set id="RDPSecret" name="secret">cisco</Set>
<Set name="localIPAddress">10.5.5.3</Set>
    <Set name="localPort" type="int"><SystemProperty name="application.portno"
default="1812"/></Set>
```

```
    <Set name="minThreads" type="int">10</Set>
    <Set name="maxThreads" type="int">256</Set>
    <Set name="maxIdleTimeMs" type="int">10000</Set>
    <Call name="startRDP"/>
  </Configure>

  <!-- ================================================================ -->
  <!-- Uncomment and modify this element when run in proxy mode
  <Configure jmxname="com.cisco.aggbu:name=AAA,connection=Proxy">
    <Set name="throttle" type="int">256</Set>
    <Set name="timeOut" type="int">4</Set>
    <Set name="retryCount" type="int">1</Set>
    <Set name="primaryIP">127.0.0.2</Set>
    <Set name="primaryPort" type="int">1812</Set>
    <Set id="AAASecret" name="secret">cisco</Set>
    <Set name="secondaryIP">127.0.0.3</Set>
    <Set name="secondaryPort" type="int">1812</Set>
    <Call name="open"/>
  </Configure>
    -->

</XmlConfig>
```

Sample CDAT MBean Configuration File

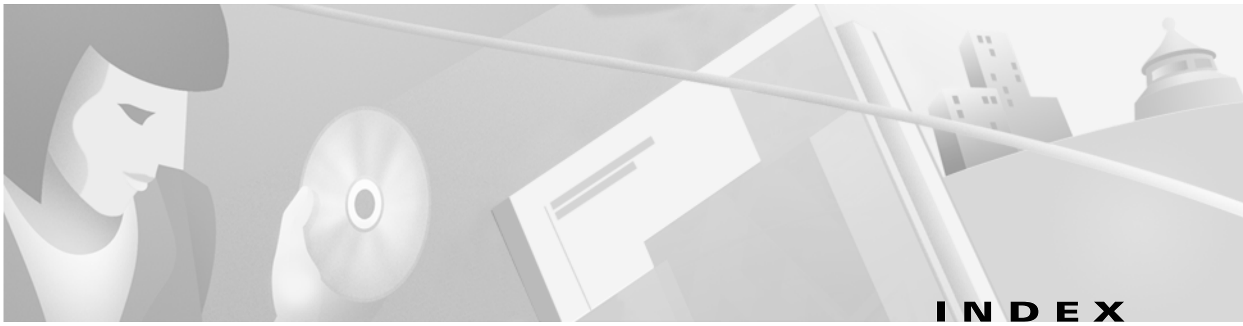
An example cdat.xml file follows.

```
<?xml version="1.0"  encoding="ISO-8859-1"?>
<!DOCTYPE XmlConfig PUBLIC "-//Cisco Systems//DTD XmlConfig 1.1//EN"
"http://www.cisco.com/aggbu/xmlconfig_1_1.dtd">
<!-- Copyright (c) 2001 by Cisco Systems, Inc. All rights reserved. -->
<!-- This is the container independent configuration for the CDAT web application.
     Container specific configuration can be found at:
         $INSTALLROOT/$CONTAINER/config/cdat.xml
-->

<XmlConfig>
  <!-- ================================================================ -->
  <Instantiate order="1"
               class="com.cisco.aggbu.jmx.LoggerMBean"
               jmxname="com.cisco.aggbu:name=Logger" />

  <Instantiate order="99"
               class="com.sun.jdmk.comm.HtmlAdaptorServer"
               jmxname="com.cisco.aggbu:name=ManagementConsole">
      <Arg type="int">
        <SystemProperty name="management.portno"/>
      </Arg>
      <Arg>
<Array class="com.sun.jdmk.comm.AuthInfo">
          <Item>
            <New class="com.sun.jdmk.comm.AuthInfo">
              <Arg>MgmtUser</Arg>
              <Arg>MgmtPassword</Arg>
            </New>
          </Item>
</Array>
      </Arg>
  </Instantiate>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=Logger">
    <Set name="debug" type="boolean"><SystemProperty name="cdat.debug"
default="false"/></Set>
    <Set name="debugPatterns"></Set>
    <Set name="debugThreads"></Set>
    <Set name="debugVerbosity">LOW</Set>
    <Set name="logDateFormat"><SystemProperty name="cdat.logDateFormat"
default="HHmmss.SSS"/></Set>
    <Set name="logFile"><SystemProperty name="application.log"
default="./logs"/>/yyyy_mm_dd.application.log</Set>
    <Set name="logFrame"  type="boolean">false</Set>
    <Set name="logStack"  type="boolean">false</Set>
    <Set name="logThread" type="boolean">false</Set>
    <Set name="logToErr"  type="boolean"><SystemProperty name="cdat.logToErr"
default="false"/></Set>
    <Set name="trace"     type="boolean">true</Set>
    <Set name="warning"   type="boolean">true</Set>
  </Configure>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=ManagementConsole">
    <Call name="start"/>
  </Configure>

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=CDAT">
    <Set name="sessionTimeout" type="int">600</Set>
    <Set name="maxVariables" type="int">40</Set>
    <Set name="queryMaxResults" type="int">500</Set>
    <Set name="queryTimeout" type="int">0</Set>
  </Configure>

</XmlConfig>
```

Sample DESS MBean Configuration File

An example DESS configuration file (config.xml) follows:

```
<?xml version="1.0"  encoding="ISO-8859-1"?>
<!DOCTYPE XmlConfig PUBLIC "-//Cisco Systems//DTD XmlConfig 1.1//EN"
"http://www.cisco.com/aggbu/xmlconfig_1_1.dtd">
<!-- Copyright (c) 2001 by Cisco Systems, Inc. All rights reserved. -->
<!-- This is the dess-auth configuration -->

<XmlConfig>
  <!-- ================================================================ -->
  <Instantiate order="2"
               class="com.cisco.aggbu.dessauth.ConnectionMBean"
               jmxname="com.cisco.aggbu:name=Directory,type=Connection,instance=Primary"
  />

  <Instantiate order="2"
               class="com.cisco.aggbu.dessauth.ConnectionMBean"
               jmxname="com.cisco.aggbu:name=Directory,type=Connection,instance=Secondary"
  />

  <Instantiate order="3"
               class="com.cisco.aggbu.dessauth.DirectoryMBean"
               jmxname="com.cisco.aggbu:name=Directory" />

  <!-- ================================================================ -->
  <Configure jmxname="com.cisco.aggbu:name=Directory,type=Connection,instance=Primary">
<Set name="poolSize" type="int">2</Set>
<Set name="URL">ldap://10.0.0.2:389/</Set>
<Set name="principal">cn=admin,ou=sesm,o=cisco</Set>
<Set name="credentials">cisco</Set>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:name=Directory,type=Connection,instance=Secondary">
<Set name="poolSize" type="int">2</Set>
<Set name="URL">ldap://10.0.0.2:389/</Set>
<Set name="principal">cn=admin,ou=sesm,o=cisco</Set>
<Set name="credentials">cisco</Set>
  </Configure>

  <Configure jmxname="com.cisco.aggbu:name=Directory">
<Set name="connectionNameRoot">com.cisco.aggbu:name=Directory,type=Connection,*</Set>
<Set name="factory">com.cisco.cns.security.jndi.JNDIConnection</Set>
<Set name="context">ou=sesm,o=cisco</Set>
<Set name="DESSPrincipal">cn=admin,ou=sesm,o=cisco</Set>
<Set name="alwaysGetAllAttributes" type="boolean">false</Set>

    <Set name="traceFileName"><SystemProperty name="application.log"
default="./logs"/>/dess.log</Set>
    <Set name="traceLevel">NONE</Set>
    <Set name="printTraceToConsole" type="boolean">false</Set>
    <Set name="stackTrace" type="boolean">false</Set>

<Set name="cacheMaxObjects" type="int">50000</Set>

<!-- Save at least cacheMinFreeMem% VM memory.          -->
<!-- i.e. Cache can occupy 100-cacheMinFreeMem% memory  -->
<Set name="cacheMinFreeMem" type="int">10</Set>

<!-- All timeout values are in seconds -->
<Set name="cacheSessionTimeout" type="int">600</Set>
<Set name="cacheExpireInterval" type="int">600</Set>
<Set name="cacheObjectTimeout" type="int">600</Set>
<Call name="commit"/>
  </Configure>
</XmlConfig>
```



INDEX