Solaris for ISPs Administration Guide

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303
U.S.A.

Part No: 805–4874
17 July 1998

Contents

Preface xi

1. Solaris for ISPs Overview 1

Solaris for ISPs Architecture 1

Solaris for ISPs Platform Extensions 2

Sun Internet Administrator 2

Host Configuration Software 3

Sun Internet Services Monitor 3

Sun Directory Services 4

Sunscreen SKIP 4

FlexLM License Server 5

HotJava Browser 5

Java Development Kit 5

ISP Services 6

SWS 6

Sun Internet News Server 7

Sun Internet FTP Server 7

How Solaris for ISPs Installs 8

Host Configuration Model 8

Repeatable Configuration 9

Managing Services with Sun Internet Administrator 10

Features for a Secure Environment 10

Three-Tier Application Architecture 11

Two-Tier Application Architecture 12

2. How to Plan Your Installation 15

The Installation Process 15

Installation Process Overview 16

Planning the Installation 18

Preparing for Installation 19

Installing Solaris for ISPs 21

Configuring Sun Directory Services 22

Configuring Sun Internet Administrator 23

Post-installation Tasks 24

3. Setup Guidelines 27

Install Scenario 27

Network Setups 27

Security Issues 30

Standard Internet Security Risks 30

Security Risks in Solaris for ISPs 31

How To Tighten Security 32

Things to Consider 32

Changes to Solaris 32

Log File Management 38

Creating User-Defined Scripts 39

Restoring Default AWS Settings 40

4. Installation Requirements 41

Installation Prerequisites 41

Operating System Requirements 42

Hardware Requirements 43

Web Browser Supported 45

Component Dependencies 45

5. Using Directory Services 47

Solaris for ISPs Directory Structure 48

OSI Tree Structure 48

DC Tree Structure 50

Making Directory Entries from the Command Line 51

Creating Directory Entries: General Procedure 51

Structure of an `ldif(4)` File 52

Creating Domain Entries 53

Creating Subscriber Entries 54

Creating Group Entries 59

Solaris for ISPs Access Control 60

Rules Enabling Sun Internet Administrator Functionality 61

Rules Enabling Service Functionality 61

Rules Enabling Proper User Access 61

6. Solaris for ISPs Directory Services Schema 63

Maintaining the Schema 63

What to Back Up 64

Restoring the Schema 64

▼ Restoring the Solaris for ISPs Schema 64

Solaris for ISPs Object Classes 65

`ispAdministrator` Class 66

`ispManagedService` Class 67

`ispService` Class 69

`ispSubscriber` Class 71

Solaris for ISPs Attributes 72

associatedDomain Attribute 73
associatedName Attribute 73
commonName Attribute 73
description Attribute 74
gidNumber Attribute 74
homeDirectory Attribute 74
host Attribute 74
ispAdministeredService Attribute 74
ispAuthorizedServices Attribute 75
ispCategory Attribute 75
ispContentDirectory Attribute 75
ispDirectoryRoot Attribute 75
ispImageFile Attribute 76
ispPrivateData Attribute 76
ispServiceContext Attribute 76
ispServiceLocation Attribute 76
ispServlets Attribute 76
ispServletClasspath Attribute 77
ispParameterizedOperation Attribute 77
ispSupplementaryInformation Attribute 77
ispSupportedOperation Attribute 77
ispVersion Attribute 78
labeledURI Attribute 78
mail Attribute 78
objectClass Attribute 78
ou Attribute 79
surname Attribute 79
uidNumber Attribute 79

`userCertificate` Attribute 79

`userid` Attribute 79

`userPassword` Attribute 79

7. Integrating Existing Service Applications 81

General Steps to Integrating an Existing Service 81

Integrating X-Based Services 82

Integrating Command-Line Programs 83

Integrating Two-Tier Web-Based Applications 84

Registering Information for a Two-Tier Web-Based Application 84

Configuring for Administrator Account Coordination 85

Index 87

Figures

Figure 1–1 Basic architecture of Solaris for ISPs 1

Figure 1–2 Solaris for ISPs Host Configuration Process 9

Figure 1–3 Three-Tier ISP Service Architecture 11

Figure 1–4 Two-Tier ISP Service Architecture 12

Figure 2–1 Installation Process 16

Figure 2–2 Planning the Installation 18

Figure 2–3 Preparing for the Installation 19

Figure 2–4 Installing Solaris for ISPs 21

Figure 2–5 Configuring Sun Directory Services 22

Figure 2–6 Configuring Sun Internet Administrator 23

Figure 2–7 Post-installation Tasks 24

Figure 3–1 Base Configuration 28

Figure 3–2 Expanded Configuration 29

Figure 5–1 Solaris for ISPs OSI Tree 48

Figure 5–2 OSI Tree Entries 49

Figure 5–3 Domain Structure in the OSI Tree 50

Figure 5–4 Solaris for ISPs DC Tree 50

Figure 5–5 Directory Structure with a Domain Added 54

Preface

Solaris for ISPs Administration Guide provides information about Solaris™ for ISPs™ platform extensions and services. It discusses a wide range of topics, providing an overview of the software and examining the directory services schema, preinstallation requirements, and postinstallation scripts.

Who Should Read This Book

The audience for this book includes system administrators; the person who maintains multi-user computer systems, or a network; and individuals adding and configuring new workstations, setting up user accounts, and installing system-wide software.

How This Book Is Organized

This book contains the following chapters:

Chapter 1, "Solaris for ISPs Overview," discusses the features of Solaris for ISPs platform extensions and services.

Chapter 2, "How to Plan Your Installation," examines the various steps in the installation of Solaris for ISPs platform extensions and services.

Chapter 3, "Setup Guidelines," discusses the guidelines for configuring hosts for the installation of Solaris for ISPs platform extensions and services.

Chapter 4, "Installation Requirements," examines the host server requirements for the installation of Solaris for ISPs platform extensions and services.

Chapter 5, "Using Directory Services," describes the required directory structure and making entries in the directory from the command line.

Chapter 6, "Solaris for ISPs Directory Services Schema," provides reference material on the object classes and attributes added to the Sun™ Directory Services schema.

Chapter 7, “Integrating Existing Service Applications,” examines the requirements for integrating existing service applications with Sun™ Internet Administrator™.

Solaris™ for ISPs™ Related Books

Documentation related to Solaris for ISPs includes printed manuals, AnswerBooks, and man pages. Each is listed below.

Manuals

The documents listed here are provided in printed version. These manuals are also available as AnswerBooks™ and as on line help files on the CD.

- *Solaris for ISPs Administration Guide* part # 805-4874 (the book you are currently reading)
- *Solaris for ISPs Installation Guide* part # 805-4875
- *Solaris for ISPs Command Reference*, part # 805-4876

AnswerBook

These documents are provided as AnswerBook™ on the CD.

- *Solaris for ISPs Administration Guide*, part # 805-5032
- *Solaris for ISPs Installation Guide*, part # 805-5031
- *Sun Directory Services 3.1 Administration Guide*
- *Sun Directory Services 3.1 User’s Guide*

Man Pages

Each manual page, commonly known as a “man” page, discusses one subject, such as a user command or library function.

The location of Solaris for ISPs platform extensions and services man pages are listed below:

- The JDK 1.1.5™ man pages are located in `/usr/share/man`.
- The FLEXLm man pages are located in `/opt/SUNWste/license_tools/man`.
- The Sun™ Internet Administrator™ console man pages are located in `/opt/SUNWixamc/man`.
- The Sun™ Directory Services man pages are located in `/opt/SUNWconn/man`.
- The Sunscreen™ SKIP man pages are located in `/opt/SUNWicg/man`.
- The host configuration man pages are located in `/opt/SUNWisp/man`.
- The Sun™ Internet FTP Server™ man pages are located in `/opt/SUNWixfta/1.0/man`.
- The Sun™ Internet News Server™ man pages are located in `/opt/SUNWsns/man`.
- The Sun™ WebServer™ (SWS) man pages are located in `/usr/share/man`.

The README *File*

The Solaris for ISPs `README.1st` file is a short file on the product CD that contains late breaking news, bugs, release information, and pointers to software readme files and documents for installing.

Other Related Documents

You may also want to consult the following books related to the subject matter discussed in this book.

- *Automating Solaris Installations (A Custom JumpStart Guide),* by Paul Anthony Kasper and Alan L. McClellan, SunSoft Press, 1995.
- *Solaris Advanced Installation Guide,* part # 802–5740–10.

Related Web Sites

You may also want to consult the following Web sites for information on the subject matter discussed in this book.

- `http://access1.sun.com/Products/ISP/` for the required and recommended patches.
- `http://skip.incog.com/spec/SKIP.html` for SKIP documentation.
- `http://www.javasoft.com/products/jdk/1.1/index115.html` for JDK 1.1.5 documentation.

- `http://www.nai.com/default_pgp.asp` for PGP information.
- `http://www.isc.org/` for reference on implementation of core Internet protocols.
- `http://www.sun.com/servers/ultra_enterprise/sw/webstart/` for JumpStart installation.
- `http://sunsolve1.sun.com/` for public patches.

Ordering Sun Documents

The SunDocs[SM] program provides more than 250 manuals from Sun Microsystems, Inc. If you live in the United States, Canada, Europe, or Japan, you can purchase documentation sets or individual manuals using this program.

For a list of documents and how to order them, see the catalog section of SunExpress™ on the Internet at `http://www.sun.com/sunexpress`.

Definitions of Typefaces

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

Typeface or Symbol	**Meaning**	**Example**
`AaBbCc123`	The names of commands, files, and directories	Edit your `.login` file. Use `ls -a` to list all files.
AaBbCc123	Command-line placeholder: replace with a real name or value	To delete a file, type `rm` *filename*
AaBbCc123	Book Titles, new words or terms, or words to be emphasized	Read Chapter 6 in *User's Guide*. These are called *class* options. You *must* be root to do this.

CHAPTER 1

Solaris for ISPs Overview

Welcome to Solaris™ for ISPs™. This Solaris overpack builds upon the open architecture and scalability of the Solaris operating system to provide the optimum operating environment for Internet service providers and their customers.

Solaris for ISPs Architecture

Solaris for ISPs is organized into two collections of software. The platform software extends the Solaris foundation, adding features that enable ISP services but are not directly accessed by ISP subscribers. The ISP services provide subscriber functionality such as Internet news, FTP, and World Wide Web access while taking advantage of the extended Solaris environment.

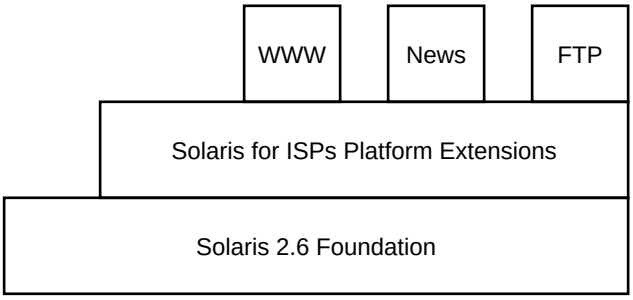


Figure 1–1 Basic architecture of Solaris for ISPs

Solaris for ISPs Platform Extensions

Solaris for ISPs includes the following enhancements to the Solaris operating system:

- Sun™ Internet Administrator™
- Host configuration software
- Sun™ Internet Services Monitor™
- Sun™ Directory Services
- Sunscreen™ SKIP
- FlexLM License Server
- HotJava™ browser
- Java Development Kit

Sun Internet Administrator

Sun Internet Administrator provides secure central management for distributed ISP services. It gives ISP administrators the following features:

- *Single sign-on for administrators.* ISP administrators log onto Sun Internet Administrator once to access all functions for which they have authorization. Services developed according to ISP guidelines and managed from Sun Internet Administrator receive log-on information from it; the user is not subsequently challenged.
- *Secure communications between administrators' client machines and remote service hosts.* The optional SKIP software can be installed and configured on all connections to the console, and from the console to the service host machines, making those communications snoop-proof and spoof-proof.
- *Logging of administrator actions for traceability.* Each administrator action, from initial log-on attempt through logout, is logged via the `syslog` utility. This provides both troubleshooting and accountability information.
- *Remote management of existing ISP services.* Service components provided with Solaris for ISPs can all be managed from the Sun Internet Administrator, regardless of their location on the network. Additionally, Sun™ Internet FTP Server™ and Sun™ Internet News Server™ are three-tier components and receive all the security benefits built into Sun Internet Administrator. See "Managing Services with Sun Internet Administrator" on page 10 for more information on service interaction with Sun Internet Administrator.
- *Extensibility for existing services.* ISPs can integrate their own applications with Sun Internet Administrator and manage them in the same way as services provided

with Solaris for ISPs. See Chapter 7 for instructions on integrating applications with Sun Internet Administrator.

Host Configuration Software

The Solaris for ISPs host configuration software provides the following functionality:

- *Software installation.* Administrators install and uninstall all Solaris for ISPs software using the host configuration software. Administrators can save installation scenarios for use in a JumpStart finish script to repeat installations automatically.
- *Solaris foundation configuration.* To improve security and conserve resources, unneeded Solaris services are disabled. Security-related components of Solaris are configured appropriately for an ISP environment.
- *Intrusion detection.* Periodically, the intrusion detector checks its log file, determining whether any failed log-on attempts have occurred since the last check. If an intrusion attempt has occurred, the detector collects the logged data and passes it to the user-specified notification mechanism (such as electronic mail).
- *Server process management.* This `cron` job ensures that server processes (such as news servers) are indeed running. If any server has stopped abnormally, the server process manager starts that server.
- *Log file management.* Audit and syslog logs are cycled daily. The log file management daemon archives logs weekly and deletes any archive older than one month.

The host configuration software is a required software component. It is installed on every Solaris for ISPs host machine.

Sun Internet Services Monitor

The performance monitoring software allows an ISP to set up special client machines that emulate a subscriber's experience with the ISP services. The performance monitoring applet can be set to connect to any combination of Web, mail, news, and directory services servers and collect information on their performance from a subscriber's perspective. This data is collected on the monitoring host machine and viewable with a Web browser.

Sun Internet Services Monitor is a two-tier application. It is manageable through Sun Internet Administrator, but does not receive the benefits of single sign-on or administrator authentication. See "Managing Services with Sun Internet Administrator" on page 10 for more information on the two-tier architecture.

Sun Directory Services

This Lightweight Directory Access Protocol (LDAP) implementation provides a shared repository for both user (administrator) and service configuration information. Administrators store subscriber information in the repository as well. Features in this release of Sun Directory Services include:

- Conformance to LDAP v3 Internet standards.
- A Remote Access Dialup User Service (RADIUS) server that provide authentication for remote users connecting to the network through a Network Access Server (NAS).
- A Network Information System (NIS) server that integrates into an existing NIS environment to provide an integrated naming service.
- A complete suite of administration tools, including the Deja directory editor, a Java—implemented administration console for management of the directory, and a Web gateway for access from any browser.

Sun Directory Services is manageable from Sun Internet Administrator as an X-based application.

Sun Directory Services installs with a no-license limitation of 1K (one thousand) entries in the directory. A license certificate for 5K (five thousand) entries ships with Solaris for ISPs and must be redeemed and registered with the FlexLM license server before it takes effect. See the instructions in the *Solaris for ISPs Installation Guide* for details of redeeming and installing the license certificate.

See Chapter 5 and Chapter 6 of this book for information about the role of Sun Directory Services in Solaris for ISPs. The Sun Directory Services documentation consists of two books, *Sun Directory Services 3.1 Administration Guide* and *Sun Directory Services 3.1 User's Guide*, both delivered as AnswerBook2™ packages. The Sun Directory Services Deja tool also has full on-line help.

Sunscreen SKIP

Sunscreen™ SKIP is based on the Simple Key-management for Internet Protocols (SKIP) standard of key management for IP encryption. Characteristics of SKIP include:

- Automatic certificate exchanges
- Sessionless protocols
- Multicast and unicast packet protocols for IPv4 and IPv6
- Certificate Discovery Protocol (CDP)

The full SKIP technology is available only in North America, but a version exists for export to other parts of the world. When SKIP is installed, its manual pages are located at `/opt/SUNWicp/man`.

FlexLM License Server

The FlexLM license server is used by Sun Directory Services to manage licenses of various sizes. If you already have a license server in your network (version 4.1 or later), you can use it to serve Sun Directory Services licenses.

Sun Directory Services allows 1K (one thousand) entries before requiring a license. This is sufficient to install and initialize the directory. In any reasonable ISP application, however, more entries will quickly be required. Follow the directions in the *Solaris for ISPs Installation Guide* for acquiring a license key and configuring the server.

After it is installed, the FlexLM manual pages are located at `/opt/SUNWste/licene_tools/man`.

HotJava Browser

The HotJava browser is provided with Solaris for ISPs to support Sun Internet Administrator and other administration user interfaces in the product. It supports the following Internet standards and protocols:

- Java Development Kit 1.1.6
- HTTP 1.1 Protocol
- HTML 3.2
- Tables and Frames
- Persistent Cookies
- GIF and JPEG Media Formats
- AU Audio Format
- FTP and Gopher File Transfer Protocols
- SMTP and MIME E-mail Protocols
- SOCKS Protocol
- Secure Sockets Layer (SSL) 3.0
- Java Archive (JAR) Format

Java Development Kit

The Java Development Kit (JDK) is provided with Solaris for ISPs to support the use of Java in the product. JDK version 1.1.5 includes the following new capabilities:

- Internationalization
- Signed applets

- JAR file format
- AWT (window toolkit) enhancements
- JavaBeans™ component model
- Networking enhancements
- Math package for large numbers
- Remote Method Invocation (RMI)
- Reflection
- Database connectivity (JDBC)

ISP Services

Services in this version of Solaris for ISPs include:

- Sun™ WebServer™ (SWS)
- Sun™ Internet News Server™
- Sun™ Internet FTP Server™

In addition to any graphical user interfaces, all ISP services also provide full command-line access for scripting.

SWS

SWS is a highly reliable, secure, standards-based Web server for accessing, managing, and distributing information over the Internet, extranets, or intranets. Features added in this release of SWS include:

- *Support for HTTP/1.1.* SWS supports the latest version of the hypertext transfer protocol, including named virtual hosts and content negotiation.
- *Enhanced scalability.* Multiple instances of the server process can run on a single machine and can be managed through the same administration graphical user interface. The number of available virtual hosts is thus raised geometrically.
- *Java servlet support.* Using servlets rather than the more usual CGI scripts provides the cross-platform advantages of Java on the server side while improving processing speed.
- *Secure HTTP communications.* SWS includes secure-socket layer (SSL) functionality and support for Verisign certificates for safe, encrypted communications.
- *Microsoft FrontPage support.* SWS supports popular FrontPage extensions in the areas of authoring, administration, and dynamic content.

SWS is a two-tier application. It is manageable through the Sun Internet Administrator, but does not receive the benefit of single sign-on. It has been configured to share administrator data with Sun Internet Administrator. See Chapter 7 for details on this configuration and “Managing Services with Sun Internet Administrator” on page 10 for more information on the two-tier architecture.

Sun Internet News Server

Sun Internet News Server is a high-performance, highly-scalable news server. Significant features include:

- *High performance and client-connection scalability.* A multithreaded, multi-process daemon handles client connections, taking advantage of multiprocessor Solaris servers to provide scalability in handling large numbers of simultaneous news reader connections.
- *Full-featured news feed handling.* Based on INN from the Internet Software Consortium (INN release 1.5 sec. 2), Sun Internet News Server maintains all the usability improvements of that implementation in the area of news feed handling.
- *Separate feed handling and news reader service functionality.* The service for news readers is separable from the feed handling functions to enable horizontal scalability in handling news reader connections.
- *Centralized, browser-based management.* Sun Internet News Server is a three-tier service that integrates with Sun Internet Administrator for centralized administration and full security benefits. See “Managing Services with Sun Internet Administrator” on page 10 for more information on the three-tier architecture.

Sun Internet FTP Server

This scalable, high-performance FTP server offers the following enhancements:

- *Multiple domains on a single host.* Sun Internet FTP Server supports IP-based virtual hosting. Each virtual host has its own configuration files that tune the server for that particular domain.
- *Configurable user authentication.* Sun Internet FTP Server can use either UNIX accounts or entries in the SunDS registry to authenticate administrators.
- *Centralized, browser-based management.* Sun Internet FTP Server is a three-tier service that integrates with Sun Internet Administrator for centralized administration and full security benefits. See “Managing Services with Sun Internet Administrator” on page 10 for more information on the three-tier architecture.

How Solaris for ISPs Installs

Because the typical UNIX server must run a variety of applications, the default Solaris installation assumes that most UNIX services are needed. ISPs focus more narrowly on providing specific services in a public environment. They have heavy performance and security requirements.

To configure Solaris to their needs, ISP administrators typically perform elaborate hardening and tuning tasks. They stop unneeded Solaris services and change file permissions to close security vulnerabilities. This process can take hours.

The host configuration software in Solaris for ISPs automates this hardening and tuning process for the administrator. In addition to copying the necessary software packages to their proper locations, it hardens the underlying Solaris 2.6 foundation, changing file owners and modes where appropriate as well as configuring Solaris security and logging mechanisms. A final step in this process is selectively disabling standard Solaris services (such as `finger` or `rlogin`) when they do not support the purpose of a given host machine.

Solaris for ISPs host configuration can be performed interactively by using its graphical user interface, or repeatably and non-interactively using JumpStart.

Host Configuration Model

The configuration process works by building a scenario of the current state of the system, what software components are available to be installed, and what the user has selected for install or uninstall.

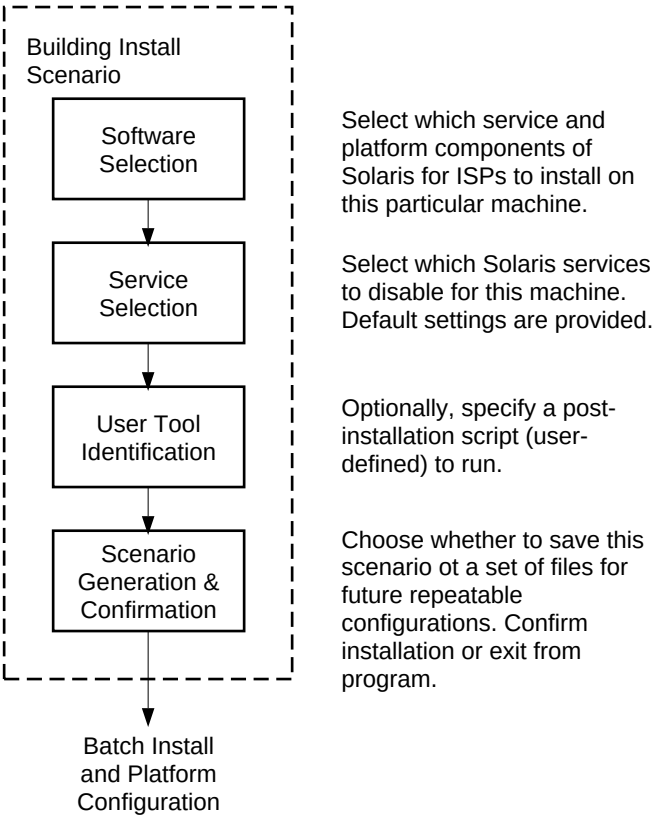


Figure 1–2 Solaris for ISPs Host Configuration Process

The host configuration software can also be used to reconfigure a host after installation, adding and removing services as needed.

Repeatable Configuration

Interactive host configuration (using the graphical user interface) provides the option to save a configuration scenario (in the form of a binary and some associated files). By creating and saving a scenario, the ISP administrator can use it in a JumpStart™ finish script, forming a non-interactive, one-step installation. Such JumpStart installations are repeatable and can be used to configure identically.

JumpStart is a part of the Solaris operating system that can perform customized, repeatable installations of Solaris both locally and remotely. See the *Solaris Advanced Installation Guide* for details on how to create a custom JumpStart installation. See Chapter 2 of this guide for information on how to use a scenario file in a finish script for a custom JumpStart installation.

Managing Services with Sun Internet Administrator

Sun Internet Administrator provides secure centralized management for all ISP services, both locally and across a network of hosts. It launches the administration GUIs (where present) of individual services upon request from an authorized administrator. Command-line interfaces can also be accessed where appropriate.

Features for a Secure Environment

Sun Internet Administrator provides the following security features:

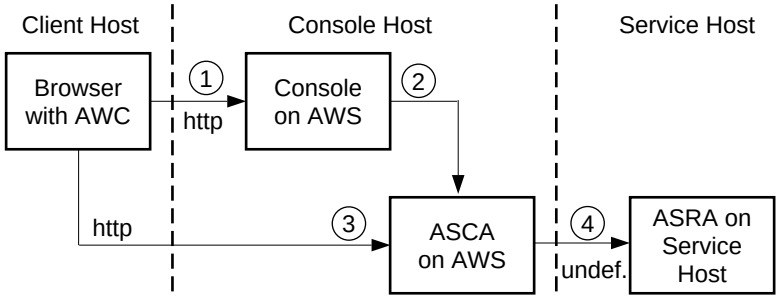
- *Administrator authentication.* Administrators are required to supply a valid user name and password when accessing the GUI.
- *Administrator access control.* Access is controlled per ISP service. An administrator allowed to manage FTP servers on the network may or may not also have access to news servers. Console administrators (those who can manage Sun Internet Administrator processes) have access to all services managed by Sun Internet Administrator.
- *Central auditing.* Administrators' actions are logged for traceability and accountability.
- *Privacy and integrity protection for all network traffic.* The optional SKIP software can be configured to protect all connections to and from Sun Internet Administrator. SSL can also be used for secure HTTP traffic.

Sun Internet Administrator supports services in two architectures: three-tier and two-tier. Only the three-tier architecture receives all of the above-listed security benefits. Four types of service UIs are supported:

- *Three-tier, browser-based* applications receive all security benefits offered by Sun Internet Administrator.
- *Two-tier, browser-based* applications cannot make use of the single sign-on feature, but are manageable through the Sun Internet Administrator. If they use SWS to support the administration application, they can configure it to provide administrator authentication. (See Chapter 7 for details on this configuration.) The two-tier architecture is included to support legacy applications.
- *X-based* applications receive all the benefits of a three-tier application.
- *Command-line* functions (scripts, programs, or in combination) receive all the benefits of three-tier applications. Any number of them can be registered for a given service and managed by Sun Internet Administrator, which constructs a Web interface to the command-line programs.

Three-Tier Application Architecture

The recommended three-tier browser-based application architecture receives all Sun Internet Administrator security benefits.



Legend

AWC: Administration Web Client. That part of the Sun Internet Administrator GUI that is downloaded to the administrator's browser.

AWS: Administration Web Server. The Web server that serves the Sun Internet Administrator GUI and any ASCAs present for managed services.

ASCA: Administration Server Client Agent. The portion of a managed service's user interface co-resident with the AWS.

ASRA: Administration Server Remote Agent. The portion of a managed service's user interface co-resident with the service.

Figure 1–3 Three-Tier ISP Service Architecture

As shown in Figure 1–3, an administrator uses the following steps to access a service's administration functions:

1. From a browser, the administrator requests a specific URL (the location of the main Sun Internet Administrator GUI page).

   The AWC is downloaded to the client browser, where the administrator can choose a service to manage.

2. Sun Internet Administrator prompts the administrator for user name and password. The administrator need not use a UNIX account for access to the console GUI; a directory services repository (Sun Directory Services) manages administrator information for Sun Internet Administrator. This connection should be secured by using secure HTTP.

   The selected service resolves to an URL, designating the services's ASCA. The server agent GUI is downloaded in response. At this step, control passes to the service's administration program.

3. Subsequent access is directly between the client browser and the component's server agent on the AWS.

   The AWS authenticates the administrator against the directory services, and logs each administrator request. If the administrator has appropriate access, requests are passed to the ASRA.

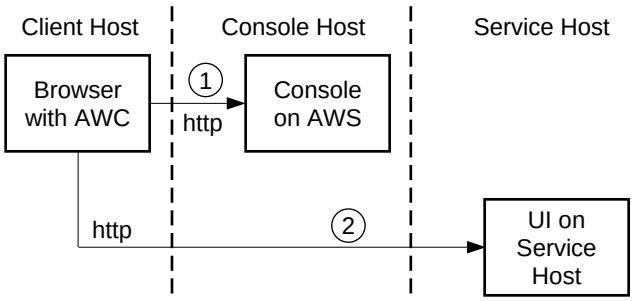
4. The ASCA communicates with the ASRA via a protocol chosen by the developer of the service. Appropriate IP-level security measures should be taken to protect this connection and its traffic.

   The ASRA again authenticates and logs each administrator action. To protect the network communications, the ISP can add IP-level encryption, if that is desired, by using SKIP.

ASCA and ASRA modules for command line and X-based programs are provided in Solaris for ISPs. Sun Internet Administrator uses them automatically when you register these applications.

Two-Tier Application Architecture

For some applications, especially existing services, a two-tier architecture for access via Sun Internet Administrator is more practical. These services can be managed from Sun Internet Administrator, but do not receive the security benefits of single sign-on and logging.



Legend

AWC: Administration Web Client. That part of the Sun Internet Administrator GUI that is downloaded to the administrator's browser.

AWS: Administration Web Server. The Web server that serves the Sun Internet Administrator GUI and any ASCAs present for managed services.

Figure 1–4 Two-Tier ISP Service Architecture

As shown in Figure 1–4, an administrator uses the following steps to access a service's administration functions:

1. From a browser, the administrator requests a specific URL (the location of the main console GUI page).

   This step is the same as for the three-tier architecture. The AWC is downloaded to the client browser, where the administrator can choose a service to manage.

2. The selected service resolves to a URL, designating the component's ASRA. If the service's administration GUI is not browser-based, other protocols may be used at the developer's option.

3. Subsequent access is directly between the client browser and the service's remote agent. Appropriate IP-level security measures should be taken to protect this connection and its traffic.

   In a two-tier architecture, services are not able to take advantage of the single sign-on feature.

CHAPTER 2

How to Plan Your Installation

For the installation of Solaris™ for ISPs™, careful planning is required. This chapter presents a high-level overview of the various steps involved in the installation of Solaris for ISPs platform extensions and services. For detailed instructions on installing the software, refer *Solaris for ISPs Installation Guide*.

The Installation Process

This section describes the steps in the installation process in sequential flowcharts:

- Planning the Installation.
- Preparing for Installation.
- Installing Solaris™ for ISPs™.
- Configuring Sun™ Directory Services.
- Configuring Sun™ Internet Administrator™.
- Post-installation Tasks

Note - For each step discussed in the process, please refer to the chapter indicated.

Installation Process Overview

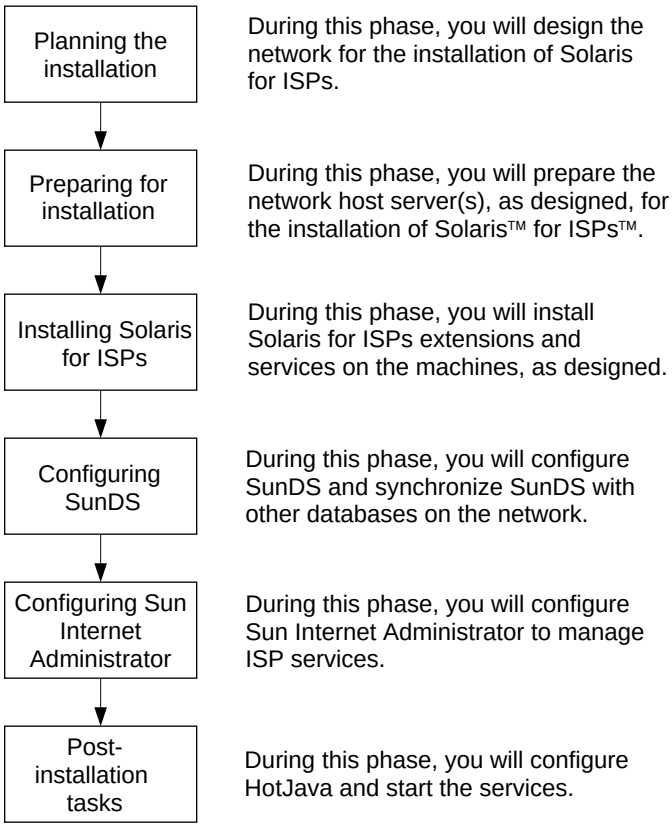


Figure 2–1 Installation Process

You *must* perform the steps discussed here. This section examines the steps in the installation process illustrated in Figure 2–1.

- *Step 1*: During this phase of the process, you will read Chapter 1 and Chapter 3 to plan the installation of Solaris for ISPs.

  The goal of this step is to:

  - Understand the features of Solaris for ISPs components.
  - Design network host setup and designate role for Solaris for ISPs host server.

- *Step 2*: During this phase of the process, you will refer to "*Solaris for ISPs Platform Extensions*" on page 44 to prepare network hosts, as you had designed in the planning phase.

  The goal of this step is to ensure that your network hosts meet:

  - Specified hardware requirements.

- Specified operating system requirements.

Note - You will repeat these steps to prepare all the hosts on the network to install Solaris for ISPs platform extensions and services.

- *Step 3*: During this phase of the process, you will refer to "*Solaris for ISPs Installation Guide*" and install Solaris for ISPs platform extensions and services on the network hosts, as designed in the planning phase.

  The goal of this step is to ensure

  - Successful installation of Solaris for ISPs platform extensions.
  - Successful installation of Solaris for ISPs services.

  Note - You must install the Solaris for ISPs Platform software on every machine. At least one installation of Sun Internet Administrator and Sun Directory Services is required to support the Solaris for ISPs services.

- *Step 4*: During this phase of the process, you will refer to *Sun Directory Services Administration Guide* and configure Sun Directory Services.

  The goal of this step is to:

  - Start Sun Directory Services.
  - Synchronize Sun Directory Services with other independent subscriber databases on the network.

- *Step 5*: During this phase of the process, you will refer to on-line help and configure Sun Internet Administrator.

  The goal of this step is to:

  - Customize Sun Internet Administrator to suit your environment.
  - Register and manage services from Sun Internet Administrator.

- *Step 6*: During this phase of the process, you will refer to *Solaris for ISPs Installation Guide* and configure HotJava™ browser and start the services.

  The goal of this step is to:

  - Configure HotJava to support applet security requirements for Sun Internet Administrator and Sun™ Internet Services Monitor™.
  - Access the services through Sun Internet Administrator or directly from a browser and start the services.

Planning the Installation

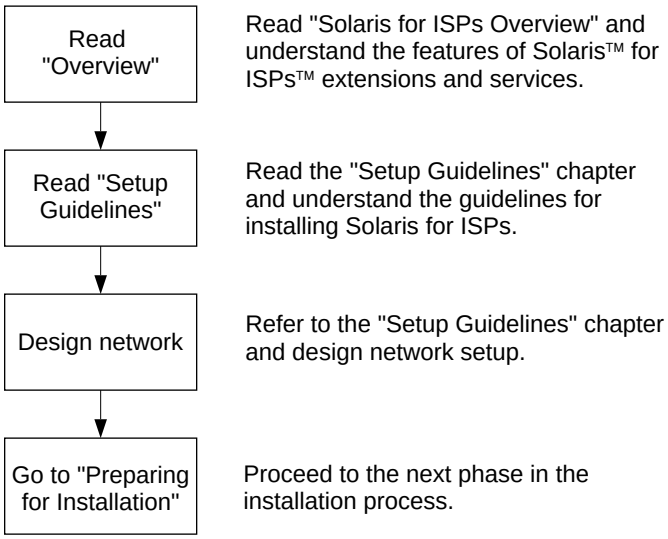


Figure 2–2 Planning the Installation

You *must* perform the steps discussed here. This section examines the steps discussed in the planning phase of the installation process as illustrated in Figure 2–2.

- *Step 1*: Read Chapter 1. The goal of this step is to ensure that you:

  - Understand the various features of Solaris for ISPs platform extensions and services.
  - Understand how Solaris for ISPs installs.

  The Solaris for ISPs overview documentation is available in several formats for your convenience.

  - At the root directory of your installation media, the overview documentation is at *see media_root/docs/README.1st for the location*.
  - In the printed documentation, this information is in Chapters 1 through 4 of this book.
  - On the World Wide Web, you can access this information at `http://access1.sun.com/Products/ISP`.

- *Step 2*: Read Chapter 3. The goal of this step is to assist you to:

  - Plan for the installation.
  - Prepare for the installation.

- *Step 3*: Design your Solaris for ISPs network. See “Install Scenario” on page 27 to design network. The goal of this step is to ensure that you:

- Design your network setup
- Designate network host servers for Solaris for ISPs platform extensions.
- Designate network host servers for Solaris for ISPs services.

- *Step 4*: Proceed to prepare for the installation of Solaris for ISPs.

Preparing for Installation

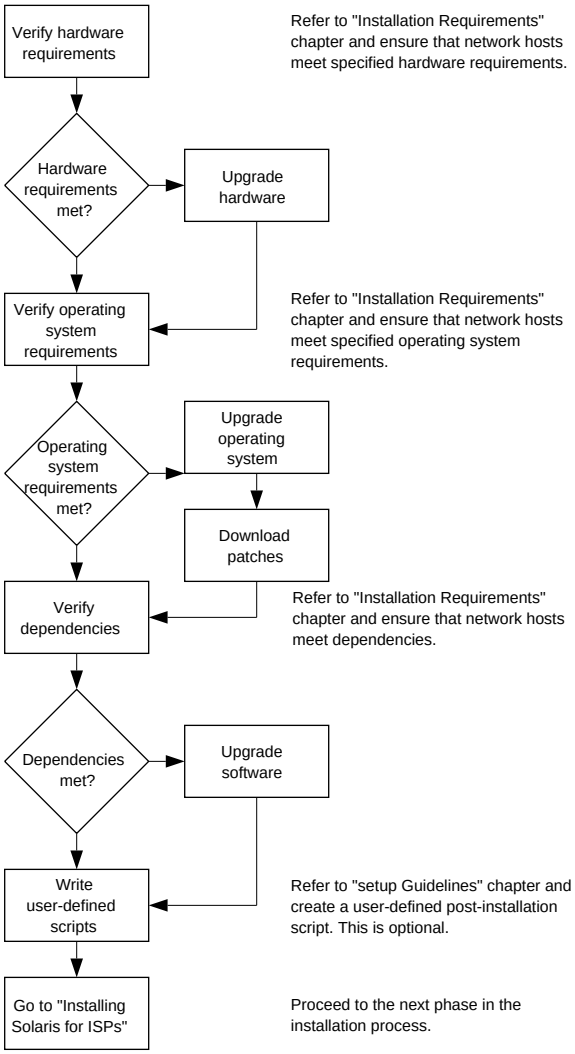


Figure 2–3 Preparing for the Installation

You *must* perform the steps discussed here. This section examines the steps in the preparing phase of the installation process as illustrated in Figure 2–3.

Note - You *must* repeat these steps to prepare all the hosts on the network to install Solaris for ISPs platform extensions and services.

- *Step 1*: Verify system hardware requirements for Solaris for ISPs platform extensions and services. Refer to “Hardware Requirements” on page 43 in “*Solaris for ISPs Platform Extensions*” on page 44.

  The goal of this step is to ensure that your Solaris for ISPs network host servers meet:

  - Specified CPU requirements.
  - Specified disk space requirements.
  - Specified RAM requirements.
  - Specified SWAP requirements.

- *Step 2*: Verify operating system requirements for Solaris for ISPs platform extensions and services. Refer to “Installation Prerequisites” on page 41 in “*Solaris for ISPs Platform Extensions*” on page 44.

  The goal of this step is to ensure that you:

  - Meet specified operating system requirements.
  - Download specified operating system patches.

- *Step 3*: Verify Solaris for ISPs platform extensions and services dependencies. See “Component Dependencies” on page 45 in Chapter 4.

  The goal of this step is to ensure that, for successful installation and functioning of Solaris for ISPs software,

  - All specified bundled and independent package dependencies are met.
  - All inter-software dependencies have been noted. This will enable you to install and configure Solaris for ISPs platform extensions and services based on their inter-dependency.

  Note - If you are from the browser, the component dependencies are displayed on the graphical user interface. If you are installing from the command line, see *Solaris for ISPs Installation Guide*.

- *Step 4*: Write a post-installation script to be executed after installation of Solaris for ISPs. This is optional. For guidelines to write the script, refer to “Creating User-Defined Scripts” on page 39, in Chapter 3.

  The goal of this step is to :

- Write a post-installation script.
- Supply a path to the script.

- *Step 5*: Proceed to installing Solaris for ISPs platform extensions and services on the respective network hosts.

Installing Solaris for ISPs

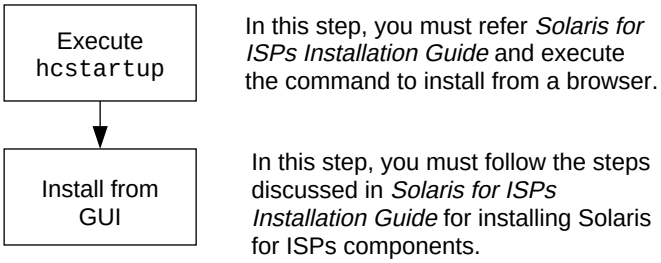


Figure 2–4 Installing Solaris for ISPs

You *must* perform the steps discussed here. This section examines the steps in the installation phase of the process as illustrated in Figure 2–4.

- *Step 1*: Execute `hcstartup` and start the host configuration tool. Refer to *Solaris for ISPs Installation Guide* to start host configuration tool.
- *Step 2*: Install from the Graphical User Interface. To install,

  - Select extension(s) and service(s) to install on the host from the list of Solaris for ISPs platform extensions and services.

Note - You *must* install the Solaris for ISPs Platform component. Check component dependencies on other Solaris for ISPs components for the services you have selected.

Note - Upon selecting the platform extension or service, you can customize the selected component to suit your environment. If you do not customize, you are accepting the default settings.

  - Select Solaris services to enable or disable from the list of Solaris services. See "Reconfigurable Settings" on page 34 in Chapter 3 for more information on enabling or disabling Solaris services.
  - Specify the path to the post-installation script you wrote during preparation for installation. This script will be executed during batch installation.
  - Optionally, save the configured scenario. If you choose to save the scenario, you can use this saved scenario for JumpStart™ installation of Solaris for ISPs

platform extensions and services on other hosts on the network. See Chapter 1 for more information on repeatable installations.

- Install Solaris for ISPs platform extensions and services. Refer to *Solaris for ISPs Installation Guide* or on line help.

Configuring Sun Directory Services

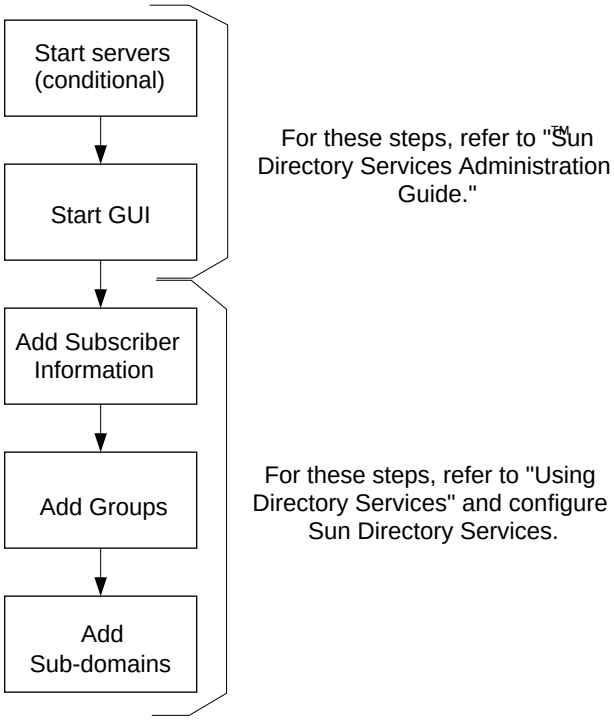


Figure 2–5 Configuring Sun Directory Services

You *must* perform the steps discussed here. This section examines the steps in the configuring phase of the installation process. You *must* first configure Sun Directory Services as illustrated in Figure 2–5.

- *Step 1*: Start Sun Directory Services RADIUS, administration, and Web gateway servers. To start, refer Chapter 3 of the *Solaris for ISPs Installation Guide*.

Note - You must manually start the services only if the servers have not already been started after rebooting the machine after installation.

- *Step 2*: Start browser.

- *Step 3*: Add subscriber information in Sun Directory Services. Synchronize subscriber information in Sun Directory Services with any other independent subscriber database on the network. Refer to “Creating Subscriber Entries” on page 54.
- *Step 4*: Add group entries in Sun Directory Services. See Chapter 5.
- *Step 5*: Add required domains below root domain. Refer “Creating Domain Entries” on page 53.

Note - You can also refer to the Sun Internet Administrator on line help for steps to create domains and to add subscriber and group entries.

Configuring Sun Internet Administrator

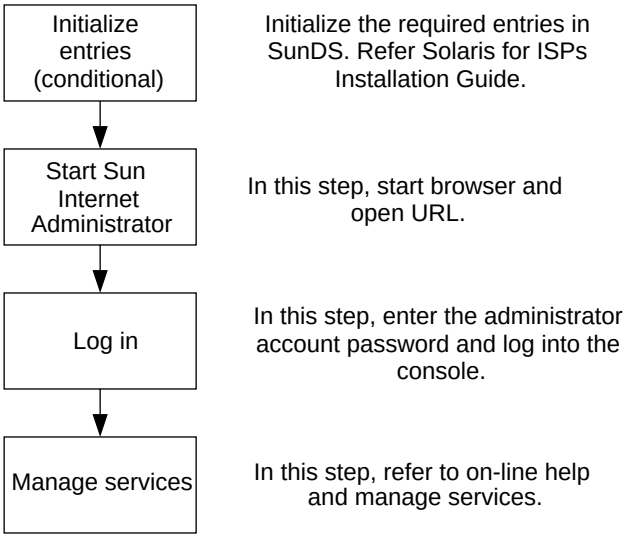


Figure 2–6 Configuring Sun Internet Administrator

You *must* perform the steps discussed here. This section examines the steps in the configuration phase of the installation process. After configuring Sun Directory Services, you must configure Sun Internet Administrator as illustrated in Figure 2–6.

- *Step 1*: Run `mcdsinit` to initialize the required entries in Sun Directory Services. Refer Chapter 3 of the *Solaris for ISPs Installation Guide* to make the initial entries.

Note - You must run this command after installation only if the entries have not already been automatically made for you.

- *Step 2*: Start Sun Internet Administrator console GUI. To start console, go to `http://`*console host name*`:50080/ispmc`. Specify the Sun Internet Administrator host name and the port number, if you installed on a port other than the default port.
- *Step 3*: Enter Sun Internet Administrator account password and log into the console.
- *Step 4*: Manage and administer services. To manage services, you must
  - Register Solaris for ISPs services. See on-line help.
  - Create Administrator accounts. This is optional. To create administrator accounts, see on-line help.

Post-installation Tasks

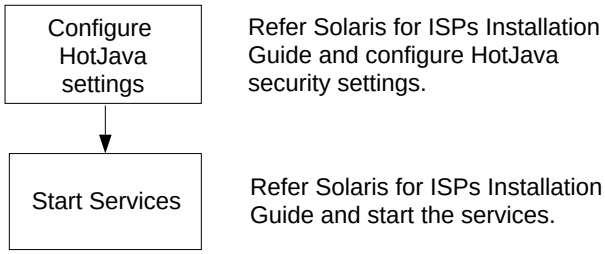


Figure 2–7 Post-installation Tasks

You *must* perform the steps discussed here. This section examines the post-installation tasks that you must perform.

- *Step 1*: Configure HotJava security settings. Refer *Solaris for ISPs Installation Guide* and configure HotJava security settings to support:
  - Browser security requirements for Sun Internet Administrator remote administration applets.
  - Browser security setting requirements for Sun™ Internet Services Monitor™ Java applets.
- *Step 2*: Start the services. Refer Chapter 3 of the *Solaris for ISPs Installation Guide* and access the services:
  - From Sun Internet Administrator. If you want to access the services from Sun Internet Administrator, first register the services and follow steps in Starting Sun Internet Administrator section in Chapter 3 of the *Solaris for ISPs Installation Guide.*
  - Directly using a browser. If you want to access the service user interface directly from a browser, verify the browser supported by the service and refer

to the service on line help to start running the service. To access the service user interface, refer Chapter 3 of the *Solaris for ISPs Installation Guide* for the service URL.

CHAPTER 3

Setup Guidelines

Solaris™ for ISPs™ provides a core infrastructure with a set of platform extensions and services for ISPs. This chapter discusses the guidelines for configuring your network host for the installation of Solaris for ISPs extensions and services. This configuration information is essential for successful installation. Please read carefully.

Install Scenario

You *must* design your network before installing Solaris for ISPs. This section discusses two tested examples of a Solaris for ISPs network hosts setup. You may use the network hosts setup example that most closely suits your environment.

Network Setups

This section describes a base and an expanded example network setup, and the requirements and recommendations for the hardware configuration of the setups.

Note - We do not assume the existence of a firewall in our example network setups. If you are using an Internet firewall product to control network traffic to or from any Solaris for ISPs host, you should examine the security policy controlling this host to make sure the relevant types of communication are allowed. This document does *not* offer recommendations related to Internet firewalls.

Base Setup

This figure illustrates a base example setup.

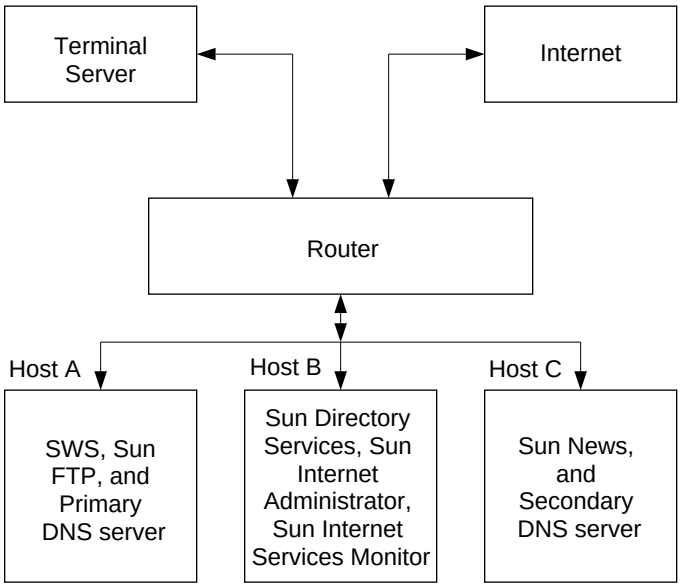


Figure 3–1 Base Configuration

Base Hardware Requirements

- Three high-end workstations.
- Primary and secondary DNS servers.
- The hosts *must* be on a network connected to the Internet.
- You can use any server on the network to act as a client host.
- You need a Web browser on the client host for Sun Internet Administrator and Sun Internet Services Monitor client software.

Base Example Setup Design

The three servers are referred to as Host A, Host B, and Host C. For example, the host servers may be configured as follows:

- *Host A*: Configure this server to act as a service host. Install SWS, Sun Internet FTP Server, and configure the host as a primary DNS server.
- *Host B*: Configure this server to act as a console host. Install Sun Internet Administrator, Sun Directory Services, and Sun Internet Services Monitor.

- *Host C*: Configure this server to act as a service host. Install Sun Internet News Server and configure the host as a secondary DNS server.

Note - The DNS server configurations discussed in this example setup are suggestions only. You need not install Solaris for ISPs extensions and services on a host acting as a DNS server. Most extensions and services only require the ability to perform name lookups regularly.

Expanded Setup

This figure illustrates an expanded example network setup.

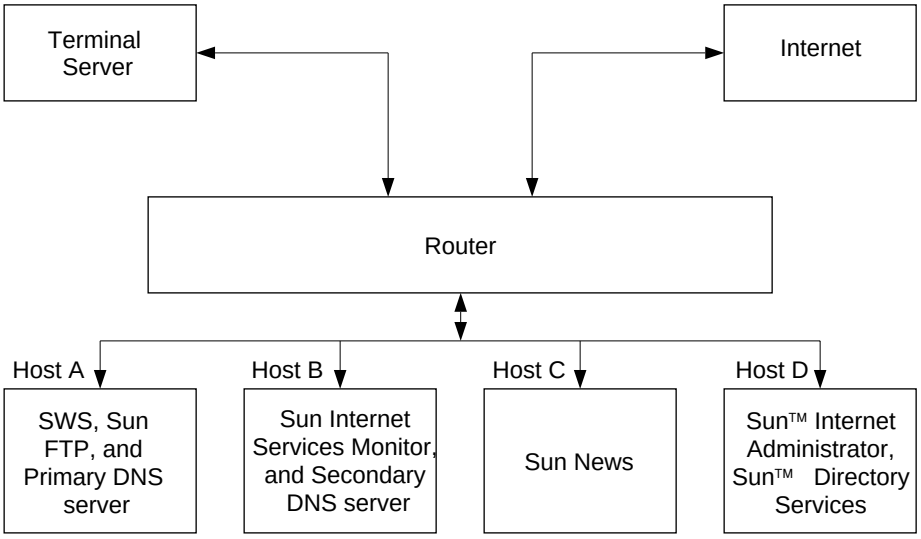


Figure 3–2 Expanded Configuration

Expanded Hardware Requirements

- Four high-end workstations
- Primary and secondary DNS servers
- The hosts *must* be on a network connected to the Internet.
- You can use any server on the network to act as a client host.
- You need a Web browser on the client host for Sun Internet Administrator and Sun Internet Services Monitor client software.

Expanded Example Setup Design

The four servers are referred to as Host A, Host B, Host C, and Host D. For example, the host servers may be configured as follows:

- *Host A*: Configure this server to act as a service host. Install SWS, Sun Internet FTP Server, and configure the host as a primary DNS server.
- *Host B*: Configure this server to act as a console host. Install Sun Internet Services Monitor, and configure the host as a secondary DNS server.
- *Host C*: Configure this server to act as a service host. Install Sun Internet News Server.
- *Host D*: Configure this server to act as a console host. Install Sun Internet Administrator and Sun Directory Services on this host.

Note - The DNS server configurations shown in this example setup are suggestions only. You need not install Solaris for ISPs extensions and services on a host acting as a DNS server. Most extensions and services only require the ability to perform name lookups regularly.

Note - After designing the network hosts setup, prepare the hosts for the installation of Solaris for ISPs. Refer to "*Solaris for ISPs Platform Extensions*" on page 44 to prepare the hosts as designed.

Security Issues

Any network with Internet access exposes the network to some security risks. In addition, some Solaris for ISPs features introduce others. In every case, there are several security measures that can be implemented to protect your network. The following sections discuss some good system administration practices and Solaris for ISPs features that will enable you to secure your network from these risks.

Standard Internet Security Risks

Connecting the network to other networks on the Internet exposes your network to potential service interruptions, unauthorized intrusion, and considerable damage. This section discusses such standard network security risks that you must be aware of. Protections against these risks are discussed in "How To Tighten Security" on page 32.

- Denial of Service Attacks: These attacks disable the system from serving customers and make a service unavailable for the customer. For example, the attacks can

flood the network with useless traffic resulting in inability to serve customers. Most often, such attacks could crash the system or just make the system really slow in serving customers.

- Buffer Overrun Exploits: These include exploiting the software weakness to add arbitrary data into a program, which when run as root, may give the exploiter root access to your system. This may also result in a denial of service attack.
- Snooping and Replay Attacks: The snooping attacks involve an intruder listening to traffic between two machines on your network. The traffic may include passing unencrypted passwords back and forth while using `telnet`, `rlogin`, or `ftp`. This might result in an unauthorized individual breaking into your network or reading confidential data.
- IP Spoofing: Attacks based on IP spoofing involve unauthorized access to computers. The intruder listening to your network traffic finds an IP address of a trusted host, and sends messages indicating that the message is coming from that trusted host.
- Internal Exposure: Most network break-ins are the result of a malicious or disgruntled present or former employee misusing access to information or breaking into your network.

Security Risks in Solaris for ISPs

This section discusses some Solaris for ISPs features that leave the software open to some security risks. Please refer to “How To Tighten Security” on page 32 for protection against these risks.

- Administration Products: Solaris for ISPs administration products for individual services, such as ftp or news, provides new paths for accessing privileged operations. This intruder, by knowing, guessing, or cracking the password of the administrator, may change the behavior of the services by exploiting their administrator interface through the network. However, Sunscreen™ SKIP, bundled with Solaris for ISPs, authenticates incoming traffic and ensures that outgoing data is not viewed by others while in transit.
- Remote Command Execution Mechanism: Sun Internet Administrator provides access to all of the command-line administration commands through its administrator console remote execution mechanism. An intruder may break this mechanism and gain access to those commands. However, access to these commands can be restricted, by root, to registered system administrators only.
- Sun Directory Services: This Solaris for ISPs software can be used to store and manage passwords and information about other Solaris for ISPs extensions and services. An intruder may break in and exploit access to such privileged information. However, most passwords in Sun Directory Services are encrypted. Unencrypted passwords in Sun Directory Services require root access.

How To Tighten Security

To protect your system from unauthorized users accessing, corrupting, or changing information, and to make the network available to authorized users:

- Regularly change passwords and encourage using not-easy-to-guess passwords. Solaris for ISPs software forces all passwords to change periodically if local files are used for password management.
- Use public-key cryptography to encrypt all traffic between trusted hosts at the IP level. SKIP, bundled with Solaris for ISPs, authenticates incoming IP traffic and ensures that outgoing data is not altered or viewed by others while in transit.
- Use routers that can identify trusted hosts and block spoofed IP addresses.
- Fix vulnerabilities and bulletproof your software.
- Disable unwanted services that open your network to security risks. Solaris for ISPs host configuration software, as part of the initial installation process, can disable some 'r' commands to ensure protection for passwords and to restrict access to hosts for unauthorized individuals.
- Provide employees access only to the data or information their work requires.
- Implement security mechanisms such as network monitoring and firewalls.

Things to Consider

This section discusses certain Solaris for ISPs default features that you *must* consider and address during host configuration. In this section, topics include:

- Changes to Solaris—System files modified by Solaris for ISPs.
- Log File Management—Resident daemon of Solaris for ISPs.
- User-Defined Script—Creating post-installation scripts.
- Default Administration Web Server Setting—Restoring AWS default setting.

Changes to Solaris

This section discusses the onetime only changes and the reconfigurable changes that may be made to Solaris services during host configuration. If you accept the default installation setting, these changes will be made on the host where Solaris for ISPs is installed.

Note - You *must* review and may modify, if necessary, these changes to foundation Solaris during host configuration. These changes may not be incorporated in future releases of Solaris for ISPs.

Onetime Only Settings

Solaris for ISPs consists of a foundation configuration unit that runs only once to ensure security for passwords and to safeguard file permissions to the file owner. It makes a set of default changes as part of the initial installation process. The functionality of this unit is similar to the functionality of the script in `ftp://ftp.wins.uva.nl:/pub/solaris/fix-modes.tar.gz`. To undo these changes, go to "Undoing the Changes" on page 37.

This section examines the initial installation steps automatically executed only once in the foundation package. You *must* address this section before installing Solaris for ISPs.

Note - The script *will* be executed. However, these changes will take place only if conflicting changes to the files have not already been set up by you.

- It runs a script that make modes of files installed as part of Solaris packages more secure. These changes are as follows:
  - Removes group and world read permissions for `setuid` and `setgid`.
  - Removes group and world write permissions on all non-`setuid` files that meet any of the following criteria:
    - The file has group and world readable permission, but no world writable permission.
    - The file has world executable permission.
    - The file has identical owner, group, and world permissions.
    - It is a bin-owned directory or non-volatile file and has identical group and world read and executable permissions.
  - Removes write permissions for owners on executables not owned by root.
- It adds `umask 077` to `/.cshrc` and `/.profile`. This makes the default file permission for files created under an interactive root shell readable and writable only by root.
- It adds root to `/etc/ftpusers` to disable root's ability to ftp to the host.
- It sets noshell as the default shell for `sys, uucp, nuucp,` and `listen` accounts to log unauthorized logging attempts. This makes it easier to detect intrusion on the system.

- It sets `MAXWEEKS=12` in `/etc/default/passwd`. If local files are used for password management, this forces all passwords to change periodically.
- It creates `S35umask` to make default file permission for files created by system daemons writable only by the file owner.
- It disables a denial of service attack by adding the line `ndd-set/dev/ipip_respond_to_echo_broadcast 0` in the file `/etc/rc2.d/S69inet`.
- It replaces `/etc/syslog.conf` with a new version for ensuring more granular logging and for detecting intrusion. This new version isolates messages by both facility and logging level and sends the high-level messages to a central logging server.
- It executes `bsmconv` and configures `/etc/security` to log administrative actions, and logins and logouts. This enables C2 auditing, which may catch events missed by `syslog`.
- All changes made by this unit are logged to `/var/sadm/install/contents`. This enables patch installation in the future.

Reconfigurable Settings

The installation of Solaris for ISPs platform extensions and services with their default configuration will override the default service behavior on the hosts where they are installed. This procedure creates a more secure server by disabling Solaris network utilities that are not essential to the Solaris for ISPs software installed on the system.

Note - You *must* review and may modify, if necessary, the default configuration during host configuration.

If you accept the default installation setup, these Solaris services will be disabled, unless noted otherwise. Disabling of these services is not required, but we recommend disabling these services to avoid potential security holes and to conserve resources. To change the value of these services, `inetd.conf` will be modified, unless stated otherwise. To undo these changes, go to “Undoing the Changes” on page 37

Closing Potential Security Holes

We recommend disabling of the following services to ensure protection for passwords and to restrict access to hosts for unauthorized individuals.

Note - If you accept the default setting, you will no longer be able to access the host with these disabled “r” commands.

- `rexecd`: Disable this service to discontinue support for remote command execution via the `rexec(3N)` function, which passes passwords in the clear.
- `rlogind`: Disable this service to ensure security for passwords because it relies on `.rhosts` and `hosts.equiv` for password-less authentication during remote logging.
- `rshd`: Disable this service to protect password because it relies on `.rhosts` and `hosts.equiv` for password-less authentication during remote command execution.

Note - If you accept the default setting, the following services will be enabled. You *must* review and may modify the setting.

- `telnetd`: If you accept the default installation setting, note that this service is enabled as this enables remote login mechanisms.
- `ftpd`: If you accept the default installation setting, note that this service is enabled because it provides support for file transfer to and from remote network sites in the least insecure manner. This service will be disabled if you select Sun Internet FTP Server for installation.

Note - If you require security for telnet and FTP services, set up your network such that file transfer requests are made within the network.

We recommend disabling the following services to protect information from unauthorized users. Disabling these services will enhance system security and will restrict access to system information by preventing host responses to these network requests.

- `fingerd`: Disable this service to safeguard information from a network-based finger request.
- `netstat`: Disable this service to ensure that the contents of the various network-related data structures are not exposed by remote invocation of `netstat`.
- `rstatd`: Disable this service to prevent access to system statistics.
- `rusersd`: Disable this service to protect information about logged-in users.
- `systat`: Disable this service to discontinue support for remotely running `ps` on the host.
- `routing`: Disable this service to ensure that the host is not operated as a router. If disabled, the file `/etc/notrouter` is created.
- `sendmail`: Disable this service to protect against denial of service attacks and to disable support for receiving and sending mail. `S88sendmail` is modified.
- `sprayd`: Disable this service to discontinue support to test the network and record packages sent by spray.

Conserving Resources

We recommend disabling of the following CDE and OpenWindows services unless they are required in your environment. Disabling these services will enhance system performance.

- `cmsd`: Disable this service as it is required only if CDE calendars are located on the host.
- `dtspcd`: Disable this service to discontinue support for CDE sessions.
- `kcms_server`: Disable this service to discontinue support for remote access to OpenWindows KCMS profiles.
- `ttdbserverd`: Disable this service to discontinue support for Tooltalk database server required for proper CDE operation.

We recommend disabling the following network (`inetd`) services unless required in your environment. Disabling these services will free resources and enhance system performance. Modify the default configuration if you require any network utilities listed below.

- `chargen`: Disable this service to discontinue support to test `inetd` and generate characters.
- `discard`: Disable this service to discontinue discarding all input from testing `inetd`.
- `echo`: Disable this service to discontinue support to echo back all input from testing `inetd`.
- `fs.auto`: Disable this service to disable the font server.

Note - If you accept the default setting, the following service will be enabled. You must review and may modify the setting.

- `time`: If you accept the default installation setting, note that this service will be enabled as it keeps someone from querying the system's time remotely. It returns machine-readable time.
- `cachefsd`: If you accept the default installation setting, note that this service will be enabled. This is the `cacheFS` daemon.

We recommend disabling of the following services unless they are essential for your environment. Disabling these services will enhance system performance. Please modify the default configuration if you require any services listed below.

- `automountd`: Disable this service as this supports automounting only and not normal NFS mounts. To change its value, `S74autofs` will be modified.
- `comsat`: Disable this service to discontinue `biff(1)` notification of new mail on the host.
- `daytime`: Disable this service to discontinue support to return the time of day.
- `rquotad`: Disable this service to ensure that the host is not operated as an NFS server supporting disk quotas on its file system.

- `sadmind`: Disable this service to discontinue support for performing distributed system administration operations using Solstice AdminSuite.
- `talkd`: Disable this service to discontinue support for running the interactive talk program.
- `tnamed`: Disable this service to discontinue support for DARPA name server protocol.
- `lpd`: Disable this service to ensure that the host is not operated as a BSD print server. This does not disable the system V print server.
- `uucpd`: Disable this service and discontinue support for copying files named by the source-file arguments to the destination-file argument.
- `walld`: Disable this service and discontinue support for sending messages by wall.
- `Xaserver`: Disable this service and discontinue support for X-based audio.

Note - You can also refer to the on line help during host configuration for help in enabling or disabling the Solaris services.

Undoing the Changes

The changes made during host configuration, to harden and fine tune the system for security and for performance, may not be incorporated in the next release of Solaris for ISPs. This section discusses the steps you can take to undo the changes made to foundation Solaris during host configuration.

Log into the computer where you want to undo the changes and give yourself root access. Determine the changes you want to undo and follow the instructions in the bulleted list:

- The foundation Solaris services are tuned for security and optimum performance from the host configuration graphical user interface (GUI). These changes are reconfigurable and you can reset the values using the host configuration GUI. Refer *Solaris for ISPs Installation Guide* and host configuration on line help to reset Solaris service values.
- The two additional boot files, `S35umask` and `S68echo` in `/etc/rc2.d`, created after installation are automatically removed when the Solaris for ISPs Platform component is uninstalled.

Note - You must manually undo some of the one-time only changes made to the system configuration. To undo the hardening changes:

- Enter: # **`/opt/SUNWfixm/bin/fixmodes -u`** to undo the one-time only changes.
- Enter: # **`bsmunconv`** to switch from C2 auditing mode to C1 mode. This will turn off auditing turned on to catch events missed by `syslog`.

- Compare the current version of `/.cshrc`, `/.profile`, `/.zshenv`, `/etc/ftpusers`, `/etc/default/passwd`, `/etc/syslog.conf` to the *file*`.pre-hcfconfig` version of the file. The current file contains the hardening changes and any other edits that may have been made after hardening was performed. Determine the changes made by the host configuration software and use a text editor to undo the changes.

Note - Do not copy the *file*`.pre-hcfconfig` to the current version of the *file* without determining the changes made by and after the host configuration.

Log File Management

This section describes the resident daemon, `hclfmd`, that performs log file management. This resident daemon runs as root. It starts at boot time and performs the following functions:

- It parses the list of log files in `/etc/syslog.conf` for file paths that do not start with `/dev` (files associated with system devices) and performs a cleanup, journal, and cycle pass every day.

  For every log file written by `syslogd`, it performs the following functions:

  - It renames the existing log file and creates a new daily log.
  - It sends the restart signal (`-HUP`) to the syslog daemon to create a new daily log.
  - It generates a weekly archive by compressing daily log files every week and stores it as `name.`*YYYYMMDD*`-`*YYYYMMDD*`.tar.z`.
  - It discards weekly archives that are more than a month old.

- It obtains the location of audit logs from `/etc/security/audit_control` and performs a cleanup, journal, and cycle pass every day.

  It performs the following functions for every locally mounted audit directory:

  - It executes `audit -n` to create a new daily log. This signals the audit directory to close the current audit file and open a new audit file in the current audit directory.
  - It generates a weekly archive by compressing daily log files every week and stores it as `audit.`*YYYYMMDD*`-`*YYYYMMDD*`.tar.z`.
  - It discards weekly archives that are more than a month old.

- It performs an intrusion detection check every 10 minutes.

  - It detects and reports every failed authorization entry in syslog files.
  - By default, `/etc/opt/SUNWisp/hc/hclfmd.conf` is configured to send mail to root for every failed authorization attempt entered in syslog.

Note - You may re-configure this file. By default, it is configured as follows: `/var/log/badauth:/usr/bin/mailx -s ''%f'' root < %c` where:

- `/var/log/badauth` is the file where the entries are made.
- `/usr/bin/mailx -s` is the command to send mail to root.
- "*%f*" is the subject-line of the mail, containing the name of the file where the entries were detected, and
- "*%c*" is the new content of the `syslog` file.

Creating User-Defined Scripts

This section discusses certain installation and/or configuration updates you may provide to be executed after installation of Solaris for ISPs. These parameters can be written as a shell script. For example, you can write a command similar to: `echo "`*foo*`" >> /etc/ftpusers`

The path to this script that you create can be registered while configuring the host (`Post-Configuration Command` screen) for installation of Solaris for ISPs. Alternately, you may string these commands. This post-configuration command provided by you will be executed by Solaris for ISPs post-installation script during batch installation.

Note - Creating this script is optional.

Some post-installation system setup examples that you may address in your script to be executed after installation are illustrated in the following. For example:

- Write a program to verify and confirm changes to system setup.
- Write a program to notify or print disk space availability after installation.
- Write a program to re-configure notification messages from syslog for failed authorization entries. Refer to "Log File Management" on page 38.
- Write a program to configure other independent software.
- Write a program to configure DNS host:
  - Enter some DNS server IP addresses in `/etc/hosts`.
  - Set up `/etc/resolv.conf`.
  - Modify `/etc/nsswitch.conf`.

Note - The DNS server configurations discussed in this section are suggestions only. You need not reconfigure your existing DNS server. Please refer to a relevant document to reconfigure your DNS server to support Solaris for ISPs extensions and services request for regular name lookups.

Restoring Default AWS Settings

Sun Internet Administrator uses a Web server for the administration funtions from its user interface. This Web server is referred to as the Administration Web Server (AWS). You can, if necessary, reconfigure the Administration Web Server to suit your requirements. Refer to on line help to reconfigure the Admin Web Server. To ensure that you do not lose the default configuration, this section discusses the location of the default Administration Web server configuration files and the method to restore the default settings.

The Administration Web Server default configuration files are located in `/etc/opt/SUNWixamc/awsconf/default/*` and `/var/opt/SUNWixamc/awsconf`. When reconfiguring the Admin Web Server, only the `/var/opt/SUNWixamc/awsconf` file configurations must be changed.

To restore the default settings, copy `/etc/opt/SUNWixamc/awsconf/default/*` to `/var/opt/SUNWixamc/awsconf`.

Note - For the effective functioning of Sun Internet Administrator console, *do not* change the default settings in `aws.conf`, `site.conf`, `map.conf`, `realms.conf`, and `access.acl`.

CHAPTER 4

Installation Requirements

This chapter discusses the preinstallation requirements for Solaris for ISPs. Please read carefully and ensure that your network hosts meet all prerequisites specified.

Installation Prerequisites

This section discusses Solaris for ISPs platform extensions and services preinstallation requirements.

- Operating system requirements
  - Version of Solaris
  - Operating system patches
- Hardware requirements
  - CPU
  - Disk space
  - RAM
  - Swap space
- Web browser supported
- Component dependencies
  - Dependencies on packages
  - Default Component Ports

Operating System Requirements

This section discusses the operating system requirements such as the version of Solaris required for the installation and functioning of Solaris for ISPs platform extensions and services, and the operating system patches that you must download.

Version of Solaris

The Solaris version required for the successful installation and functioning of Solaris for ISPs platform extensions and services is *Solaris 2.6 and greater*.

Patches to Download

This section discusses the operating system patches for SPARC and x86. Refer to the relevant patch number for your system and download. Please address this section while preparing the host server(s) for the installation of Solaris for ISPs.

Upgrade the operating system by downloading the following patches from `http://www.access1.sun.com/Products/ISP/`.

Note - *These Solaris 2.6 patches for the operating system are required for the effective functioning of Solaris for ISPs platform extensions and services.*

TABLE 4–1 Required Operating System Patches

Patch description	SPARC Patch #	x86 Patch #
SunOS 5.6: libc & watchmalloc patch	105210-03	105211-03
SunOS 5.6: /kernel/fs/hsfs patch	105486-02	105487-02
SunOS 5.6: /kernel/drv/glm patch	105580-03	NA
SunOS 5.6: /usr/lib/nfs/mountd patch	105615-03	105616-03
SunOS 5.6: adb & kadb patch	106031-02	NA
SunOS 5.6: kernel update patch	105181-06	105182-06
SunOS 5.6: /usr/sbin/rpcbind patch	105216-02	105217-02
SunOS 5.6: /kernel/drv/ssd patch	105356-01	NA
SunOS 5.6: /kernel/drv/ses patch	105357-01	NA
SunOS 5.6: sf & socal driver patch	105375-04	NA
SunOS 5.6: /kernel/misc/nfssrv patch	105379-01	105380-01
SunOS 5.6: /usr/bin/at patch	105393-01	105394-01
SunOS 5.6: /usr/bin/volrmmount patch	105407-01	105408-01

TABLE 4–1 Required Operating System Patches *(continued)*

Patch description	SPARC Patch #	x86 Patch #
SunOS 5.6: /usr/bin/vacation patch	105518-01	105519-01
SunOS 5.6: /kernel/fs/fifofs patch	NA	105581
SunOS 5.6: /usr/lib/nfs/mountd patch	105615-03	105616-03
SunOS 5.6: libbsm patch	105621-01	105662-01
SunOS 5.6: /usr/bin/login patch	105665-01	105666-01
SunOS 5.6: /usr/bin/rdist patch	105667-01	105668-01
SunOS 5.6: /kernel/drv/ip patch	105786-01	105787-01

Note - *These Solaris 2.6 patches for the CDE 1.2 and man pages are required for the effective functioning of Solaris for ISPs platform extensions and services.*

TABLE 4–2 Recommended Operating System Patches

Patch description	SPARC Patch #	x86 Patch #
CDE 1.2: dtlogin patch	105703-03	NA
CDE 1.2: dtpad patch	105558-01	NA
CDE 1.2: libDtSvc patch	105669-02	NA
SCDE 1.2: dtappgather patch	105837-01	NA
Motif 1.2.7 Runtime library patch [1]	105284-08	105285-08
Manual page patch for the LDAP client library	106497-01	106498-01
Manual page patch [2]	105390-02	106061-02
SGML patch [2]	106123-02	106124-01

1. This patch enhances the Motif window manager to provide compatibility with JDK 1.1.5.
2. You will require these patches to view Solaris for ISPs man pages.

Hardware Requirements

This section discusses the hardware such as CPU, disk space, RAM, and swap space required for the installation of Solaris for ISPs.

Solaris for ISPs Platform Extensions

This section discusses the hardware requirements for the installation of Solaris for ISPs platform extensions.

TABLE 4–3 Hardware Requirements for Platform Extensions

Solaris for ISPs Platform Extensions	CPU	Disk Space	RAM	Swap Space
	Recommended	Minimum	Recommended	Recommended
Solaris for ISPs [1]	High-end dual processor system	4 GB	256 MB	1 GB
Solaris for ISPs Platform [2]	High-end workstation	3 MB	NA	NA
Sun Internet Administrator		1 MB	64 MB	64 MB
Sun Directory Services		50 MB	64 MB	
FlexLM License Server		3 MB	NA	NA
Sun Internet Services Monitor		5 MB	64 MB	1 MB
Sunscreen™ SKIP1.1.1[3]		10 MB	32 MB	
JDK 1.1.5		3 MB		
HotJava 1.1.4		9 MB	32 MB	

1. The requirements specified here are for an installation of all the Solaris for ISPs components on a single machine.
2. Since this component *must* be installed on every machine running a Solaris for ISPs component, ensure that these requirements are met by every Solaris for ISPs machine on the network.
3. Go to `http://www.sun.com/security/skip/ for more information`.

Solaris for ISPs Services

This section discusses the hardware requirements for the installation of Solaris for ISPs services.

TABLE 4–4 Hardware Requirements for Services

Solaris for ISPs Services	CPU	Disk Space	RAM
	Recommended	Minimum	Minimum
Sun Internet FTP Server	High-end workstation	23.1 MB	64 MB
Sun Internet News Server		2-20 GB [1]	64 MB
SWS		4MB + documents and log files	64 MB

1. This specification expands as news data base expands.

Web Browser Supported

The web browser supported by Solaris for ISPs platform extensions and services is Sun Microsystems HotJava 1.1.4.

Component Dependencies

This section discusses the component dependencies on packages and the default component ports.

Dependencies on Packages

This section discusses dependencies of Solaris for ISPs platform extensions and services on required and recommended bundled and independent packages.

TABLE 4–5 Solaris for ISPs Component Dependencies

Required and recommended Packages	For Solaris for ISPs extensions and services	Where to find
JDK/JRE 1.1.5	All Solaris for ISPs components *require* this package.	Bundled
SSL	SWS, Sun Internet Administrator, Sun Internet FTP Server, Sun Internet News Server	Bundled
SKIP	Sun Internet Administrator[1]	Bundled

TABLE 4–5 Solaris for ISPs Component Dependencies *(continued)*

Required and recommended Packages	For Solaris for ISPs extensions and services	Where to find
JavaIDL	Sun Internet FTP Server, Sun Internet News Server	Bundled
PGP 5.0	Sun Internet News Server[2]	Go to `http://www.nai.com/default_pgp.asp`[3]

1. SKIP is recommended on the server side for this component.
2. This package is *recommended* for control message authentication.
3. For information on PGP public keys specific to news groups, go to `http://www.isc.org/inn.html`

Default Component Ports

Solaris for ISPs uses the following ports:

- 8000 is the default port for the temporary Web server used for the installation by the host configuration software.
- 50080 is the default port for the administration Web server used by Sun™ Internet Administrator™.
- 1760 is the default port for Sun™ Directory Services.
- 2380 is the default port for Sun™ WebServer™ (SWS).
- 2381 is the default port for Sun™ Internet Services Monitor™.

CHAPTER 5

Using Directory Services

Solaris™ for ISPs™ uses Sun™ Directory Services to store component software information and login information for Sun™ Internet Administrator™ and for some services. This information includes standard object classes and newly-defined objects that support Solaris for ISPs functionality.

This chapter provides information on the Solaris for ISPs information tree structure, on making entries in the tree from the command line, and on required access controls. For information on the schema extension itself, see Chapter 6.

Sun Directory Services provides the following tools for using and administering the directory:

- The Deja tool, a Java-based directory editor, provides add, modify, and delete capability.
- The Sun Directory Services administration console offers local and remote administration of the server.
- The Web gateway allows browse access from any browser.
- A complete set of command-line programs.

The Sun Directory Services books, *Sun Directory Services 3.1 Administration Guide* and *Sun Directory Services 3.1 User's Guide*, include complete information on starting and using these tools. They are provided in AnswerBook2 format on the product CD-ROM. The command-line programs are documented in man pages in section 1 (`/opt/SUNWconn/man`).

Solaris for ISPs Directory Structure

Solaris for ISPs requires a specific structure in the directory information tree (DIT), which is created during installation and configuration. The required structure consists of two naming contexts, referred to as the Open Systems Interconnection (OSI) tree and the Domain Component (DC) tree. Portions of the two trees are parallel. This parallel structure facilitates mapping of domain names from a DNS request through the DC tree to the actual content entries in the OSI tree.

OSI Tree Structure

The OSI tree contains the actual entries for Solaris for ISPs, its component services, administrators of those services, and subscribers to the services. The required structure is shown in Figure 5–1.

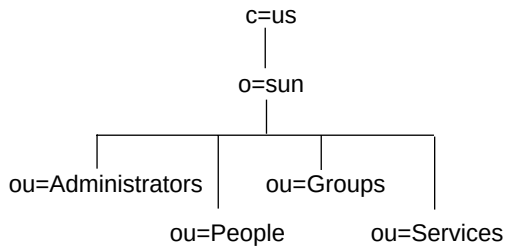


Figure 5–1 Solaris for ISPs OSI Tree

In the OSI tree, the entry for the domain sun.com is represented by the entry with the distinguished name o=sun,c=us. This entry is called the *root domain*, and represents the Solaris for ISPs customer's business. You specify the name of the root domain during installation of the directory services.

Beneath the root domain are four required `organizationalUnit` entries:

- *Administrators* contains entries for Sun™ Internet Administrator™ administrators. These entries are created by the product when you create administrators using the GUI.
- *People* contains entries for subscribers to ISP services. You create entries for your customers, whether at the command line or by using the Deja tool.
- *Groups* contains entries that group subscriber entries together for access control purposes. You create group entries according to your needs, whether at the command line or by using the Deja tool.
- *Services* contains entries created for Solaris for ISPs services. You should make entries under this unit only when you are integrating a new service or are

performing a special virtual hosting configuration of Sun™ Internet FTP Server™. See the Sun Internet FTP Server and the Sun Internet Administrator on-line help for information on this requirement.

People, *Groups*, and *Services* nodes are required under each domain entry you define. The *Administrators* node exists only under the root domain.

Figure 5–2 illustrates a typical set of entries under each organizational unit.

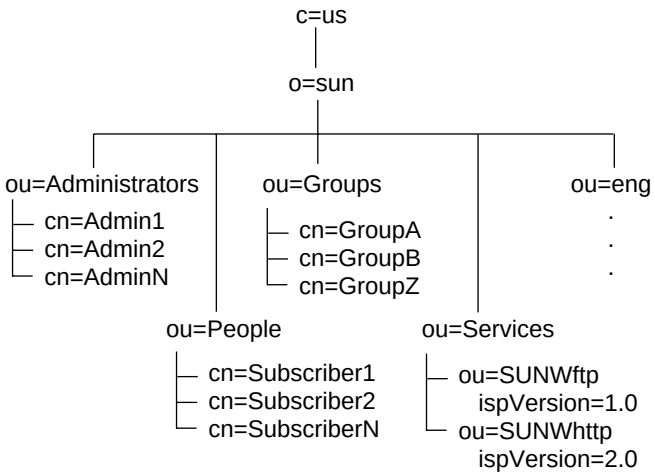
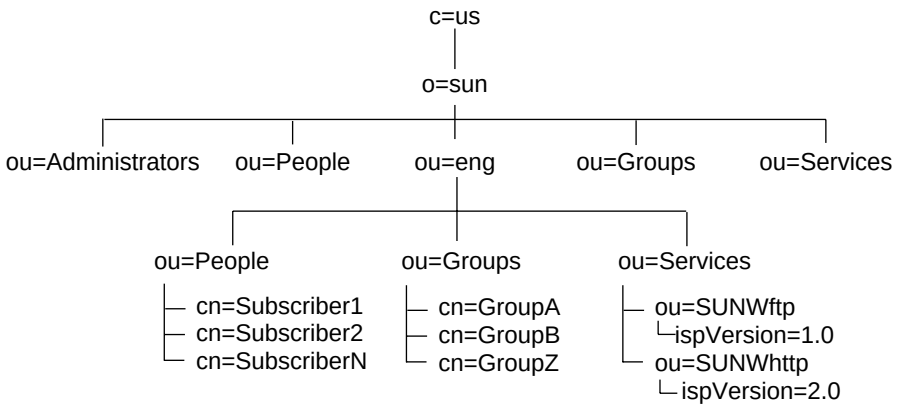


Figure 5–2 OSI Tree Entries

The `organizationalUnit` entry *eng* is an example of a *domain* entry. This might be a corporate customer of the ISP, or anyone who has virtual domain hosting services with the ISP. Domains must have two entries: one here in the OSI tree and another in the DC tree for domain name mapping. See “Creating Domain Entries” on page 53 for information on creating these two entries properly.

Domains, like the root domain, require certain `organizationalUnit` entries within them. As shown in Figure 5–3, *People*, *Groups*, and *Services* entries are also required in a domain below the root.



Note: Data entries in the root domain are omitted for clarity.

Figure 5–3 Domain Structure in the OSI Tree

When creating a domain entry in the OSI tree, you must also create the entries for *People*, *Groups*, and *Services*. When you configure services for this domain, service entries are made under the *Services* organizational unit. Subscriber information for this domain forms `ispSubscriber` entries under the *People* organization unit.

Note - Administrator entries are made only under the root domain in this version of Solaris for ISPs. These entries are created by Sun Internet Administrator when you specify them through the GUI.

DC Tree Structure

The DC (domain component) tree maps domain name format (for example, sun.com) to the distinguished name of the corresponding entry in the OSI tree. As shown in Figure 5–4, the DC is usually relatively flat and simpler than the OSI tree.

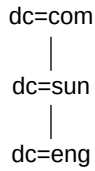


Figure 5–4 Solaris for ISPs DC Tree

In Figure 5–4, the entry dc=sun,dc=com corresponds to the o=sun,c=us entry in the OSI tree. The `eng` domain here maps to the domain name server (DNS) form eng.sun.com.

For details on how to make the two domain entries, see “Creating Domain Entries” on page 53.

Making Directory Entries from the Command Line

General information on how to create directory services entries is located in Chapter 5, “Loading and Maintaining Directory Information,” in the *Sun Directory Services 3.1 Administration Guide*. This section presents instructions on how to create the specific entries required by Solaris for ISPs.

For information on how to create directory entries using the Deja tool, see the Sun™ Internet Administrator™ on-line help.

Creating Directory Entries: General Procedure

Sun Directory Services has the following command-line utilities for creating and modifying directory entries:

- `ldapadd`
- `ldapmodify`
- `ldapdelete`

These directory services command-line utilities require root access. They are fully documented in reference manual pages (section 1).

Both `ldapadd` and `ldapmodify` can take input from the command line or from a specified file. Because information for an entry can be rather lengthy and complex, the sections that follow describe the form requiring a text file.

In each case, creating an entry (or entries) requires the following steps:

1. **Write a file specifying the entry or entries to be made in the directory. The format for this file is specified in the `ldif(4)` reference manual page.**

2. **Obtain root access and create the entry using `ldapadd`, specifying the file with the entry information.**

In every case, the form of the `ldapadd` command should be:

`# ldapadd –D "`*`BindDN`*`" –w `*`password`*` –f `*`file`

Where the *BindDN* is the distinguished name (DN) for binding to the directory with write access to this part of the directory tree, and *password* the password for binding. Replace the *file* option with the name of the `ldif` file you have created.

Structure of an `ldif(4)` File

For each entry you add at the command line, you will create an `ldif`-format file to hold the information about the entry. These are simple text files with one or more directory entries each separated by a single blank line. Each entry has the structure: of the following example.

Note - Only mandatory attributes are shown in the example. Most object classes have a number of optional attributes that may be set appropriately for your particular use of the entry.

```
dn: ou=wcgate1,ou=eng,o=sun,c=US
ou: wcgate1
associateddomain: wcgate1.eng.sun.com
objectclass: organizationalUnit
objectclass: domainRelatedObject
```

Where

dn — Indicates the distinguished name of the entry being created.

ou — Is the naming attribute of the entry being created. Common naming attributes include `commonName`, `organizationalUnit` (ou), and `domainComponent` (dc).

associatedDomain — Contains the domain name (in dot notation) of the corresponding entry in the DC tree. See “Solaris for ISPs Directory Structure” on page 48 for information on how the OSI tree and the DC tree interact. See “Creating Domain Entries” on page 53 for instructions on creating the two cross-referenced entries for a domain.

There may be many attribute:value pairs in this position, one per line.

objectClass — Is the object class (type) of the entry. There may be many `objectClass` entries; this example shows two.

For more detailed information on available object classes and attributes, see Chapter 6 of this guide, and Chapter 8, “Configuring the Directory Schema,” of the *Sun Directory Services 3.1 Administration Guide.*

Creating Domain Entries

To create a domain in the directory, you must create two parallel domain entries, one in the OSI tree and one in the DC tree, and then create the required `organizationalUnit` entries under the domain entry in the OSI tree.

To create the domain *wcgate1* under eng.sun.com, perform the following steps:

1. **Edit a text file (for example,** `domain.ldif`**) and enter the data for the OSI tree entry:**

```
dn: ou=wcgate1,ou=eng,o=sun,c=US
ou: wcgate1
associateddomain: wcgate1.eng.sun.com
objectclass: organizationalUnit
objectclass: domainRelatedObject
```

   Note that the `associatedDomain` attribute of the entry contains the name of the DC tree entry in dot notation (DNS style).

2. **Add to** `domain.ldif` **the data for the DC tree entry:**

```
dn: dc=wcgate1,dc=eng,dc=sun,dc=com
dc: wcgate1
associatedname: ou=wcgate1,ou=eng,o=sun,c=US
description: DNS-to-DN Mapping for wcgate1.eng.sun.com
labeleduri: ldap:///ou=wcgate1,ou=eng,o=sun,c=US??sub
objectclass: domain
objectclass: labeledURIObject
```

   Note that the `associatedName` attribute of the entry contains the distinguished name of the OSI tree entry. The `labeledURI` attribute contains the same information (as specified in RFC 2255).

3. **Add to** `domain.ldif` **the data for the required** `Services` **organizational unit entry:**

```
dn: ou=Services,ou=wcgate1,ou=eng,o=sun,c=US
ou: Services
objectclass: organizationalUnit
```

4. **Add to** `domain.ldif` **the data for the required** `People` **organizational unit entry:**

```
dn: ou=People,ou=wcgate1,ou=eng,o=sun,c=US
ou: People
objectclass: organizationalUnit
```

5. **Add to** `domain.ldif` **the data for the required** `Groups` **organizational unit entry:**

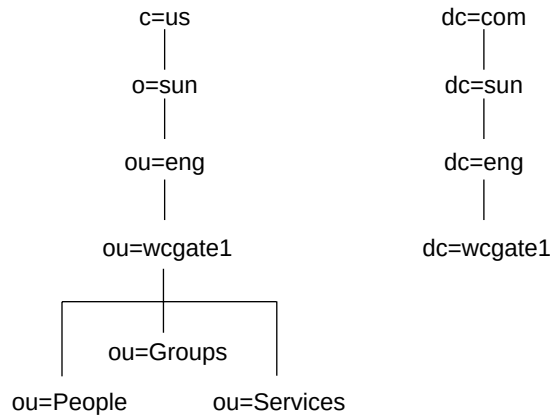
```
dn: ou=Groups,ou=wcgate1,ou=eng,o=sun,c=US
ou: Groups
objectclass: organizationalUnit
```

6. **Save and close** `domain.ldif`**.**

7. **Obtain root access and add the entries to the directory with the following command, replacing the bind DN and password with your own:**

   `# ldapadd –D "`*`cn=admin,o=sun,c=US`*`" –w` ***password*** **`–f`** ***domain.ldif***

When your `ldapadd` is complete, the directory looks like Figure 5–5.



Note: some entries omitted for clarity.

Figure 5–5 Directory Structure with a Domain Added

Creating Subscriber Entries

Solaris for ISPs subscribers come in several varieties:

- The general (basic) subscriber
- The subscriber who uses virtually-hosted FTP or Web services
- The subscriber who gains access to services through a RADIUS server

- The subscriber who uses both, and whose directory entry requires both RADIUS and FTP information

In the sections that follow, instructions are provided for building the complex subscriber entry by creating the simpler entry and adding to it.

Creating a Basic Subscriber Entry

Before you can create subscriber entries, the domain and the *People* organizational unit entries must exist. Once you have created those entries, you can edit a text file (for example, `people.ldif`) and enter the data for the subscriber. The basic subscriber entry has the single object class `ispSubscriber`, and very few mandatory attributes. The file for a basic subscriber looks like this:

```
dn: cn=Jane Doe (jldoe),ou=People,ou=wcgate1,ou=eng,o=sun,c=US
commonname: Jane Doe (jldoe)
sn: Doe
uid: jldoe
userpassword: hidden
objectclass: ispSubscriber
```

Where

dn — Is the distinguished name of the subscriber entry.

commonName — Is the naming attribute of a subscriber entry (`ispSubscriber` object class). For Solaris for ISPs subscribers and administrators, the value of the `commonName` attribute takes the form **`Firstname Lastname (userid).`**

sn — Is the surname of the subscriber.

uid — Is the login name of the subscriber.

userPassword — Is the password, limited to eight characters if you are sharing password information with UNIX accounts. This value is generated with the encryption method you set in the directory services administration console.

objectClass: ispSubscriber — Is the object class type of this subscriber entry.

You can create any number of subscriber entries by adding blocks of data with different attribute values to the file. When it is complete, save and close `people.ldif`. Obtain root access and add the subscriber entries to the directory with the following command, replacing the bind DN and password with your own:

```
# ldapadd –D "cn=admin,o=sun,c=US" –w password –f people.ldif
```

Adding FTP and Web Virtual Hosting Information

The information required for the specially-configured virtual hosting available with Sun™ Internet FTP Server™ and Sun™ WebServer™ (SWS) adds only three attributes to the data file:

```
gidnumber: 60001
uidnumber: 60001
ispcontentdirectory: jldoe
```

Where

gidNumber	Is the UNIX group ID specified for this user in the virtually-hosted domain for FTP and Web services.
uidNumber	Is the UNIX user ID specified for this user in the virtually-hosted domain for FTP and Web services.
ispContentDirectory	Is the location (relative to the associated domain's document root) where this subscriber's content files are located.

Note - Setting the values for the `uidNumber` and `gidNumber` attributes requires existing UNIX accounts properly set up to share access to the virtual FTP domain. See the Sun Internet FTP Server on-line help for information on defining a virtual host configuration.

You can create any number of subscriber entries by adding blocks of data to the file. When it is complete, save and close `people.ldif`. Obtain root access and add the subscriber entries to the directory with the following command, replacing the bind DN and password with your own:

```
# ldapadd –D "cn=admin,o=sun,c=US" –w password –f people.ldif
```

If you have already created these entries, you must perform an `ldapmodify`. Locate the manual page for `ldapmodify(1)` and follow those instructions.

Adding Remote User Information

An entry for a subscriber who gains access to ISP services through a RADIUS server must support an additional object class (`remoteUser`) and has several attributes added to the entry information.

Note - The default Solaris for ISPs configuration designates the root domain as the search base for RADIUS subscriber entries. If your configuration is different, use the directory services administration console to configure RADIUS and enter values appropriate for your search base.

The additional lines in the `ldif` file are:

```
objectclass: remoteUser
authsuffixname: @ispxpress
grpcheckinfo: authSuffixName
grpcheckinfo: userPassword
authserviceprotocol: Framed-User
framedrouting: None
framedprotocol: PPP
grpreplyinfo: authServiceProtocol
grpreplyinfo: framedProtocol
grpreplyinfo: framedRouting
```

Where

objectClass: remoteUser	Is a required object class for the subscriber accessing services using a RADIUS server.
authsuffixname: @ispxpress	Is a suffix added to the subscriber's user name to enable the RADIUS server to distinguish among entries with the same `uid` in different domains. Enter the appropriate suffix for the specific user entry.
grpcheckinfo: authSuffixName	Indicates that the RADIUS server should verify the `authSuffixName` attribute value before selecting the entry to authenticate against.
grpcheckinfo: userPassword	Indicates that the RADIUS server should verify the `userPassword` attribute value before selecting the entry to authenticate against.

authserviceprotocol: Framed-User	If you are using the default RADIUS configuration, enter this attribute exactly as shown. The correct value is determined by the configuration of your network access server.
framedrouting: None	If you are using the default RADIUS configuration, enter this attribute exactly as shown. The correct value is determined by the configuration of your network access server.
framedprotocol: PPP	If you are using the default RADIUS configuration, enter this attribute exactly as shown. The correct value is determined by the configuration of your network access server.
grpreplyinfo: authServiceProtocol	Tells the RADIUS server to include the value of the `authServiceProtocol` attribute in its reply message.
grpreplyinfo: framedProtocol	Tells the RADIUS server to include the value of the `framedProtocol` attribute in its reply message.
grpreplyinfo: framedRouting	Tells the RADIUS server to include the value of the `framedRouting` attribute in its reply message.

You can create any number of subscriber entries by adding blocks of data to the file. When it is complete, save and close `people.ldif`. Obtain root access and add the subscriber entries to the directory with the following command, replacing the bind DN and password with your own:

`# ldapadd –D` ***"cn=admin,o=sun,c=US"*** **`–w`** ***password*** **`–f`** ***people.ldif***

If you have already created these entries, you must perform an `ldapmodify`. Locate the manual page for `ldapmodify(1)` and follow those instructions.

The Complete `ldif` File

The complete `ldif` file for a complex user looks like:

```
dn: cn=Jane Doe (jldoe),ou=People,ou=wcgate1,ou=eng,o=sun,c=US
commonname: Jane Doe (jldoe)
```

(continued)

```
sn: Doe
uid: jldoe
userpassword: hidden
gidnumber: 60001
uidnumber: 60001
objectclass: ispSubscriber
objectclass: remoteUser
ispcontentdirectory: /home/users/jldoe
authsuffixname: @ispxpress
grpcheckinfo: authSuffixName
grpcheckinfo: userPassword
authserviceprotocol: Framed-User
framedrouting: None
framedprotocol: PPP
grpreplyinfo: authServiceProtocol
grpreplyinfo: framedProtocol
grpreplyinfo: framedRouting
```

Creating Group Entries

Before you can create group entries, a number of entries must already exist:

- The two domain entries (OSI and DC trees)
- The *Group* organizational unit entry
- The subscriber entries (under the *People* node) that will become the members of the group.

Once you have created those entries, you can start a text file (for example, `groups.ldif`) and enter the data for the group. A typical data set looks like the following:

```
dn: cn=isp-gp1,ou=Groups,ou=wcgate1,ou=eng,o=sun,c=US
cn: isp-grp1
objectclass: groupOfNames
member: cn=Ed Anchor (anchor),ou=People,ou=wcgate1,ou=eng,o=sun,c=US
member: cn=April Shower (showers),ou=People,ou=wcgate1,ou=eng,o=sun,c=US
member: cn=Chili Jones (relleno),ou=People,ou=wcgate1,ou=eng,o=sun,c=US
```

Where

dn	Is the distinguished name of the group to be created.
cn	Is the relative distinguished name of the group entry.
objectClass	The object class `groupOfNames` distinguishes this type of entry.

member Each `member` attribute takes as its value the distinguished name of an existing subscriber entry.

You can create any number of group entries by adding data to the file. When it is complete, save and close `groups.ldif`. Obtain root access and add the groups to the directory with the following command, replacing the bind DN and password with your own:

```
# ldapadd –D "cn=admin,o=sun,c=US" –w password –f groups.ldif
```

Solaris for ISPs Access Control

Solaris for ISPs sets access control for the directory services, to ensure proper access by parts of the software that require it while assuring security by preventing access by others. The general principal of these access controls is that all entities have read access while write access is restricted. It is very important that you do not change existing access controls, or you may introduce security risks or cause Solaris for ISPs to fail.

Remember that the access control rules are order sensitive. When Sun Directory Services checks for access, the first rule that applies to the request is used. Any remaining rules are ignored. Therefore, do not change the order of the rules in the file. When creating a new rule, be careful that it does not accidentally apply to existing Solaris for ISPs information and invalidate some access control rule already in place.

Note - Access control checking is switched off if you bind to the directory as its administrator.

Generally, the information special to Solaris for ISPs is stored in entries supporting object classes defined in the Solaris for ISPs schema extension. Each of these classes is named beginning with the string “isp.” Any rule in the access configuration file that contains such an object class (or attribute) is likely a Solaris for ISPs rule and, as such, sensitive to any changes. The access control rules are defined in `/etc/opt/SUNWconn/ldap/current/dsserv.acl.conf`.

For complete information on Sun Directory Services access controls, see Chapter 1, “Introduction to Directory Concepts,” and Chapter 4, “Configuring a Directory Server,” of the *Sun Directory Services 3.1 Administration Guide.*

The sections that follow describe the general behavior ensured by the Solaris for ISPs access controls. The phrase “has access” indicates that binding to the directory with that entry’s DN and password will give the indicated form of access.

Rules Enabling Sun Internet Administrator Functionality

Sun Internet Administrator needs the following kinds of access to do its work:

- It needs to create and delete administrators. Therefore, Sun Internet Administrator has write access to the portion of the DIT defined by the `Administrators` organization unit entry.
- It must be able to change certain administrator attributes (notably the `userPassword` and `ispAuthorizedServices` attributes).
- It must be able to control the creation of managed service entries (`ispManagedService`). Therefore, Sun Internet Administrator owns its own portion of the tree, below the top-level *SUNWixamc* entry (for example, ispVersion=1.0,ou-SUNWixamc,ou-Services,o=sun,c=us).
- It needs to create the top-level service entries for services it registers and manages. Therefore, Sun Internet Administrator has the access and information it needs to write to that portion of the DIT (for example, ou-Services,o=sun,c=us).
- It also needs to set the value of the protected `ispPrivateData` attribute on `ispService` entries. Therefore, Sun Internet Administrator has read / write access to that attribute of existing service entries. (In fact, no other entity has any access to the `ispPrivateData` attribute.)

Rules Enabling Service Functionality

The various Solaris for ISPs services need to record and access configuration information stored under their service entries (those located under the *Services* node in subdomains and virtual domains). Therefore, each has the access and information it need to write to create and modify entries in that portion of the DIT, including its own service entry.

Rules Enabling Proper User Access

Users (subscribers and administrators) have write access to their password attribute, but cannot change other parts of their entry. However, any administrator with management access to Sun Internet Administrator has global access and can change anything.

CHAPTER 6

Solaris for ISPs Directory Services Schema

Solaris™ for ISPs™ extends the standard schema defined in Sun™ Directory Services. The base schema is defined in Chapter 8, "Configuring the Directory Schema," of the *Sun Directory Services 3.1 Administration Guide*. Extensions to the base schema, both object classes and attributes, are discussed in this chapter.

Note - The schema defined by Solaris for ISPs is unstable in this first release of the product. Object class and attribute definitions that are a part of the schema extension may change without warning in future releases.

Maintaining the Schema

In general, you should not change existing object classes or attributes in the schema, but use or add to them for your purposes. If you change an attribute or object class so that the Solaris for ISPs software cannot use it, you may have to reinstall the directory. In this case, any data entries not backed up will be lost. For information on backing up the directory, see the Sun Directory Services manual pages for `ldbmcat` and `ldif2ldbm`.

If you decide to add to the schema, refer to Chapter 8, "Configuring the Directory Schema," of the *Sun Directory Services 3.1 Administration Guide* for complete details.

What to Back Up

To ensure the integrity of the directory, you should periodically back up the schema configuration files. Certainly, you should back up the schema before starting work to extend it for your own uses.

As discussed in detail in Chapter 4, "Configuring Directory Services," of the *Sun Directory Services 3.1 Administration Guide*, the following files are critical for directory services function:

- `dsserv.conf` holds the main configuration information.
- `dsserv.oc.conf` holds the object class definitions.
- `dsserv.at.conf` holds the attribute definitions.
- `dsserv.acl.conf` holds access control information.

Copies of these files are stored in three places in the system:

- `/etc/opt/SUNWconn/ldap/current` holds the current configuration files.
- `/etc/opt/SUNWconn/ldap/default` holds the default configuration files that were installed with the software. (These files are read-only.)
- `/etc/opt/SUNWconn/ldap/previous` holds the previous configuration files.

Before starting any work on the schema, back up the configuration files in `/etc/opt/SUNWconn/ldap/current` and those in `/etc/opt/SUNWconn/ldap/previous`. When you edit the files, Sun Directory Services copies the unedited files in `~/current` to `~/previous`. It does this only once per editing session, (until you restart `dsservd`). If you are making many changes to the schema, you may want to make manual backups of your changes as you work.

Backup information is presented in detail in Chapter 4, "Configuring Directory Services," of the *Sun Directory Services 3.1 Administration Guide.*

Restoring the Schema

To restore your directory services configuration to a previous version, stop `dsservd`, replace the desired configuration files in `/etc/opt/SUNWconn/ldap/current`, and restart the daemon. Step-by-step information is presented in Chapter 4, "Configuring Directory Services," of the *Sun Directory Services 3.1 Administration Guide.*

▼ Restoring the Solaris for ISPs Schema

When Solaris for ISPs is installed, the original schema configuration files are backed up at `/etc/opt/SUNWisp/SUNWconn/ldap/backup`. This files were customized at installation with your root domain and administrator information. If you are

working on the schema, and arrive at a situation where Solaris for ISPs does not work, restore the default schema as follows:

1. **Log into the machine where the directory services is running.**

2. **Give yourself root access.**

3. **Stop the directory services server, by entering:**
   ```
   # /etc/init.d/dsserv stop
   ```

   Note - You can also stop and start the server through its administration console.

4. **Copy each of the configuration files from the backup to the current directory:**
   a. `cp /etc/opt/SUNWisp/SUNWconn/ldap/backup/ispdsserv.conf /etc/opt/SUNWconn/ldap/current/dsserv.conf`

   b. `cp /etc/opt/SUNWisp/SUNWconn/ldap/backup/ispdsserv.at.conf /etc/opt/SUNWconn/ldap/current/dsserv.at.conf`

   c. `cp /etc/opt/SUNWisp/SUNWconn/ldap/backup/ispdsserv.oc.conf /etc/opt/SUNWconn/ldap/current/dsserv.oc.conf`

   d. `cp /etc/opt/SUNWisp/SUNWconn/ldap/backup/ispdsserv.acl.conf /etc/opt/SUNWconn/ldap/current/dsserv.acl.conf`

   e. `cp /etc/opt/SUNWisp/SUNWconn/ldap/backup/mapping/radius.mapping /etc/opt/SUNWconn/ldap/current/mapping/radius.mapping`

5. **Start the directory services server by entering:**
   ```
   # /etc/init.d/dsserv start
   ```

Solaris for ISPs Object Classes

This section contains a list of the object classes added to the base schema to support Solaris for ISPs. Attributes listed as mandatory must have a value entered when the entry is created. Object classes are listed in alphabetical order.

ispAdministrator Class

Purpose: Defines an entry representing an administrator of ISP services and networks. The ispAdministrator's relative distinguished name the commonName attribute and its value. Its superior object is ispSubscriber.

TABLE 6–1 ispAdministrator Attributes

Attribute name	Mandatory?	Schema	Purpose
associatedDomain	No	Base	The domain with which this administrator is associated. Reserved for future Solaris for ISPs functionality.
commonName	Yes	Base	The name of the administrator described by the entry, in the form Firstname Lastname (userid).
description	No	Base	An arbitrary description of the administrator.
gidNumber	No	Base	A UNIX group-ID. For virtually-hosted FTP or Web services, the group ID specified for the virtual domain.
homeDirectory	No	Base	The file system location of the home directory of the administrator described by the entry. (Not used by Solaris for ISPs.)
ispAdministeredService	No	Extension	The distinguished names of services this administrator is authorized to manage.
ispContentDirectory	No	Extension	A directory location where content belonging to the administrator is located. For virtually-hosted FTP or Web services, the path to user content relative to the ispRootDirectory.
labeledURI	No	Base	The Uniform Resource Identifier and label associated with the Web page of this administrator.
mail	No	Base	The advertised electronic mail address of the administrator.

TABLE 6–1 `ispAdministrator` Attributes *(continued)*

Attribute name	Mandatory?	Schema	Purpose
`objectClass`	Yes	Base	The object class of the entry (`ispAdministrator`).
`ou`	No	Base	The organizational unit to which the entry belongs. In this release of Solaris for ISPs, the *Administrator* node under the root domain.
`surname`	Yes	Base	The family name of the administrator
`userCertificate`	No	Base	A certificate containing the public key of the administrator.
`userid`	Yes	Base	The login name of the administrator.
`userPassword`	Yes	Base	The password of the administrator.
`uidNumber`	No	Base	A UNIX user-ID. For virtually-hosted FTP or Web services, the user ID specified for the virtual domain.

`ispManagedService` Class

Purpose: Defines an entry representing an ISP service that is managed by Sun Internet Administrator. This object class is reserved for use by Sun Internet Administrator. The `ispManagedService`'s relative distinguished name is the `host` attribute and its value.

TABLE 6–2 `ispManagedService` Attributes

Attribute name	Mandatory?	Schema	Purpose
`associatedName`	No	Base	The distinguished name of the top-level service entry for this service.
`commonName`	No	Base	The user-friendly name of a service, for display in the GUI of Sun Internet Administrator
`host`	Yes	Base	The fully-qualified name of the host where this service is installed.
`ispCategory`	No	Base	The type of user interface supported by this service. See the `ispCategory` attribute section for details.
`ispImageFile`	No	Extension	The name of a GIF image file containing the icon for this service.
`ispServiceLocation`	No	Extension	The path to the X-based administration application for this service.
`ispParameterizedOperation`	No	Extension	Information on a supported command-line utility that accepts parameters.
`ispServlets`	No	Extension	The fully-qualified Java class name of a servlet used in this service's administration user interface.
`ispServletClasspath`	No	Extension	The Java classpath for classes required by this service's administration user interface.
`ispSupportedOperation`	No	Extension	Information on a supported command-line utility that accepts no parameters.

TABLE 6–2 `ispManagedService` Attributes *(continued)*

Attribute name	Mandatory?	Schema	Purpose
`ispVersion`	No	Extension	The release number (major.minor) of the service described by the entry.
`labeledURI`	No	Base	The path to the main GUI page of a service. For a 3-tier service, enter a path relative to the document root. For a 2-tier service, enter the complete URL.
`objectClass`	Yes	Base	The object class of the entry (`ispManagedService`).

`ispService` Class

Purpose: Defines an entry representing a Solaris for ISPs software component. The `ispService`'s relative distinguished name is the the `ispVersion` attribute and its value.

TABLE 6–3 `ispService` Attributes

Attribute name	Mandatory?	Schema	Purpose
`associatedDomain`	No	Base	The domain with which this service is associated.
`commonName`	Yes	Base	The name of the service (not used in Solaris for ISPs).
`description`	No	Base	An arbitrary description of the service.
`host`	No	Base	The fully-qualified name of the host where the service is installed.

TABLE 6–3 `ispService` Attributes *(continued)*

Attribute name	Mandatory?	Schema	Purpose
`ispDirectoryRoot`	No	Extension	A directory prefix to a location on the file system where a domain's content is virtually-hosted. Used by Sun Internet FTP Server and SWS when in a virtual host configuration.
`ispPrivateData`	No	Extension	Software component password information for use by Sun Internet Administrator. This attribute is protected by ACLs from access by a user other than the directory root and Sun Internet Administrator.
`ispServiceContext`	No	Extension	A CORBA naming context used by Sun Internet FTP Server and Sun Internet News Server.
`ispServiceLocation`	No	Extension	A CORBA stringified object reference to the service administration server. (Used by Sun Internet FTP Server and Sun Internet News Server.)
`ispSupplementaryInformation`	No	Extension	Arbitrary information about the service. Reserved for Solaris for ISPs service-specific needs.
`ispVersion`	Yes	Extension	The release number (major.minor) of the service described by the entry.
`labeledURI`	No	Base	The path to the servlets for a three-tier GUI.
`mail`	No	Base	The advertised electronic mail address of the user. Not used by Solaris for ISPs.

TABLE 6–3 `ispService` Attributes *(continued)*

Attribute name	Mandatory?	Schema	Purpose
`objectClass`	Yes	Base	The object class of the entry (`ispService`).
`userCertificate`	No	Base	A certificate containing the public key of the user.
`userPassword`	No	Base	The password of the entry, used for binding to the directory

`ispSubscriber` Class

Purpose: Defines an entry representing a subscriber (customer) of the ISP. The `ispSubscriber`'s relative distinguished name is the `commonName` attribute and its value. Its superior object is `inetOrgPerson`.

If you are using the RADIUS server that comes with Sun Directory Services, overlay the `ispSubscriber` objects with the `remoteUser` object class.

TABLE 6–4 `ispSubscriber` Attributes

Attribute name	Mandatory?	Schema	Purpose
`associatedDomain`	No	Base	The domain with which this subscriber is associated.
`commonName`	Yes	Base	The name of the subscriber described by the entry, in the form Firstname Lastname (userid).
`gidNumber`	No	Base	A UNIX group-ID. For virtually-hosted FTP and Web services, the group ID specified for the virtual domain.
`homeDirectory`	No	Base	The file system location of the home directory of the subscriber described by the entry. (Not used by Solaris for ISPs.)
`host`	No	Base	The fully-qualified name of the host. Not used by Solaris for ISPs.

TABLE 6–4 `ispSubscriber` Attributes *(continued)*

Attribute name	Mandatory?	Schema	Purpose
`ispAuthorizedServices`	No	Extension	The distinguished names of services the subscriber is authorized to use.
`ispContentDirectory`	No	Extension	A directory location where content belonging to a subscriber is located. For virtually-hosted FTP and Web services, the path relative to the `ispRootDirectory`.
`labeledURI`	No	Base	The Uniform Resource Identifier and label associated with the Web page of this subscriber.
`mail`	No	Base	The advertised electronic mail address of the subscriber.
`objectClass`	Yes	Base	The object class of the entry (`ispSubscriber`).
`ou`	No	Base	The organizational unit to which the subscriber belongs (in Solaris for ISPs, the *People* node under a domain entry).
`surname`	Yes	Base	The family name of the subscriber.
`uidNumber`	No	Base	A UNIX user-ID. For virtually-hosted FTP and Web services, the user ID specified for the virtual domain.
`userCertificate`	No	Base	A certificate containing the public key of the subscriber.
`userid`	Yes	Base	The login name of the subscriber.
`userPassword`	Yes	Base	The password of the subscriber.

Solaris for ISPs Attributes

This section describes the attributes used by object classes added to the basic schema for Solaris for ISPs. Some of these attributes are a part of the base schema, but are included here for ease of use. Attributes created for the Solaris for ISPs schema extension begin with “isp.”

Attributes defined in the schema have one of the following syntaxes:

- `bin`: binary
- `ces`: case-exact string

  A case-sensitive alphanumeric string.
- `cis`: case-ignore string

  An alphanumeric string, not case-sensitive.
- `dn`: distinguished name
- `protected`: encrypted

  A value that has been encrypted using crypt(1)
- `int` or `long`: integer
- `tel`: telephone number
- `utctime`: UTC time

The following list presents attributes in alphabetical order.

`associatedDomain` Attribute

Summary: `cis`, multi-valued, base schema

Purpose: Specifies the domain with which the object described by the entry is associated. For the OSI tree entry of a domain, must contain the name (in dot notation, for example, eng.sun.com) of the corresponding DC tree entry.

`associatedName` Attribute

Summary: `dn`, base schema

Purpose: Specifies the distinguished name of an entry associated with this entry. For the DC tree entry of a domain, must contain the distinguished name of the corresponding OSI tree entry.

`commonName` Attribute

Alternate name: `cn`

Summary: `cis`, multi-valued, base schema

Purpose: Specifies the full name of the object described by the entry.

Subscriber and administrator entries use the form *Firstname Lastname (uid)*. For example, user John Smith who uses the login name jsmith would have an entry in

the directory. Its `commonName` attribute would contain the value `John Smith (jsmith).`

The `ispManagedService` object class uses this attribute for the user-friendly name displayed in the Sun Internet Administrator GUI.

`description` Attribute

Summary: `cis`, multi-valued, base schema

Purpose: Specifies an arbitrary description of the entry object.

`gidNumber` Attribute

Summary: `long`, single-valued, base schema

Purpose: Specifies a UNIX group-ID. For a subscriber with virtually-hosted FTP services, must be the group-ID specified for the virtual domain.

`homeDirectory` Attribute

Summary: `ces`, single-valued, base schema

Purpose: Specifies the file system location of the home directory of the user described by the entry. This attribute is not used by Solaris for ISPs, but is left for general information required by the customer.

`host` Attribute

Summary: `cis`, multi-valued, base schema

Purpose: Specifies the name of the machine associated with or managed by the object described by the entry. When used with Solaris for ISPs object entries, the host attribute must contain the fully-qualified host name.

`ispAdministeredService` Attribute

Summary: `dn`, multi-valued, Solaris for ISPs schema extension

Purpose: Specifies the services (by distinguished names of the top-level service entries) that an administrator can manage. The top-level entry for a service is an

ispService object under the root domain entry. For example, under ou=services,o=sun,c=us where sun.com is the root domain entry.

ispAuthorizedServices Attribute

Summary: dn, multivalued, Solaris for ISPs schema extension

Purpose: Specifies the services (by distinguished name) that an ispSubscriber is authorized to use.

ispCategory Attribute

Summary: cis, single-valued, Solaris for ISPs schema extension

Purpose: Specifies the category of graphical user interface supported by the service described by the entry. Acceptable values are:

- 2tier, indicating that the user interface is Web-based and uses the two-tier architecture supported by Sun Internet Administrator.
- 3tier, indicating that the user interface is Web-based and uses the three-tier architecture supported by Sun Internet Administrator.
- CLI, indicating that the user interface is a command-line utility.
- X, indicating that the user interface is an X-based program.

ispContentDirectory Attribute

Summary: ces, single-valued, Solaris for ISPs schema extension

Purpose: Specifies a directory location (which may be distinct from that in the homeDirectory attribute) where content belonging to a user is located. Used by ispAdministrator and ispSubscriber classes. For a user with virtually-hosted FTP services, must contain the path (relative to the ispDirectoryRoot of the domain) to the user's FTP content.

ispDirectoryRoot Attribute

Summary: ces, single-valued, Solaris for ISPs schema extension

Purpose: Specifies a root directory for an ISP service, usually on a per-domain basis.

ispImageFile Attribute

Summary: cis, single-valued, Solaris for ISPs schema extension

Purpose: Specifies the name of a file containing the image described by the entry. The ispManagedService object class uses this attribute to specify its GIF-format icon file.

ispPrivateData Attribute

Summary: ces, single-valued, Solaris for ISPs schema extension

Purpose: Specifies software component password information used by Sun Internet Administrator. This attribute is protected by ACLs from access by users other than the Sun Directory Services root administrator.

ispServiceContext Attribute

Summary: ces, single-valued, Solaris for ISPs schema extension

Purpose: Specifies a service context for use by an ISP service. For Sun Internet FTP Server and Sun Internet News Server, this is a CORBA naming context.

ispServiceLocation Attribute

Summary: ces, single-valued, Solaris for ISPs schema extension

Purpose: Specifies the location of the ispService object described by the entry.

Sun Internet FTP Server and Sun Internet News Server use this attribute in an ispService object to store a CORBA stringified object reference to their administration servers. Sun Internet Administrator uses this attribute in an ispManagedService object to store the path to an X-based user interface application.

ispServlets Attribute

Summary: ces, multivalued, Solaris for ISPs schema extension

Purpose: Specifies the fully-qualified Java class name of a servlet used in an ispManagedService's administration user interface. For each servlet used, assign an ispServlets attribute with the following value:

1. The path to the servlet, relative to the document root of the administration Web server (part of Sun Internet Administrator)

2. The fully-qualified Java class name of the servlet
3. Any required servlet arguments, listed by name and value

`ispServletClasspath` Attribute

Summary: `ces`, single-valued, Solaris for ISPs schema extension

Purpose: Specifies the Java classpath for classes required by an `ispManagedService`'s administration user interface. A Java classpath may contain several paths; separate them with colons (:).

`ispParameterizedOperation` Attribute

Summary: `ces`, multivalued, Solaris for ISPs schema extension

Purpose: Specifies information about a command-line function that takes parameters. For an `ispManagedService` object, three space-delimited fields of information are required:

1. The complete path to the command-line utility
2. The complete path to a help file for the utility, or `NONE` if there is no help information for the utility.
3. An arbitrary string, containing the space-delimited parameters required by the utility. If parameters are accepted from the user on the command line, this field holds the string `SOME`.

`ispSupplementaryInformation` Attribute

Summary: `cis`, multivalued, base schema

Purpose: Holds additional information concerning the object described by the entry. In object classes that extend the base schema, this attribute is reserved for Solaris for ISPs service-specific needs.

`ispSupportedOperation` Attribute

Summary: `ces`, multivalued, Solaris for ISPs schema extension

Purpose: Stores information on a supported command-line utility that accepts no parameters. For an `ispManagedService` object, three space-delimited fields of information are required:

1. The complete path to the command-line utility
2. The complete path to a help file for the utility, or `NONE` if there is no help information for the utility.
3. The string `NONE`.

`ispVersion` Attribute

Summary: `ces`, single-valued, Solaris for ISPs schema extension

Purpose: Specifies the release version of the `ispService` object described by this entry. Solaris for ISPs uses the form major.minor for the version attribute.

`labeledURI` Attribute

Alternate name: `labeledURL`

Summary: `ces`, multi-valued, base schema

Purpose: Specifies a Uniform Resource Identifier (URI) and label associated with the object described by the entry.

`mail` Attribute

Alternate name: `preferredRfc822Originator`

Summary: `cis`, multi-valued, base schema

Purpose: Specifies the advertised electronic mail address, in RFC822 format, of the object described by the entry. Sun Internet FTP Server uses this attribute as an error-reporting address.

`objectClass` Attribute

Summary: `cis`, multivalued, base schema

Purpose: Specifies the object class of the type of entry.

ou Attribute

Alternate name: organizationUnitName

Summary: cis, base schema

Purpose: Specifies the name of the organization unit to which the object described by the entry belongs.

surname Attribute

Alternate name: sn

Summary: cis, base schema

Purpose: Specifies the surname of the person described by the entry.

uidNumber Attribute

Summary: long, single-valued, base schema

Purpose: Specifies a UNIX user-ID. For a subscriber with virtually-hosted FTP services, must be the user-ID specified for the virtual domain.

userCertificate Attribute

Summary: bin, base schema

Purpose: Specifies a certificate containing the public key of the user described by the entry.

userid Attribute

Alternate name: uid

Summary: cis, multi-valued, base schema

Purpose: Specifies the login name of the user described by the entry.

userPassword Attribute

Summary: protected, multi-valued, base schema

Purpose: Specifies the password for the entity described by the entry.

CHAPTER 7

Integrating Existing Service Applications

Solaris™ for ISPs™ is designed to allow integration of your existing services with Sun™ Internet Administrator™. When you integrate these services, they gain the benefits of centralized remote administration and administrator account management provided by that product. Because Sun Internet Administrator manages administrator access, integrating an application with it is an easy way to add the security of administrator authorization to an existing service application.

General Steps to Integrating an Existing Service

Sun Internet Administrator integrates and manages Web-based, X-based, and command-line administration interfaces using the following, general steps.

1. Install the service application on a computer that is accessible by Sun Internet Administrator via the network. This host computer must have the Solaris for ISPs platform component (SUNWisp) installed on it, to provide platform extensions that Sun Internet Administrator uses.
2. Run the `mcreg(1m)` command on the service host computer with the correct parameters for this type of service. The form of the `mcreg` command is different for each type of service and is fully described later in this chapter as well as on the man page for the command. You must have root access to run `mcreg`.

Note - You must run `mcreg` again if you change the URL or other service configuration information (for example, if you change the port where a Web server is running). If you run `mcreg` more than once for a service (on a single host machine), the later information overwrites the information recorded previously.

---

3. In Sun Internet Administrator, manage the service host and select the new service.
4. Also in Sun Internet Administrator, create administrators with access to the new service or modify existing administrators to grant them access.

   You must be an administrator with rights to manage Sun Internet Administrator to perform these tasks.

Integrating X-Based Services

When the service application is installed on a computer in the network, run the `mcreg(1m)` command to store information about the application. This information is used by Sun Internet Administrator to correctly handle the presentation and launch of the administration GUI. The information you need for the `mcreg` command is:

- The component identifier, a unique string that identifies the service application. Because they are unique, Solaris for ISPs recommends using package names as component identifiers. The service application developer chooses the component identifier.
- A user-friendly name, appropriate for presentation to the administrator. This is the service name that Sun Internet Administrator displays in its *Manage Services* screen, where the administrator accesses all services. The service application developer chooses the application name.
- The version number of the application being registered. The recommended format for the version number is *major.minor* (for example, 1.2). The service application developer chooses the version number.
- The complete file system path to the X-based administration interface executable on the service host.
- The user name under which the administration interface runs. This identifies the UNIX UID that Sun Internet Administrator uses to invoke the program.
- The group name under which the administration interface runs. This identifies the UNIX GID that Sun Internet Administrator uses to invoke the program.

When you have all required information, obtain root access and register the information as shown below. Run this command on the computer where the service administration interface is installed.

 # mcreg –c *componentID* –n *name* –v *version* –x *X_path* –u *user_name* –g *group_name*

See the `mcreg(1m)` man page for an example of this form of the command.

After running `mcreg`, you can log into Sun Internet Administrator as a console administrator and register the new application for management. If you want other administrators to have rights to manage the new application, edit their administrator access to include it. The Sun Internet Administrator on-line help has full instructions on performing these tasks.

When an administrator invokes the X application from Sun Internet Administrator, the display is routed to display zero (*hostname*:0) on the client machine. Any `stdout` or `stderr` messages generated by the X application are displayed in the Sun Internet Administrator *X Application Output* screen.

Integrating Command-Line Programs

To integrate a command-line administration program with Sun Internet Administrator, run the `mcreg(1m)` command to store information about the application. This information is used by Sun Internet Administrator to correctly handle the presentation and launch of the administration GUI. The information you need for the `mcreg` command is:

- The component identifier, a unique string that identifies the service application. The service application developer chooses the component identifier.
- A user-friendly name, appropriate for presentation to the administrator. This is the service name that Sun Internet Administrator displays in its *Manage Services* screen, where the administrator accesses all services. The service application developer chooses the application name.
- The version number of the application being registered. The recommended format for the version number is *major.minor* (for example, 1.2). The service application developer chooses the version number.
- The complete path to the command-line program, plus information about its parameters and documentation. Enclose this command option in quotes (“). There are three fields:
  - The full path to the executable, with any static parameters. For example, `-p ‘‘/usr/bin/ps -ef’’`. This field is required.
  - A **`-a`** indicates that this command takes parameters from the user at run time. This field is optional
  - A **`-h helpfile`** indicates the on-line documentation associated with this command. This field is optional.
- The user name under which the administration interface runs. This identifies the UNIX UID that Sun Internet Administrator uses to invoke the program.
- The group name under which the administration interface runs. This identifies the UNIX GID that Sun Internet Administrator uses to invoke the program.

When you have all required information, obtain root access and register the information as shown below. Run this command on the computer where the service administration interface is installed.

 # mcreg –c *componentID* –n *name* –p "*prog_path* [–a] [–h *help_file*]"... –v *version* –u *user_name* –g *group_name*

See the `mcreg(1m)` man page for examples of this and other forms of the command.

To record information about multiple command-line operations, enter multiple –p arguments.

After running `mcreg`, you can log into Sun Internet Administrator as a console administrator and register the new application for management. If you want other administrators to have rights to manage the new application, edit their administrator access lists to include it. The Sun Internet Administrator on-line help has full instructions on performing these tasks.

Integrating Two-Tier Web-Based Applications

A two-tier Web-based application is a existing application whose administration interface is accessed through a Web browser. It can be implemented in HTML, CGI, or Java Applets and Servlets (or some combination). Such an interface requires a Web server installed and running on the computer where the service is installed.

Because this interface is accessible via URLs, there are some security issues to consider. The interface should be protected by the Web server's ACLs so only authorized administrators can access it. To take advantage of the administrator management provided by Sun Internet Administrator, you must use Sun™ WebServer™ (SWS) and configure it with the same ACLs used by the Sun Internet Administrator administration Web server (see "Configuring for Administrator Account Coordination" on page 85).

To secure the connection between the browser and the administration interface, consider secure HTTP (HTTPS) or SKIP. See the SWS on-line help and the SKIP man pages (`/opt/SUNWicp/man`) for instructions on how to configure these security tools.

Registering Information for a Two-Tier Web-Based Application

When the service application is installed on a computer in the network, run the `mcreg(1m)` command to store information about the application. This information is used by Sun Internet Administrator to correctly handle the presentation and launch of the administration GUI. The information you need for the `mcreg` command is:

- The component identifier (*componentID*), a unique string that identifies the service application. Because they are unique, Solaris for ISPs recommends using package names as component identifiers. The service application developer chooses the component identifier.
- A user-friendly *name*, appropriate for presentation to the administrator. This is the service name that Sun Internet Administrator displays in its *Manage Services* screen, where the administrator accesses all services. The service application developer chooses the application name.
- The version number of the application being registered. The recommended format for the version number is *major.minor* for example, 1.2). The service application developer chooses the version number.
- The *URL* to the entry point of the Web-based administration interface. Enter the absolute (not a relative) path.

When you have all required information, obtain root access and register the information as shown below. Run this command on the computer where the service administration interface is installed.

 # mcreg –c *componentID* –n *name* –v *version* –w *URL*

See the `mcreg(1m)` man page for an example of this form of the command.

After running `mcreg`, you can log into Sun Internet Administrator as a console administrator and register the new application for management. If you want other administrators to have rights to manage the new application, edit their administrator access lists to include it. The Sun Internet Administrator on-line help has full instructions on performing these tasks.

Configuring for Administrator Account Coordination

If the two-tier service application uses SWS for its administration interface, it is possible to configure the server's ACLs to use the same administrator login information as Sun Internet Administrator. Thus, you have a single set of administrator accounts (in the directory services) to manage. This simplifies management of these accounts and enhances your ability to respond to security risk situations.

You must first install SWS on the service host (the computer where the service application is installed), and arrange the interface documents and files within the document tree of the server. Then, configure SWS as follows:

1. Create a realm in the default Web site of the SWS instance:

   # htrealm add –i *instance* –h *hostname* –r *realmname* –s *ISPADMIN* –d *ComponentID-VersionNo*

   Where

- *instance* is the name of the httpd instance being configured.
- *hostname* is the name of the host containing the realm.
- *realmname* is the name of the realm you are creating.
- *ISPAdmin* is the source of the realm. Enter ISPADMIN, indicating that these are Sun Internet Administrator administrators.
- *ComponentID-VersionNo* is the directory where the administration GUI files (HTML and others) are located.

2. Add ACL protection to the URL where the administration GUI for this service resides:

   # htaccess add –i *instance* –h *hostname* –U *URI* –r *realname* –s *BASIC*

   Where

   - *instance* is the name of the httpd instance being configured.
   - *hostname* is the name of the computer where the Web server is running.
   - *realmname* is the name of the realm you are creating.
   - *BASIC* is the authentication scheme wanted. Enter BASIC.

3. Restart the default site, if it is already running. Use the SWS administration GUI to perform this task. Full instructions are available in the on-line help.

Index

A

access control
  Sun Directory Services, 60
  SWS, 85
administration tools
  Sun Directory Services, 47
  Sun Internet Administrator, 2, 10
administration Web server, 40, 85
architecture
  Solaris for ISPs, 1
  three-tier applications, 11
  two-tier applications, 12
associatedDomain attribute, 73
associatedName attribute, 73
attributes (SunDS schema), 72
  associatedDomain, 73
  associatedName, 73
  cn (commonName), 73
  commonName, 73
  description, 74
  gidNumber, 74
  homeDirectory, 74
  host, 74
  ispAdministeredService, 74
  ispAuthorizedServices, 75
  ispCategory, 75
  ispContentDirectory, 75
  ispDirectoryRoot, 75
  ispImageFile, 76
  ispParameterizedOperation, 77
  ispPrivateData, 76
  ispServiceContext, 76
  ispServiceLocation, 76
  ispServletClasspath, 77
  ispServlets, 76
  ispSupplementaryInformation, 77
  ispSupportedOperation, 77
  ispVersion, 78
  labeledURI, 78
  labeledURL (labeledURI), 78
  mail, 78
  objectClass, 78
  organizationUnitName (ou), 79
  ou, 79
  preferredRfc822Originator (mail), 78
  sn (surname), 79
  surname, 79
  syntax defined, 73
  uid (userid), 79
  uidNumber, 79
  userCertificate, 79
  userid, 79
  userPassword, 79
auditing, 38
AWS, *see* administration Web server,

B

backing up the schema, 64
basic network layout, 28

C

changes to Solaris, 32
    onetime only, 33
    reconfigurable, 34
    undoing the changes, 37
classes (SunDS schema), 65
    ispAdministrator, 66
    ispManagedService, 67
    ispService, 69
    ispSubscriber, 71
closing security holes, 32, 34
cn (commonName) attribute, 73
command-line applications
    integrating, 81, 83
    specifying ispCategory, 75
commonName attribute, 73
component default ports, 46
configuration scenario, 8, 9
configuration software features, 3
configuring
    AWS (Sun Internet Administrator), 40
    Sun Directory Services, 22
    Sun Internet Administrator, 23
    Sun WebServer, 85
conserving resources, 36
console, *see* Sun Internet Administrator,
CPU required, 44
creating directory entries, 51
    command-line procedure, 51
    complete ldif example, 58
    domains, 53
    group organizational units, 59
    remote access information, 57
    subscribers, 54
    virtual hosting information, 56

D

DC tree
    illustrated, 50
    structure, 50
deinstalling Solaris for ISPs, *see* Solaris for ISPs Installation Guide,
dependencies of software, 45
description attribute, 74
designing the network, 27
    base setup example, 28
    expanded setup example, 30
detecting intrusion, 38
directory information tree, 48
directory services, 4, 47
directory structure, 48
disk space required, 44
DIT, *see* directory information tree,
DNS, 29, 30
domain component tree, *see* DC tree,
Domain Name Service, *see* DNS,
domains
    defined, 49
    directory entries, 53, 73
    root domain, 48
    structure in DIT, 50

E

encryption, 4
entries, *see* Sun Directory Services, creating entries,
expanded network layout, 29
extensions, *see* platform extensions,

F

features
    FlexLM, 5
    host configuration software, 3
    HotJava browser, 5
    Java Development Kit, 5
    Solaris for ISPs, 1
    Sun Directory Services, 4
    Sun Internet Administrator, 2
    Sun Internet FTP Server, 7
    Sun Internet News Server, 7
    Sun Internet Services Monitor, 3
    Sun WebServer, 6
    SunScreen SKIP, 4
FlexLM, 5
FrontPage support, 6
FTP, *see* Sun Internet FTP Server,

G

gidNumber attribute, 74
groups (SunDS), 59
guidelines for installing, 27

H

hardening and tuning, 32
hardening and tuning Solaris, 3, 8, 32
hardening security, 33
hardware requirements, 43
    base setup, for, 28
    expanded setup, for, 29
homeDirectory attribute, 74
host attribute, 74
host configuration
    approach, 8
    changes to Solaris, 32
    features, 3
    hardening security, 33
    model, 8
    repeatable, 9
    tuning Solaris services, 34
HotJava browser, 5
htaccess command, 86
htrealm command, 85

I

installation approach, 8
installing Solaris for ISPs
    See also Solaris for ISPs Installation Guide,
    approach, 8
    example scenario, 27
    hardware requirements, 43
    host configuration software, 3
    how to plan, 15
    network setup examples, 27
    operating system requirements, 42
    overview, 16
    planning phase, 18
    post-installation tasks, 24
    preparing phase, 19
    process, 15, 16
    requirements, 41
integrating applications, 81
    command-line applications, 83
    Web applications, 84
    X applications, 82
intrusion detection, 3, 38
ispAdministeredService attribute, 74
ispAdministrator class, 61, 66
ispAuthorizedService attribute, 61
ispAuthorizedServices attribute, 75
ispCategory attribute, 75
ispContentDirectory attribute, 75
ispDirectoryRoot attribute, 75
ispImageFile attribute, 76
ispManagedService class, 61, 67
ispParameterizedOperation attribute, 77
ispPrivateData attribute, 61, 76
ispService class, 69
ispServiceContext attribute, 76
ispServiceLocation attribute, 76
ispServletClasspath attribute, 77
ispServlets attribute, 76
ispSubscriber class, 71
ispSupplementaryInformation attribute, 77
ispSupportedOperation attribute, 77
ispVersion attribute, 78

J

Java Development Kit, 5
JumpStart, xiii, 9

L

labeledURI attribute, 78
labeledURL (labeledURI) attribute, 78
layout of network, 27
    basic example, 28
    expanded example, 29
LDAP, 4, 47
ldapadd command, 51
ldapdelete command, 51
ldapmodify command, 51
ldif file
    complete example, 58
    structure, 52
license server, 5
log file management, 3, 38
logging, 3, 38

M

mail attribute, 78
maintaining the schema (SunDS), 63
man page patches, 43
management console, *see* Sun Internet Administrator,
managing services, 10
mcreg command, 81, 82, 84, 85
monitoring, *see* Sun Internet Services Monitor,

N

naming contexts, 48, 50
network setup, 27
    base example, 28
    expanded example, 29
news, *see* Sun Internet News Server,

O

object classes (SunDS schema), 65
    ispAdministrator, 66
    ispManagedService, 67
    ispService, 69
    ispSubscriber, 71
objectClass attribute, 78
onetime changes to Solaris, 33
Open Systems Interconnection tree, *see* OSI tree,
operating system requirements, 42
organizational unit entries, 48
organizationUnitName (ou) attribute, 79
OSI tree
    entries illustrated, 49
    structure, 48
ou attribute, 79

P

patches required and recommended, 42
planning network, 27
Planning the installation, 18
platform extensions, 2
    FlexLM license server, 5
    host configuration software, 3
    HotJava browser, 5
    Java Development Kit, 5
    Sun Directory Services, 4
    Sun Internet Administrator, 2
    Sun Internet Services Monitor, 3
    Sunscreen SKIP, 4
ports for components, 46, 81
post-configuration commands, *see* user-defined scripts,
post-installation scripts, *see* user-defined scripts,
post-installation tasks, 24
preferredRfc822Originator (mail) attribute, 78
product architecture, 1
protecting the network, 31, 32

R

RADIUS, 57, 71
RAM required, 44
reconfigurable changes to Solaris, 34
    closing security holes, 34
    conserving resources, 36
remote access information, 57
restoring the schema (SunDS), 64
root domain, 48

S

scenario, 8, 9
schema (SunDS), 63
    attributes, 72
    backup, 64
    maintenance, 63
    object classes, 65
    restoring, 64
scripts (user-defined), 39
security
    features, 10
        hardening and tuning, 32
        host configuration software, 3
        SKIP, 4
        Sun Internet Administrator, 2
        SWS, 6
    patches, 42
    risks, 30
        Solaris for ISPs, 31
        standard Internet, 30
server process management, 3
service integration

command-line applications, 83
three-tier architecture, 11
two-tier architecture, 12
Web applications, 84
X applications, 82
services, 6
Sun Internet FTP Server, 7
Sun Internet News Server, 7
Sun WebServer, 6
servlet support, 6
setup guidelines, 27
SKIP, *see* SunScreen SKIP,
sn (surname) attribute, 79
software dependencies, 45
Solaris for ISPs
access control, 60
architecture, 1
security issues, 31
Solaris services, 34
SPM, *see* server process management,
subscriber entries (SunDS), 54, 58
basic entry, 55
RADIUS access, 57
virtual hosting, 56
Sun Directory Services
access control, 60
administration tools, 47
backup, 64
command-line programs, 51
configuring after install, 22
creating entries, 51, 54
directory structure, 48
features, 4
groups, 59
role in Solaris for ISPs, 47
schema, 63
Sun Internet Administrator
configuring after install, 23
features, 2
managing services, 10
operations, 11, 12
security benefits, 10
three-tier architecture, 11
two-tier architecture, 12
Sun Internet FTP Server, 7
Sun Internet News Server, 7
Sun Internet Services Monitor, 3
Sun WebServer
configuring, 85
features, 6
SunDS, *see* Sun Directory Services,
SunScreen SKIP, 4
surname attribute, 79
swap space required, 44
SWS, *see* Sun WebServer,
syslog logging, 38
system changes, 33

T

three-tier architecture, 10, 11
tightening security, 32
tuning Solaris services, 34
two-tier architecture, 10, 12

U

uid (userid) attribute, 79
uidNumber attribute, 79
undoing changes to Solaris, 37
uninstalling Solaris for ISPs, *see* Solaris for ISPs Installation Guide,
user-defined scripts, 39
userCertificate attribute, 79
userid attribute, 79
userPassword attribute, 61, 79

V

version of Solaris, 42
virtual host information, 56, 79, 74

W

Web browser supported, 45
Web server, *see* Sun WebServer,
Web-based applications
coordinating administrators, 85
integrating, 81, 84
registering, 84
specifying ispCategory, 75

X

X applications
integrating, 81, 82

specifying ispCategory, 75