Solaris 10 Installation Guide: Network-Based Installations

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

Part No: 817–5504–10
January 2005

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

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REPONDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.

Adobe PostScript

041118@10082

Contents

Preface

This book describes how to install the Solaris™ Operating System (Solaris OS) remotely over a local area network or a wide area network.

This book does not include instructions about how to set up system hardware or other peripherals.

Note – This Solaris release supports systems that use the SPARC® and x86 families of processor architectures: UltraSPARC®, SPARC64, AMD64, Pentium, and Xeon EM64T. The supported systems appear in the *Solaris 10 Hardware Compatibility List* at `http://www.sun.com/bigadmin/hcl`. This document cites any implementation differences between the platform types.

In this document the term "x86" refers to 64-bit and 32-bit systems manufactured using processors compatible with the AMD64 or Intel Xeon/Pentium product families. For supported systems, see the *Solaris Hardware Compatibility List*.

Who Should Use This Book

This book is intended for system administrators who are responsible for installing the Solaris software. This book provides advanced Solaris installation information for enterprise system administrators who manage multiple Solaris machines in a networked environment.

For basic installation information, see *Solaris 10 Installation Guide: Basic Installations*.

Related Books

Table P–1 lists related information that you need when you install the Solaris software.

TABLE P–1 Related Information

Information	Description
Solaris 10 Installation Guide: Basic Installations	This book describes how to perform a basic Solaris installation with a graphical user interface (GUI).
Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning	This book describes how to use CD or DVD media to upgrade a system to the Solaris OS. This book also describes how to use the Solaris Live Upgrade feature to create and maintain boot environments, and how to upgrade systems to these boot environments.
Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations	This book describes how to create the files and directories necessary to perform an unattended custom JumpStart installation. This book also describes how to create RAID-1 volumes during a JumpStart installation. This book describes how to create a Solaris Flash archive and deploy the archive over the network to quickly install the Solaris OS. This book also describes how to maintain these archives, and how to quickly update clone systems by using differential Flash archives.
Solaris 10 Installation Guide: Solaris Flash Archives (Creation and Installation)	This book describes how to create a Solaris Flash archive and deploy the archive over the network to quickly install the Solaris OS. This book also describes how to maintain these archives, and how to quickly update clone systems by using differential Flash archives.
System Administration Guide: Devices and File Systems	This book describes how to back up system files.
Solaris 10 Release Notes	This book describes any bugs, known problems, software that is being discontinued, and patches that are related to the Solaris release.
SPARC: *Solaris 10 Sun Hardware Platform Guide* on `http://docs.sun.com`	This book contains information about supported hardware.
Solaris 10 Package List	This book lists and describes the packages in the Solaris 10 OS.
x86: Solaris Hardware Compatibility List	This list contains supported hardware information and device configuration details.

Accessing Sun Documentation Online

The docs.sun.com^{SM} Web site enables you to access Sun technical documentation online. You can browse the docs.sun.com archive or search for a specific book title or subject. The URL is `http://docs.sun.com`.

Ordering Sun Documentation

Sun Microsystems offers select product documentation in print. For a list of documents and how to order them, see “Buy printed documentation” at `http://docs.sun.com`.

Typographic Conventions

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

TABLE P–2 Typographic Conventions

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

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

AaBbCc123	Command-line placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.

TABLE P–2 Typographic Conventions *(Continued)*

Typeface or Symbol	Meaning	Example
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. Perform a *patch analysis*. Do *not* save the file. [Note that some emphasized items appear bold online.]

Shell Prompts in Command Examples

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–3 Shell Prompts

Shell	Prompt
C shell prompt	`machine_name%`
C shell superuser prompt	`machine_name#`
Bourne shell and Korn shell prompt	`$`
Bourne shell and Korn shell superuser prompt	`#`

PART I Planning to Install Over the Network

This part describes how to plan your installation over the network.

CHAPTER 1

Solaris Installation and Upgrade (Roadmap)

This chapter provides you with information about decisions you need to make before you install or upgrade the Solaris Operating System (Solaris OS). This chapter contains the following sections:

Note – This book uses the term *slice*, but some Solaris documentation and programs might refer to a slice as a partition.

x86: To avoid confusion, this book distinguishes between x86 fdisk partitions and the divisions within the Solaris fdisk partition. The x86 fdisk divisions are called partitions. The divisions within the Solaris fdisk partition are called slices.

Task Map: Installing or Upgrading the Solaris Software

The following task map is an overview of the steps necessary to install or upgrade the Solaris OS when using any installation program. Use this task map to identify all of the decisions that you need to make to complete the most efficient installation for your environment.

TABLE 1–1 Task Map: Installing or Upgrading the Solaris Software

Task	Description	For Instructions
Choose initial installation or upgrade.	Decide if you want to perform an initial installation or an upgrade.	"Initial Installation, or Upgrade?" on page 20.
Choose an installation program.	The Solaris OS provides several programs for installation or upgrade. Choose the installation method that is most appropriate for your environment.	"Choosing a Solaris Installation Method" on page 21.
(Solaris installation program) Choose a default or custom installation.	Decide which type installation is suitable for your environment. ■ If you are using a graphical user interface (GUI) you can choose a default or a custom installation. ■ A default installation formats the hard disk and installs a preselected set of software, including the Sun Java™ Enterprise System. ■ A custom installation enables you to modify the hard disk layout and select the software that you want to install. ■ If you use a text installer (non-graphical interface), you can select the default values or edit the values to select the software you want to install.	For information about the Sun Java Enterprise System, see *Sun Java Enterprise System Technical Overview* at `http://docs.sun.com`
Review system requirements. Also, plan and allocate disk space and swap space.	Determine if your system meets the minimum requirements to install or upgrade. Allocate disk space on your system for the components of the Solaris OS that you want to install. Determine the appropriate swap-space layout for your system.	Chapter 2.
Choose to install a system from local media or from the network.	Decide on the most appropriate installation media for your environment.	"Installing From the Network or From DVD or CDs?" on page 19.

TABLE 1–1 Task Map: Installing or Upgrading the Solaris Software *(Continued)*

Task	Description	For Instructions
Gather information about your system.	■ For the Solaris installation program, complete the worksheet to collect all of the information that you need to install or upgrade. ■ For the custom JumpStart installation method, decide which profile keywords to use in your profile. Then review the keyword descriptions to find the information about your system that you need.	■ For the Solaris installation program, see either of the following documents: ■ Chapter 3 ■ Chapter 3, Gathering Information Before Installation or Upgrade (Planning) ■ For the custom JumpStart installation method, see Chapter 9, "Custom JumpStart (Reference)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.
(Optional) Set system parameters.	You can preconfigure system information to avoid being prompted for the information during the installation or upgrade.	Chapter 4.
(Optional) Prepare to install the Solaris software from the network.	If you chose to install the Solaris software from the network, create an installation server, create a boot server (if necessary), and set up the systems to be installed from the network.	To install over a local area network, see Chapter 7. To install over a wide area network, see Chapter 11.
(Upgrade only) Perform tasks prior to upgrade.	Back up your system and determine if you can upgrade with disk space reallocation.	"Upgrade" on page 31.
Perform an installation or upgrade.	Use the Solaris installation method that you chose to install or upgrade the Solaris software.	The chapter or chapters that provide detailed instructions for the installation programs.
Troubleshoot installation problems	Review the troubleshooting information when you encounter problems with your installation.	Appendix A.

Installing From the Network or From DVD or CDs?

The Solaris software is distributed on DVD or CD media so that you can install or upgrade systems that have access to a DVD-ROM or CD-ROM drive.

You can set up the systems to install from the network with remote DVD or CD images. You might want to set up systems this way for the following reasons:

- If you have systems that do not have local DVD-ROM or CD-ROM drives
- If you are installing several systems and do not want to insert the discs into every local drive to install the Solaris software

You can use all of the Solaris installation methods to install a system from the network. However, by installing systems from the network with the Solaris Flash installation feature or with a custom JumpStart installation, you can centralize and automate the installation process in a large enterprise. For more details about the different installation methods, refer to "Choosing a Solaris Installation Method" on page 21.

Installing the Solaris software from the network requires initial setup. For information about preparing to install from the network, choose one of the following options.

For detailed instructions about preparing to install from a local area network	Chapter 7
For instructions about preparing to install over a wide area network	Chapter 11
For instructions about how to install x86 based clients over the network by using PXE	"x86: Booting and Installing Over the Network With PXE" on page 287

Initial Installation, or Upgrade?

You can choose to perform an initial installation or, if your system is already running the Solaris OS, you can upgrade your system.

Initial Installation

An initial installation overwrites the system's disk with the new version of the Solaris OS. If your system is not running the Solaris OS, you must perform an initial installation.

If the system is already running the Solaris OS, you can choose to perform an initial installation. If you want to preserve any local modifications, before you install, you must back up the local modifications. After you complete the installation, you can restore the local modifications.

You can use any of the Solaris installation methods to perform an initial installation. For detailed information about the different Solaris installation methods, refer to "Choosing a Solaris Installation Method" on page 21.

Upgrade

You can upgrade the Solaris OS by using two upgrade methods: standard and Solaris Live Upgrade. A standard upgrade maintains as many existing configuration parameters as possible of the current Solaris OS. Solaris Live Upgrade creates a copy of the current system. This copy can be upgraded with a standard upgrade. The upgraded Solaris OS can then be switched to become the current system by a simple reboot. If a failure occurs, you can switch back to the original Solaris OS with a reboot. Solaris Live Upgrade enables you to keep your system running while you upgrade and enables you to switch back and forth between Solaris OS releases.

You can upgrade any system that is running the Solaris 7, Solaris 8, or Solaris 9 software. For more information about upgrading and the list of upgrade methods, see "Upgrade" on page 31.

Choosing a Solaris Installation Method

The Solaris OS provides several programs for installation or upgrade. Each installation technology offers different features that are designed for specific installation requirements and environments. Use the following table to help you decide which installation method to use.

TABLE 1–2 Choosing Your Installation Method

Task	Installation Method	Instructions
Install one system from CD-ROM or DVD-ROM media with an interactive program.	Solaris installation program	*Solaris 10 Installation Guide: Basic Installations*
Install one system over a local area network.	Solaris installation program over the network	Part II
Automate the installation or upgrade of multiple systems based on profiles you create.	Custom JumpStart	Chapter 4, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*
Replicate the same software and configuration on multiple systems.	Solaris Flash archives	Chapter 1, "Solaris Flash (Overview)," in *Solaris 10 Installation Guide: Solaris Flash Archives (Creation and Installation)*
Install systems over a wide area network (WAN) or the Internet.	WAN boot	Chapter 9

TABLE 1–2 Choosing Your Installation Method *(Continued)*

Task	Installation Method	Instructions
Upgrade a system while it is running.	Solaris Live Upgrade	Chapter 4, "Solaris Live Upgrade (Overview)," in *Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning*
After installing the Solaris OS, create an isolated application environment.	Solaris Zones	Chapter 16, "Introduction to Solaris Zones," in *System Administration Guide: Solaris Containers—Resource Management and Solaris Zones*

Sun Java Enterprise System Software

The Sun Java™ Enterprise System software provides applications for enterprise systems that can be installed when you install the Solaris OS. These applications are for production environments or Internet service providers. When using the Solaris installation program, you can install the software with either the custom or default options. If you choose the default option, you install a preselected set of Java Enterprise System software. If you choose the custom option, you install only the Java Enterprise System software that you want.

For information about configuring Sun Java Enterprise System, see *Sun Java Enterprise System Technical Overview*.

Sun Java System Application Server Platform Edition 8

The Sun Java System Application Server Platform Edition 8 provides for broad deployment of application services and web services. This software is automatically installed with the Solaris OS. You can find documentation for the server in the following areas:

For documentation about starting the server	See *Sun Java System Application Server Platform Edition 8 QuickStart Guide* in the installation directory at `/docs/QuickStart.html`
For the full Application Server documentation set	`http://docs.sun.com/db/coll/ApplicationServer8_04q2`

For a tutorial	`http://java.sun.com/j2ee/1.4/docs/tutorial/doc/index.html`

CHAPTER 2

Solaris Installation and Upgrade (Planning)

This chapter describes system requirements to install or upgrade to the Solaris OS. General guidelines for planning the disk space and default swap space allocation are also provided. This chapter contains the following sections:

System Requirements and Recommendations

TABLE 2–1 Memory, Swap, and Processor Recommendations

System	Size
Memory to install or upgrade	256 MB is the recommended size. 64 MB is the minimum size. **Note –** Some optional installation features are enabled only when sufficient memory is present. For example, if you install from a DVD with insufficient memory, you install through the Solaris installation program's text installer, not through the graphical user interface (GUI). For more information about these memory requirements, see Table 2–2.

TABLE 2–1 Memory, Swap, and Processor Recommendations *(Continued)*

System	Size
Swap area	512 Mbytes is the default size. **Note –** You might need to customize the swap space. Swap space is based on the size of the system's hard disk.
x86: Processor requirements	SPARC: 200–MHz or faster processor is required. x86: 120–MHz or faster processor is recommended. Hardware floating-point support is required.

You can choose to install the software with a GUI or with or without a windowing environment. If there is sufficient memory, the GUI is displayed by default. Other environments are displayed by default if memory is insufficient for the GUI. You can override defaults with the `nowin` or `text` boot options. But, you are limited by the amount of memory in your system or by installing remotely. Also if the Solaris installation program does not detect a video adapter, it automatically displays in a console-based environment. Table 2–2 describes these environments and lists minimal memory requirements for displaying them.

TABLE 2–2 Memory Requirements for Display Options

Memory	Type of Installation	Description	Windowing Environment
64–127 MB	Console-based	Contains no graphics and no windowing environment. If no video adapter is detected, the installer displays a console-based environment. If you are installing remotely through a `tip` line or using the `nowin` boot option, you are limited to the console-based installation.	
128–383 MB	Console-based	Contains no graphics, but provides a window and the ability to open other windows. Requires a local or remote DVD-ROM or CD-ROM drive or network connection, video adapter, keyboard, monitor. If you install by using the `text` boot option and have enough memory, you are installing in a windowing environment.	X
384 MB and greater	GUI-based	Provides windows, pull-down menus, buttons, scrollbars, and iconic images. A GUI requires a local or remote DVD-ROM or CD-ROM drive or network connection, video adapter, keyboard, monitor.	X

Allocating Disk and Swap Space

Before you install the Solaris software, you can determine if your system has enough disk space by doing some high-level planning.

General Disk Space Planning and Recommendations

Planning disk space is different for everyone. Consider allocating space for the following conditions, depending on your needs.

TABLE 2–3 General Disk Space and Swap Space Planning

Conditions for Space Allocations	Description
File systems	For each file system that you create, allocate an additional 30 percent more disk space than you need to enable you to upgrade to future Solaris versions. By default, the Solaris installation methods create only root (/) and `/swap`. When space is allocated for OS services, the `/export` directory is also created. If you are upgrading to a major Solaris release, you might need to reslice your system or allocate double the space that you need at installation time. If you are upgrading to an update, you could prevent having to reslice your system by allocating extra disk space for future upgrades. A Solaris update release needs approximately 10 percent more disk space than the previous release. You can allocate an additional 30 percent of disk space for each file system to allow space for several Solaris updates.
The `/var` file system	If you intend to use the crash dump feature `savecore`(1M), allocate double the amount of your physical memory in the `/var` file system.

TABLE 2–3 General Disk Space and Swap Space Planning *(Continued)*

Conditions for Space Allocations	Description
Swap	The Solaris installation program allocates a default swap area of 512 Mbytes under the following conditions: ■ If you use the installation program's automatic layout of disk slices ■ If you avoid manually changing the size of the swap slice By default, the Solaris installation programs allocate swap space by placing swap so that it starts at the first available disk cylinder (typically cylinder 0 on SPARC based systems). This placement provides maximum space for the root (`/`) file system during the default disk layout and enables the growth of the root (`/`) file system during an upgrade. If you think you might need to expand the swap area in the future, you can place the swap slice so that it starts at another disk cylinder by using one of the following methods. ■ For the Solaris installation program, you can customize the disk layout in cylinder mode and manually assign the swap slice to the desired location. ■ For the custom JumpStart installation program, you can assign the swap slice in the profile file. For more information about the JumpStart profile file, see "Creating a Profile" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*. For an overview of the swap space, see Chapter 20, "Configuring Additional Swap Space (Tasks)," in *System Administration Guide: Devices and File Systems*.
A server that is providing home directory file systems	By default, home directories are usually located in the `/export` file system.
The Solaris software group you are installing	A software group is a grouping of software packages. When you are planning disk space, remember that you can add or remove individual software packages from the software group that you select. For information about software groups, see "Disk Space Recommendations for Software Groups" on page 29.
Upgrade	■ If you are using Solaris Live Upgrade to upgrade an inactive boot environment and want information about disk space planning, see "Solaris Live Upgrade Disk Space Requirements" in *Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning*. ■ If you are using other Solaris installation methods to plan disk space, see "Upgrading With Disk Space Reallocation" on page 33.
Language support	For example, Chinese, Japanese, or Korean.
Printing or mail support	Allocate additional space.
Additional software or third-party software	Allocate additional space.

Disk Space Recommendations for Software Groups

The Solaris software groups are collections of Solaris packages. Each software group includes support for different functions and hardware drivers.

- For an initial installation, you select the software group to install, based on the functions that you want to perform on the system.
- For an upgrade, you must upgrade to a software group that is installed on the system. For example, if you previously installed the End User Solaris Software Group on your system, you cannot use the upgrade option to upgrade to the Developer Solaris Software Group. However, during the upgrade you can add software to the system that is not part of the currently installed software group.

When you are installing the Solaris software, you can choose to add or remove packages from the Solaris software group that you selected. When you are selecting which packages to add or remove, you need to know about software dependencies and how the Solaris software is packaged.

The following figure shows the grouping of software packages. Reduced Network Support contains the minimal number of packages and Entire Solaris Software Group Plus OEM Support contains all the packages.

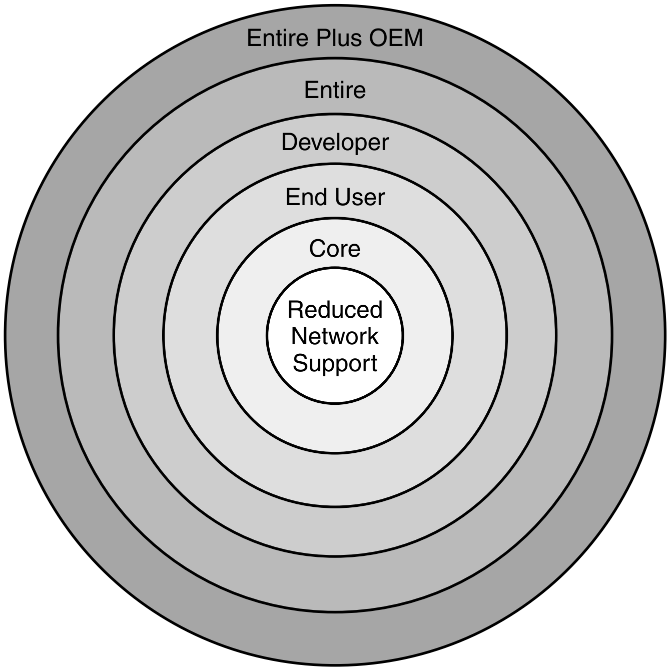

FIGURE 2–1 Solaris Software Groups

Table 2–4 lists the Solaris software groups and the recommended amount of disk space that you need to install each group.

Note – The disk space recommendations in Table 2–4 include space for the following items.

- Swap space
- Patches
- Additional software packages

You might find that the software groups require less disk space than the amount that is listed in this table.

TABLE 2–4 Disk Space Recommendations for Software Groups

Software Group	Description	Recommended Disk Space
Entire Solaris Software Group Plus OEM Support	Contains the packages for the Entire Solaris Software Group plus additional hardware drivers, including drivers for hardware that is not on the system at the time of installation.	6.7 Gbytes
Entire Solaris Software Group	Contains the packages for the Developer Solaris Software Group and additional software that is needed for servers.	6.5 Gbytes
Developer Solaris Software Group	Contains the packages for the End User Solaris Software Group plus additional support for software development. The additional software development support includes libraries, include files, man pages, and programming tools. Compilers are not included.	6.0 Gbytes
End User Solaris Software Group	Contains the packages that provide the minimum code that is required to boot and run a networked Solaris system and the Common Desktop Environment.	5.0 Gbytes
Core System Support Software Group	Contains the packages that provide the minimum code that is required to boot and run a networked Solaris system.	2.0 Gbytes
Reduced Network Support Software Group	Contains the packages that provide the minimum code that is required to boot and run a Solaris system with limited network service support. The Reduced Network Support Software Group provides a multiuser text-based console and system administration utilities. This software group also enables the system to recognize network interfaces, but does not activate network services.	2.0 Gbytes

Upgrade

You can upgrade a system by using one of three different upgrade methods: Solaris Live Upgrade, the Solaris installation program, and custom JumpStart.

TABLE 2–5 Solaris Upgrade Methods

Current Solaris OS	Solaris Upgrade Methods
Solaris 7, Solaris 8, Solaris 9	■ Solaris Live Upgrade – Upgrades a system by creating and upgrading a copy of the running system ■ The Solaris installation program – Provides an interactive upgrade with a graphical user interface or command-line interface ■ Custom JumpStart method – Provides an automated upgrade

Upgrade Limitations

Issue	Description
Upgrading to a different software group	You cannot upgrade your system to a software group that is not installed on the system. For example, if you previously installed the End User Solaris Software Group on your system, you cannot use the upgrade option to upgrade to the Developer Solaris Software Group. However, during the upgrade you can add software to the system that is not part of the currently installed software group.
Using the Solaris installation program to upgrade from a CD or DVD	You must have a free slice on the disk that does not store files and can be overwritten by the installation software. The `swap` slice is preferred, but you can use any slice that is not located in any of the "upgradable" root slices that are listed in `/etc/vfstab`. The size of this slice must be at least 512 Mbytes.

Upgrade Programs

You can perform a standard interactive upgrade with the Solaris installation program or an unattended upgrade with the custom JumpStart installation method. Solaris Live Upgrade enables you to upgrade a running system.

Upgrade Program	Description	For More Information
Solaris Live Upgrade	Enables you to create a copy of the currently running system. The copy can be upgraded and then a reboot switches the upgraded copy to become the currently running system. Using Solaris Live Upgrade reduces the downtime that is required to upgrade the Solaris OS. Also, Solaris Live Upgrade can prevent problems with upgrading. An example is the inability to recover from an upgrade if the power fails, because the copy being upgraded is not the currently running system.	To plan for disk space allocation when using Solaris Live Upgrade, see "Solaris Live Upgrade Requirements" in *Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning*.
Solaris installation program	Guides you through an upgrade with a GUI.	Chapter 2, "Installing With the Solaris Installation Program (Tasks)," in *Solaris 10 Installation Guide: Basic Installations*.
Custom JumpStart program	Provides an automated upgrade. A profile file and optional preinstallation and postinstallation scripts provide the information required. When creating a custom JumpStart profile for an upgrade, specify `install_type upgrade`. You must test the custom JumpStart profile against the system's disk configuration and currently installed software before you upgrade. Use the `pfinstall -D` command on the system that you are upgrading to test the profile. You cannot test an upgrade profile by using a disk configuration file.	■ For more information about testing the upgrade option, refer to "Testing a Profile" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*. ■ For more information about creating a upgrade profile, see "Profile Examples" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations* ■ For more information about performing an upgrade, see "Performing a Custom JumpStart Installation" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*

Installing a Solaris Flash Archive Instead of Upgrading

The Solaris Flash installation feature provides a method of creating a copy of the whole installation from a master system that can be replicated on many clone systems. This copy is called a Solaris Flash archive. You can install an archive by using any installation program. For information about installing an archive, see the following table.

Solaris Live Upgrade	“Installing Solaris Flash Archives on a Boot Environment” in *Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning*
Custom JumpStart	“To Prepare to Install a Solaris Flash Archive With a Custom JumpStart Installation” in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*
Solaris installation program	Chapter 4, “Installing and Administering Solaris Flash Archives (Tasks),” in *Solaris 10 Installation Guide: Solaris Flash Archives (Creation and Installation)*
WAN boot installation method	Chapter 12

Upgrading With Disk Space Reallocation

The upgrade option in the Solaris installation program and the `upgrade` keyword in the custom JumpStart program provide the ability to reallocate disk space. This reallocation automatically changes the sizes of the disk slices. You can reallocate disk space if the current file systems do not have enough space for the upgrade. For example, file systems might need more space for the upgrade for the following reasons:

- The Solaris software group that is currently installed on the system contains new software in the new release. Any new software that is included in a software group is automatically selected to be installed during the upgrade.
- The size of the existing software on the system has increased in the new release.

The auto-layout feature attempts to reallocate the disk space to accommodate the new size requirements of the file system. Initially, auto-layout attempts to reallocate space, based on a set of default constraints. If auto-layout cannot reallocate space, you must change the constraints on the file systems.

Note – Auto-layout does not have the ability to “grow” file systems. Auto-layout reallocates space by the following process:

1. Backing up required files on the file systems that need to change.
2. Repartitioning the disks on the basis of the file system changes.
3. Restoring the backup files before the upgrade happens.

- If you are using the Solaris installation program, and auto-layout cannot determine how to reallocate the disk space, you must use the custom JumpStart program to upgrade.
- If you are using the custom JumpStart method to upgrade and you create an upgrade profile, disk space might be a concern. If the current file systems do not contain enough disk space for the upgrade, you can use the `backup_media` and

layout_constraint keywords to reallocate disk space. For an example of how to use the backup_media and layout_constraint keywords in a profile, refer to “Profile Examples” in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

Using the Solaris Installation Program to Upgrade From DVD or CD Media

When you use the Solaris installation program from a DVD or CD to upgrade, you must have a slice on the disk that does not store files. The swap slice is preferred, but you can use any slice that is not located in any of the upgradable root slices that are listed in the /etc/vfstab. The size of this slice must be at least 512 Mbytes.

Backing Up Systems Before Upgrading

Backing up your existing file systems before you upgrade to the Solaris OS is highly recommended. If you copy file systems to removable media, such as tape, you can safeguard against data loss, damage, or corruption. For detailed instructions to back up your system, refer to Chapter 23, “Backing Up and Restoring File Systems (Overview),” in *System Administration Guide: Devices and File Systems*.

How to Find the Version of the Solaris OS That Your System Is Running

To see the version of Solaris software that is running on your system, type either of the following commands.

```
$ uname -a
```

The cat command provides more detailed information.

```
$ cat /etc/release
```

Locale Values

As a part of your installation, you can preconfigure the locale that you want the system to use. A *locale* determines how online information is displayed in a specific language and specific region. A language might also include more than one locale to accommodate regional differences, such as differences in the format of date and time, numeric and monetary conventions, and spelling.

You can preconfigure the system locale in a custom JumpStart profile or in the `sysidcfg` file.

Setting the locale in a profile	"Creating a Profile" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*
Setting the locale in the `sysidcfg` file	"Preconfiguring With the `sysidcfg` File" on page 57
List of locale values	*International Language Environments Guide*

Platform Names and Groups

When you are adding clients for a network installation, you must know your system architecture (platform group). If you are writing a custom JumpStart installation rules file, you need to know the platform name.

Some examples of platform names and groups follow. For a full list of SPARC based systems, see *Solaris 10 Sun Hardware Platform Guide* at `http://docs.sun.com/`.

TABLE 2–6 Example of Platform Names and Groups

System	Platform Name	Platform Group
Sun Blade™	SUNW,Sun-Blade-100	sun4u
x86 based	i86pc	i86pc

Note – On a running system, you can also use the `uname -i` command to determine a system's *platform name* or the `uname -m` command to determine a system's *platform group*.

Planning to Install and Configure Zones

The following introduction provides high-level planning information for global and non-global zones. For more specific planning information and specific procedures, see Chapter 16, "Introduction to Solaris Zones," in *System Administration Guide: Solaris Containers—Resource Management and Solaris Zones*.

After the Solaris OS is installed, you can install and configure zones. In a zones environment, the global zone is the single instance of the operating system that is running and is contained on every Solaris system. The global zone is both the default zone for the system and the zone that is used for system-wide administrative control. A non-global zone is a virtualized operating system environment.

Solaris Zones are a software partitioning technology used to virtualize operating system services and provide an isolated and secure environment for running applications. When you create a zone, you produce an application execution environment in which processes are isolated from all other zones. This isolation prevents processes that are running in one zone from monitoring or affecting processes that are running in any other zones. Even a process running in a non-global zone with superuser credentials cannot view or affect activity in any other zones. A process running in the global zone with superuser credentials can affect any process in any zone.

The global zone is the only zone from which a non-global zone can be configured, installed, managed, or uninstalled. Only the global zone is bootable from the system hardware. Administration of the system infrastructure, such as physical devices, routing, or dynamic reconfiguration (DR), is only possible in the global zone. Appropriately privileged processes running in the global zone can access objects associated with any or all other zones.

When installing the OS, the software group installed in the global zone is the set of packages that is shared by all the non-global zones. For example, if you install the Entire software group, all zones contain these packages. By default any additional packages installed in the global zone also populate the non-global zones. You can segregate into non-global zones applications, namespaces, servers, and network connections such as NFS and DHCP as well as other software. Each non-global zone is not aware of nor does it contain the segregated software in other non-global zones and each can operate independently. For example, you might have installed the Entire software group on the global zone and have running on separate non-global zones the Java Enterprise System Messaging Server, a database, DHCP, and a web server. When installing non-global zones keep in mind performance requirements of the applications running in each non-global zone.

Disk Space Requirements for Non-Global Zones

When installing the global zone, be sure to reserve enough disk space to house all of the zones you might create. Each non-global zone might have unique disk space requirements. The following description is a brief overview of planning information. For complete planning requirements and recommendations, see Chapter 18, "Planning and Configuring Non-Global Zones (Tasks)," in *System Administration Guide: Solaris Containers—Resource Management and Solaris Zones*.

No limits are placed on how much disk space can be consumed by a zone. The global zone administrator is responsible for space restriction. Even a small uniprocessor system can support a number of zones running simultaneously.

The nature of the packages installed in the global zone affects the space requirements of the non-global zones that are created. The number of packages and space requirements are factors. The following are general disk space guidelines.

- Approximately 100 Mbytes of free disk space is suggested when the global zone has been installed with all of the standard Solaris packages. Increase this amount if additional packages are installed in the global zone. By default, any additional packages installed in the global zone also populate the non-global zones. The directory location in the non-global zone for these additional packages is specified through the `inherit-pkg-dir` resource.
- Add 40 Mbytes of RAM per zone if the system has sufficient swap space. This addition is recommended to make each zone operational. When planning your system size, consider this addition of RAM.

Restricting Non-Global Zone Size

The following options can be used to restrict zone size.

- You can place the zone on a `lofi`-mounted partition. This action limits the amount of space consumed by the zone to that of the file used by `lofi`. For more information, see the `lofiadm`(1M) and `lofi`(7D) man pages.
- You can use soft partitions to divide disk slices or logical volumes into partitions. You can use these partitions as zone roots, and thus limit per-zone disk consumption. The soft partition limit is 8192 partitions. For more information, see Chapter 12, "Soft Partitions (Overview)," in *Solaris Volume Manager Administration Guide*.
- You can use the standard partitions of a disk for zone roots, and thus limit per-zone disk consumption.

SPARC: 64–bit Packaging Changes

In previous Solaris releases, the Solaris OS was delivered in separate packages for 32-bit and 64-bit components. In the Solaris 10 OS, packaging has been simplified with the delivery of most 32-bit and 64-bit components in a single package. The combined packages retain the names of the original 32-bit packages, and the 64-bit packages are no longer delivered. This change reduces the number of packages and simplifies installation.

The 64-bit packages are renamed with the following conventions:

- If a 64-bit package has a 32-bit counterpart, the 64-bit package is named with the 32-bit package name. For example, a 64-bit library such as /usr/lib/sparcv9/libc.so.1 previously would have been delivered in SUNWcslx, but now is delivered in SUNWcsl. The 64-bit SUNWcslx package is no longer delivered.
- If a package does not have a 32-bit counterpart, the "x" suffix is removed from the name. For example, SUNW1394x becomes SUNW1394. This change means that you might need to modify your custom JumpStart script or other package installation scripts to remove references to the 64-bit packages.

x86: Partitioning Recommendations

When using the Solaris OS on x86 based systems, follow these guidelines for partitioning your system.

The Solaris installation program uses a default boot-disk partition layout. These partitions are called `fdisk` partitions. An fdisk partition is a logical partition of a disk drive that is dedicated to a particular operating system on x86 based systems. To install the Solaris software, you must set up at least one Solaris `fdisk` partition on an x86 based system. x86 based systems allow up to four different `fdisk` partitions on a disk. These partitions can be used to hold individual operating systems. Each operating system must be located on a unique `fdisk` partition. A system can only have one Solaris `fdisk` partition per disk.

TABLE 2–7 x86: Default Partitions

Partitions	Partition Name	Partition Size
First partition (on some systems)	Diagnostic or service partition	Existing size on system
Second partition	x86 boot partition	Greater than 10 Mbytes, depending on disk size
Third partition	Solaris OS partition	Remaining space on the boot disk

Default Boot-Disk Partition Layout Preserves the Service Partition

The Solaris installation program uses a default boot-disk partition layout to accommodate the diagnostic or service partition. If your system currently includes a diagnostic or service partition, the default boot-disk partition layout enables you to preserve this partition.

Note – If you install the Solaris OS on an x86 based system that does not currently include a diagnostic or service partition, the installation program does not create a new diagnostic or Service partition by default. If you want to create a diagnostic or service partition on your system, see your hardware documentation.

Dual Booting With Solaris and Linux

`Problem With Dual Booting With Solaris and Linux`

Cause: If you are using the Linux system, the Solaris `fdisk` partition and the Linux `swap` partition use the same identifier, 0x82.

Solution: To resolve the problem, you can do one of the following:

- Choose not to use a Linux `swap` partition, provided that you have enough memory.
- Put the Linux `swap` partition on another drive.
- Back up the Linux data you want to keep to storage media, install the Solaris OS, and *then* reinstall Linux.

Caution – You might decide to install Linux after the Solaris OS. When the Linux installation program asks if you want to format the Linux `swap` partition (actually the Solaris `fdisk` partition) as a `swap` file, reply no.

CHAPTER 3

Gathering Information Before Installation or Upgrade (Planning)

This chapter contains checklists to help you gather all of the information that you need to install or upgrade your system.

- "Checklist for Installation" on page 41
- "Checklist for Upgrading" on page 47

Checklist for Installation

Use the following checklist to gather the information that you need to install the Solaris OS. You do not need to gather all of the information that is requested on the worksheet. You only need to collect the information that applies to your system.

TABLE 3–1 Installation Checklist

Information for Installation		Description or Example	Answer — Defaults are noted with an asterisk (*)
Network connection		Is the system connected to a network?	Networked/Non-networked*
DHCP		Can the system use Dynamic Host Configuration Protocol (DHCP) to configure its network interfaces?	Yes/No*
If you are not using DHCP, note the network address.	IP Address	If you are not using DHCP, supply the IP address for the system. Example: 172.31.255.255 To find this information on a running system, type the following command. `# ypmatch host-name hosts`	

TABLE 3–1 Installation Checklist *(Continued)*

Information for Installation		Description or Example	Answer — Defaults are noted with an asterisk (*)
	Subnet	If you are not using DHCP, is the system part of a subnet? If yes, what is the netmask of the subnet? Example: 255.255.255.0 To find this information on a running system, type the following command. `# more /etc/netmasks`	255.255.255.0*
	IPv6	Do you want to enable IPv6 on this machine?	Yes/No*
Host Name		Host name that you choose for the system. To find this information on a running system, type the following command. `# uname -n`	
Kerberos		Do you want to configure Kerberos security on this machine? If yes, gather this information: Default Realm: Administration Server: First KDC: (Optional) Additional KDCs:	Yes/No*
If the system uses a name service, provide the following information.	Name Service	Which name service should this system use? To find this information on a running system, type the following command. `# cat /etc/nsswitch.conf`	NIS+/NIS/DNS/LDAP/None*
	Domain Name	Provide the name of the domain in which the system resides. To find this information on a running system, type the following command. `# domainname`	
	NIS+ and NIS	Do you want to specify a name server or let the installation program find one? If you want to specify a name server, provide the following information.	Specify One/Find One*

TABLE 3–1 Installation Checklist *(Continued)*

Information for Installation		Description or Example	Answer — Defaults are noted with an asterisk (*)

		Server's host name:
		To display the server's host name, type the following command.
		`# ypwhich`
		Server's IP Address:
		To display the server's IP address, type the following command.
		`# nismatch` *nameserver-name* `hosts.org_dir`	

	DNS	Provide IP addresses for the DNS server. You must enter at least one IP address, but you can enter up to three addresses. Server's IP Address: To display the server's IP address, type the following command. `# getent ipnodes dns` You can enter a list of domains to search when a DNS query is made. Search Domain: Search Domain: Search Domain:	
	LDAP	Provide the following information about your LDAP profile. Profile Name: Profile Server: If you specify a proxy credential level in your LDAP profile, gather this information. Proxy-bind distinguished name: Proxy-bind password:	

TABLE 3–1 Installation Checklist *(Continued)*

Information for Installation	Description or Example	Answer — Defaults are noted with an asterisk (*)
Default Route	Do you want to specify a default route IP address or let the Solaris installation program find one? The default route provides a bridge that forwards traffic between two physical networks. An IP address is a unique number that identifies each host on a network. You have the following choices: ■ You can specify the IP address. An `/etc/defaultrouter` file is created with the specified IP address. When the system is rebooted, the specified IP address becomes the default route. ■ You can let the Solaris installation program detect an IP address. However, the system must be on a subnet that has a router that advertises itself by using the ICMP router discovery protocol. If you are using the command-line interface, the software detects an IP address when the system is booted. ■ You can choose None if you do not have a router or do not want the software to detect an IP address at this time. The software automatically tries to detect an IP address on reboot.	Specify one/Detect one/None*
Time Zone	How do you want to specify your default time zone?	Geographic region* Offset from GMT Time zone file
Root Password	Provide the root password for the system.	
Locales	For which geographic regions do you want to install support?	
SPARC: Power Management (only available on SPARC systems that support Power Management)	Do you want to use Power Management? **Note –** If your system has Energy Star version 3 or later, you are not prompted for this information.	Yes*/No
Proxy Server Configuration	Do you have a direct connection to the Internet or do you need to use a proxy server to gain access to the Internet? If you use a proxy server, provide the following information. Host: Port:	Direct connection*/Proxy server

TABLE 3–1 Installation Checklist *(Continued)*

Information for Installation	Description or Example	Answer — Defaults are noted with an asterisk (*)
Automatic reboot or CD/DVD ejection	Reboot automatically after software installation? Eject CD/DVD automatically after software installation?	Yes*/No Yes*/No
Default or Custom Install	Do you want to perform a default installation, or customize the installation? ■ Select Default installation to format the entire hard disk and install a preselected set of software, including Sun Java Enterprise System. For information about Sun Java Enterprise System software configuration, see *Sun Java Enterprise System Technical Overview* on `http://docs.sun.com`. ■ Select Custom installation to modify the hard disk layout and select the software that you want to install. **Note –** The text installer does not prompt you to select a Default or Custom Installation. To perform a default installation, accept the default values that are provided in the text installer. To perform a custom installation, edit the values in the text installer screens.	Default installation*/Custom installation
Software Group	Which Solaris Software Group do you want to install?	Entire Plus OEM Entire* Developer End User Core Reduced Networking
Custom Package Selection	Do you want to add or remove software packages from the Solaris Software Group that you install? **Note –** When you select which packages to add or remove, you need to know about software dependencies and how Solaris software is packaged.	
Select Disks	On which disks do you want to install the Solaris software? Example: `c0t0d0`	

TABLE 3–1 Installation Checklist *(Continued)*

Information for Installation	Description or Example	Answer — Defaults are noted with an asterisk (*)
x86: `fdisk` partitioning	Do you want to create, delete, or modify a Solaris `fdisk` partition? Each disk that is selected for file system layout must have a Solaris `fdisk` partition. Only one x86 Boot partition is allowed per system. If your system currently has a service partition, the Solaris installation program preserves the service partition by default. If you do not want to preserve the service partition, you must customize the `fdisk` partitions. For more information about preserving a service partition, see "Default Boot-Disk Partition Layout Preserves the Service Partition" on page 39.	
	Select Disks for `fdisk` Partition Customization?	Yes/No*
	Customize `fdisk` partitions?	Yes/No*
Preserve Data	Do you want to preserve any data that exists on the disks where you are installing the Solaris software?	Yes/No*
Auto-layout File Systems	Do you want the installation program to automatically lay out file systems on your disks? If yes, which file systems should be used for auto-layout? Example: `/, /opt, /var` If no, you must provide file system configuration information.	Yes*/No
Mount Remote File Systems	Does this system need to access software on another file system? If yes, provide the following information about the remote file system.	Yes/No*
	Server:	
	IP Address:	
	Remote File System:	
	Local Mount Point:	

TABLE 3–1 Installation Checklist *(Continued)*

Information for Installation	Description or Example	Answer — Defaults are noted with an asterisk (*)
If you are installing through a `tip` line, follow these instructions.	Ensure that your window display is at least 80 columns wide and 24 rows long. For more information, see `tip`(1). To determine the current dimensions of your `tip` window, use the `stty` command. For more information, see the man page, `stty`(1).	
Check your Ethernet connection.	If the system is part of a network, verify that an Ethernet connector or similar network adapter is connected to your system.	
Review the planning chapter and other relevant documentation.	■ Review the entire planning chapter or specific sections in Chapter 2. ■ Review the *Solaris 10 Release Notes* on `http://docs.sun.com` and vendor release notes to ensure that the software you use is supported in the new Solaris release. ■ Review the *Solaris 10 Sun Hardware Platform Guide* to ensure that your hardware is supported. ■ Review the documentation that accompanied your system to ensure that your system and devices are supported by the Solaris release.	

Checklist for Upgrading

Use the following checklist to gather the information that you need to upgrade the Solaris OS. You do not need to gather all of the information that is requested on the checklist. You only need to collect the information that applies to your system. If you are performing the upgrade over the network, the installation program provides the information for you, based on the current system configuration.

You cannot change basic system identification, such as host name or IP address. The installation program might prompt you for basic system identification, but you must enter the original values. If you use the Solaris installation program to upgrade, the upgrade fails if you attempt to change any of the values.

TABLE 3–2 Upgrade Checklist

Information for Upgrade		Description or Example	Answer – Defaults are noted with an asterisk (*)
Network connection		Is the system connected to a network?	Networked/Nonnetworked*
DHCP		Can the system use Dynamic Host Configuration Protocol (DHCP) to configure its network interfaces?	Yes/No*
If you are not using DHCP, note the network address.	IP Address	If you are not using DHCP, supply the IP address for the system. Example: 172.31.255.255 To find this information on a running system, type the following command. `# ypmatch host-name hosts`	
	Subnet	If you are not using DHCP, is the system part of a subnet? If yes, what is the netmask of the subnet? Example: 255.255.255.0 To find this information on a running system, type the following command. `# more /etc/netmasks`	255.255.255.0*
	IPv6	Do you want to enable IPv6 on this machine?	Yes/No*
Host Name		Host name that you choose for the system. To find this information on a running system, type the following command. `# uname -n`	
Kerberos		Do you want to configure Kerberos security on this machine? If yes, gather this information: Default Realm: Administration Server: First KDC: (Optional) Additional KDCs:	Yes/No*

TABLE 3–2 Upgrade Checklist *(Continued)*

Information for Upgrade		Description or Example	Answer – Defaults are noted with an asterisk (*)
If the system uses a name service, provide the following information.	Name Service	Which name service should this system use? To find this information on a running system, type the following command. # **`cat /etc/nsswitch.conf`**	NIS+/NIS/DNS/ LDAP/None*
	Domain Name	Provide the name of the domain in which the system resides. To find this information on a running system, type the following command. # **`domainname`**	
	NIS+ and NIS	Do you want to specify a name server or let the installation program find one? If you want to specify a name server, provide the following information. Server's host name: To display the server's host name, type the following command. # **`ypwhich`** Server's IP Address: To display the server's IP address, type the following command. # **`nismatch`** *nameserver-name* **`hosts.org_dir`**	Specify one/Find one*
	DNS	Provide IP addresses for the DNS server. You must enter at least one IP address, but you can enter up to three addresses. Server's IP Address: To display the server's IP address, type the following command. # **`getent ipnodes dns`** You can enter a list of domains to search when a DNS query is made.	

TABLE 3–2 Upgrade Checklist *(Continued)*

Information for Upgrade		Description or Example	Answer – Defaults are noted with an asterisk (*)
		Search Domain: Search Domain: Search Domain:	
	LDAP	Provide the following information about your LDAP profile. Profile Name: Profile Server: If you specify a proxy credential level in your LDAP profile, gather this information. Proxy-bind distinguished name: Proxy-bind password:	
Default Route		Do you want to specify a default route IP address or let the Solaris installation program find one? The default route provides a bridge that forwards traffic between two physical networks. An IP address is a unique number that identifies each host on a network. You have the following choices: ■ You can specify the IP address. An `/etc/defaultrouter` file is created with the specified IP address. When the system is rebooted, the specified IP address becomes the default route. ■ You can let the Solaris installation program detect an IP address. However, the system must be on a subnet that has a router that advertises itself by using the ICMP router discovery protocol. If you are using the command-line interface, the software detects an IP address when the system is booted. ■ You can choose None if you do not have a router or do not want the software to detect an IP address at this time. The software automatically tries to detect an IP address on reboot.	Specify one/Detect one/None*

TABLE 3–2 Upgrade Checklist *(Continued)*

Information for Upgrade	Description or Example	Answer – Defaults are noted with an asterisk (*)
Time Zone	How do you want to specify your default time zone?	Geographic region* Offset from GMT Time zone file
Root Password	Provide the root password for the system.	
Default or Custom Install	Do you want to perform a default installation, or customize the installation? ■ Select Default installation to format the entire hard disk and install a preselected set of software, including Sun Java Enterprise System. For information about Sun Java Enterprise System software configuration, see *Sun Java Enterprise System Technical Overview* on `http://docs.sun.com`. ■ Select Custom installation to modify the hard disk layout and select the software that you want to install. **Note –** The text installer does not prompt you to select a Default or Custom Installation. To perform a default installation, accept the default values that are provided in the text installer. To perform a custom installation, edit the values in the text installer screens.	Default installation*/Custom installation
Locales	For which geographic regions do you want to install support?	
SPARC: Power Management (only available on SPARC systems that support Power Management)	Do you want to use Power Management? **Note –** If your system has Energy Star version 3 or later, you are not prompted for this information.	Yes/No
Proxy Server Configuration	Do you have a direct connection to the Internet or do you need to use a proxy server to gain access to the Internet? If you use a proxy server, provide the following information. Host: Port:	Direct Connection*/Proxy Server

TABLE 3–2 Upgrade Checklist *(Continued)*

Information for Upgrade	Description or Example	Answer – Defaults are noted with an asterisk (*)
Automatic reboot or CD/DVD ejection	Reboot automatically after software installation? Eject CD/DVD automatically after software installation?	Yes*/No Yes*/No
Disk space reallocation	Do you want the installation program to automatically re-layout the systems on your disks? If yes, which file system should be used for auto-layout? Example: /, /opt, /var If no, you must provide information for the system configuration.	Yes/No*
If you are installing through a `tip` line, follow these instructions.	Ensure that your window display is at least 80 columns wide and 24 rows long. For more information, see `tip`(1). To determine the current dimensions of your `tip` window, use the `stty` command. For more information, see the man page, `stty`(1).	
Check your Ethernet connection.	If the system is part of a network, verify that an Ethernet connector or similar network adapter is connected to your system.	
Solaris Live Upgrade use	■ Determine your resource requirements for creating a new boot environment and upgrading it. For detailed information, refer to Chapter 5, "Solaris Live Upgrade (Planning)," in *Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning*. ■ Determine the requirements if you are using RAID-1 volumes. For detailed information, refer to "Guidelines for Selecting Slices for File Systems" in *Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning*.	
Check the system for the existence of Prestoserve software.	If you begin the upgrade process by shutting down the system with the `init 0` command and you're using Prestoserve software, you might lose data. Refer to the Prestoserve documentation for shutdown instructions.	

TABLE 3–2 Upgrade Checklist *(Continued)*

Information for Upgrade	Description or Example	Answer – Defaults are noted with an asterisk (*)
Check for patches needed.	The most recent patch list is provided at http://sunsolve.sun.com.	
Review the planning chapter and other relevant documentation.	■ Review the entire planning chapter or specific sections in Chapter 2. ■ Review the *Solaris 10 Release Notes* on `http://docs.sun.com` and vendor release notes to ensure that the software you use is supported in the new Solaris release. ■ Review the *Solaris 10 Sun Hardware Platform Guide* to ensure that your hardware is supported. ■ Review the documentation that accompanied your system to ensure that your system and devices are supported by the Solaris release.	

CHAPTER 4

Preconfiguring System Configuration Information (Tasks)

This chapter describes how to preconfigure system information. Preconfiguration can help you to avoid being prompted for this information when you install the Solaris OS. This chapter also describes how to preconfigure Power Management™ information. This chapter contains the following sections:

Advantages of Preconfiguring System Configuration Information

The installation methods require configuration information about a system, such as peripheral devices, host name, Internet Protocol (IP) address, and name service. Before the installation tools prompt you for configuration information, they check for the information in the `sysidcfg` file and then in the name service databases.

When the Solaris installation program or the custom JumpStart installation program detects preconfigured system information, the installation program does not prompt you to enter the information. For example, you have several systems and you do not want a time zone prompt every time you install the Solaris 10 software on one of the systems. You can specify the time zone in the `sysidcfg` file or the name service databases. When you install the Solaris 10 software, the installation program does not prompt you to type a time zone.

Ways to Preconfigure System Configuration Information

You can choose one of the following ways to preconfigure system configuration information. You can add the system configuration information to either of the following.

- A sysidcfg file on a remote system or diskette
- The name service database available at your site

If your site uses DHCP, you can also preconfigure some system information in the site DHCP server. For more information about how to use a DHCP server to preconfigure system information, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.

Use the following table to determine whether to use a sysidcfg file or a name service database to preconfigure system configuration information.

TABLE 4–1 Methods to Preconfigure System Configuration Information

Preconfigurable System Information	Preconfigurable With the sysidcfg File?	Preconfigurable With the Name Service?
Name service	Yes	Yes
Domain name	Yes	No
Name server	Yes	No
Network interface	Yes	No
Host name	Yes Because this information is system specific, edit the name service rather than create a different sysidcfg file for each system.	Yes
Internet Protocol (IP) address	Yes Because this information is system specific, edit the name service rather than create a different sysidcfg file for each system.	Yes
Netmask	Yes	No
DHCP	Yes	No

TABLE 4–1 Methods to Preconfigure System Configuration Information *(Continued)*

Preconfigurable System Information	Preconfigurable With the `sysidcfg` File?	Preconfigurable With the Name Service?
IPv6	Yes	No
Default route	Yes	No
Root password	Yes	No
Security policy	Yes	No
Language (locale) in which to display the install program and desktop	Yes	Yes, if NIS or NIS+ No, if DNS or LDAP
Terminal type	Yes	No
Time zone	Yes	Yes
Date and time	Yes	Yes
Web proxy	No You can configure this information with the Solaris installation program, but not through the `sysidcfg` file or the name service.	No
x86: Monitor type	Yes	No
x86: Keyboard language, keyboard layout	Yes	No
x86: Graphics card, color depth, display resolution, screen size	Yes	No
x86: Pointing device, number of buttons, IRQ level	Yes	No
SPARC: Power Management (autoshutdown) You cannot preconfigure Power Management through the `sysidcfg` file or the name service. "SPARC: Preconfiguring Power Management Information" on page 88 contains details.	No	No

Preconfiguring With the `sysidcfg` File

You can specify a set of keywords in the `sysidcfg` file to preconfigure a system. The keywords are described in "`sysidcfg` File Keywords" on page 59.

You must create a unique sysidcfg file for every system that requires different configuration information. You can use the same sysidcfg file to preconfigure the time zone on a set of systems if you want all the systems to be assigned the same time zone. However, if you want to preconfigure a different root (superuser) password for each of those systems, you need to create a unique sysidcfg file for each system.

You can place the sysidcfg file in one of the following.

- NFS file system – If you put the sysidcfg file in a shared NFS file system, you must use the -p option of the add_install_client(1M) command when you set up the system to install from the network. The -p option specifies where the system can find the sysidcfg file when you install the Solaris 10 software.
- UFS or PCFS diskette – Place the sysidcfg file in the root (/) directory on the diskette.
- HTTP or HTTPS server – If you want to perform a WAN boot installation, place the sysidcfg file in the document root directory of the web server.

Note – If you are performing a custom JumpStart installation and you want to use a sysidcfg file on a diskette, you must place the sysidcfg file on the profile diskette. To create a profile diskette, see “Creating a Profile Diskette for Standalone Systems” in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

You can place only one sysidcfg file in a directory or on a diskette. If you are creating more than one sysidcfg file, you must place each file in a different directory or on a different diskette.

Syntax Rules for the sysidcfg File

You can use two types of keywords in the sysidcfg file: independent and dependent. Dependent keywords are guaranteed to be unique only within independent keywords. A dependent keyword exists only when it is identified with its associated independent keyword.

In this example, name_service is the independent keyword, while domain_name and name_server are the dependent keywords:

```
name_service=NIS {domain_name=marquee.central.example.com
name_server=connor(192.168.112.3)}
```

Syntax Rule	Example

Independent keywords can be listed in any order.	`pointer=MS-S`
	`display=ati {size=15-inch}`

Keywords are not case sensitive.	`TIMEZONE=US/Central`
	`terminal=sun-cmd`

Enclose all dependent keywords in curly braces ({}) to tie them to their associated independent keyword.	`name_service=NIS`
	`{domain_name=marquee.central.example.com`
	`name_server=connor(192.168.112.3)}`

You can optionally enclosed values in single (‘) or double quotes (“).	`network_interface='none'`

For all keywords except the `network_interface` keyword, only one instance of a keyword is valid. However, if you specify the keyword more than once, only the first instance of the keyword is used.	`name_service=NIS`
	`name_service=DNS`

sysidcfg File Keywords

Table 4–2 lists the keywords you can use to configure system information in the sysidcfg file.

TABLE 4–2 Keywords You Can Use in sysidcfg

Configuration Information	Keyword
Name service, domain name, name server	“name_service Keyword” on page 60
Network interface, host name, Internet Protocol (IP) address, netmask, DHCP, IPv6	“network_interface Keyword” on page 63
Root password	“root_password Keyword” on page 68
Security policy	“security_policy Keyword” on page 69
Language in which to display the install program and desktop	“system_locale Keyword” on page 69
Terminal type	“terminal Keyword” on page 70
Time zone	“timezone Keyword” on page 70
Date and time	“timeserver Keyword” on page 70
x86: Monitor type	“x86: monitor Keyword” on page 71
x86: Keyboard language, keyboard layout	“x86: keyboard Keyword” on page 71

TABLE 4–2 Keywords You Can Use in `sysidcfg` *(Continued)*

Configuration Information	Keyword
x86: Graphics card, screen size, color depth, display resolution	"x86: `display` Keyword" on page 71
x86: Pointing device, number of buttons, IRQ level	"x86: `pointer` Keyword" on page 72

The following sections describe the keywords that you can use in the `sysidcfg` file.

`name_service` Keyword

You can use the `name_service` keyword to configure the name service, the domain name, and the name server for the system. The following sample shows the general syntax for the `name_service` keyword.

```
name_service=name-service {domain_name=domain-name
                           name_server=name-server
                           optional-keyword=value}
```

Choose only one value for `name_service`. Include all or none of the `domain_name`,`name_server`, or optional keywords, as needed. If no keywords are used, omit the curly braces {}.

The following sections describe the keyword syntax to configure the system to use a specific name service.

NIS Syntax for `name_service` Keyword

Use the following syntax to configure the system to use the NIS name service.

```
name_service=NIS {domain_name=domain-name
                   name_server=hostname(ip-address)}
```

domain-name Specifies the domain name

hostname Specifies the host name of the name server

ip-address Specifies the IP address of the name server

EXAMPLE 4–1 Specifying a NIS Server With the `name_service` Keyword

The following example specifies a NIS server with the domain name `west.example.com`. The server's host name is `timber`, and the server IP address is 192.168.2.1.

```
name_service=NIS {domain_name=west.example.com
                  name_server=timber(192.168.2.1)}
```

For more information about the NIS name service, see *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

NIS+ Syntax for `name_service` Keyword

Use the following syntax to configure the system to use the NIS name service.

```
name_service=NIS+ {domain_name=domain-name
                   name_server=hostname(ip-address)}
```

domain-name Specifies the domain name

hostname Specifies the host name of the name server

ip-address Specifies the IP address of the name server

EXAMPLE 4–2 Specifying a NIS+ Server With the `name_service` Keyword

The following example specifies a NIS+ server with the domain name `west.example.com`. The server's host name is `timber`, and the server IP address is 192.168.2.1.

```
name_service=NIS+ {domain_name=west.example.com
                   name_server=timber(192.168.2.1)}
```

For more information about the NIS+ name service, see *System Administration Guide: Naming and Directory Services (NIS+)*.

DNS Syntax for `name_service` Keyword

Use the following syntax to configure the system to use DNS.

```
name_service=DNS {domain_name=domain-name
                  name_server=ip-address,ip-address,ip-address
                  search=domain-name,domain-name,domain-name,
                  domain-name,domain-name,domain-name}
```

domain_name=*domain-name* Specifies the domain name.

name_server=*ip-address* Specifies the IP address of the DNS server. You can specify up to three IP addresses as values for the `name_server` keyword.

search=*domain-name* (Optional) Specifies additional domains to search for name service information. You can specify up to six domain names to search. The total length of each search entry cannot exceed 250 characters.

EXAMPLE 4–3 Specifying a DNS Server With the `name_service` Keyword

The following example specifies a DNS server with the domain name `west.example.com`. The server IP addresses are 10.0.1.10 and 10.0.1.20. `example.com` and `east.example.com` are listed as additional domains to search for name service information.

EXAMPLE 4–3 Specifying a DNS Server With the `name_service` Keyword *(Continued)*

```
name_service=DNS {domain_name=west.example.com
                  name_server=10.0.1.10,10.0.1.20
                  search=example.com,east.example.com}
```

For more information about the DNS name service, see *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

LDAP Syntax for `name_service` Keyword

Use the following syntax to configure the system to use LDAP.

```
name_service=LDAP {domain_name=domain_name
                   profile=profile_name profile_server=ip_address
                   proxy_dn="proxy_bind_dn" proxy_password=password}
```

domain_name Specifies the domain name of the LDAP server.

profile_name Specifies the name of the LDAP profile you want to use to configure the system.

ip_address Specifies the IP address of the LDAP profile server.

proxy_bind_dn (Optional) Specifies the proxy bind distinguished name. You must enclose the *proxy_bind_dn* value in double quotes.

password (Optional) Specifies the client proxy password.

EXAMPLE 4–4 Specifying an LDAP Server With the `name_service` Keyword

The following example specifies an LDAP server with the following configuration information.

- The domain name is `west.example.com`.
- The installation program uses the LDAP profile that is named `default` to configure the system.
- The IP address of the LDAP server is 172.31.2.1.
- The proxy bind distinguished name includes the following information.
  - The common name for the entry is `proxyagent`.
  - The organizational unit is `profile`.
  - The proxy domain includes the `west`, `example`, and `com` domain components.
- The proxy password is `password`.

```
name_service=LDAP {domain_name=west.example.com
                   profile=default
                   profile_server=172.31.2.1
                   proxy_dn="cn=proxyagent,ou=profile,
                   dc=west,dc=example,dc=com"
                   proxy_password=password}
```

For more information about how to use LDAP, see *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

`network_interface` Keyword

Use the `network_interface` keyword to perform the following tasks.

- Specify a host name
- Specify an IP address
- Specify a netmask value
- Use DHCP to configure the network interface
- Enable IPv6 on the network interface

The following sections describe how to use the `network_interface` keyword to configure the system interfaces.

Syntax for Nonnetworked Systems

To turn off networking for the system, set the `network_interface` value to none. For example:

```
network_interface=none
```

Syntax for Configuring a Single Interface

You can use the `network_interface` keyword to configure a single interface in the following ways.

- **With DHCP** – You can use a DHCP server on your network to configure the network interface. For more information on how to use a DHCP server during your installation, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.

  To use the DHCP server to configure a single interface on the system, use the following syntax for the `network_interface` keyword.

  ```
  network_interface=PRIMARY or value
                    {dhcp protocol_ipv6=yes-or-no}
  ```

  `PRIMARY` Instructs the installation program to configure the first up, non-loopback interface that is found on the system. The order is the same as the order that is displayed with the `ifconfig` command. If no interfaces are up, then the first non-loopback interface is used. If no non-loopback interfaces are found, then the system is nonnetworked.

value — Instructs the installation program to configure a specific interface, such as `hme0` or `eri1`.

`protocol_ipv6=`*yes-or-no* — Instructs the installation program to configure the system to either use IPv6 or to not use IPv6.

For WAN boot installations, you must set the value of `protocol_ipv6=no`.

- **Without DHCP** – If you do not want to use DHCP to configure the network interface, you can specify the configuration information in the `sysidcfg` file. To instruct the installation program to configure a single interface on the system without using DHCP, use the following syntax.

```
network_interface=PRIMARY or value
                  {hostname=host_name
                   default_route=ip_address
                   ip_address=ip_address
                   netmask=netmask
                   protocol_ipv6=yes_or_no}
```

PRIMARY — Instructs the installation program to configure the first up, non-loopback interface that is found on the system. The order is the same as the order that is displayed with the `ifconfig` command. If no interfaces are up, then the first non-loopback interface is used. If no non-loopback interfaces are found, then the system is not networked.

Note – Do not use the `PRIMARY` keyword value if you want to configure multiple interfaces.

value — Instructs the installation program to configure a specific interface, such as `hme0` or `eri1`.

`hostname=`*host_name* — (Optional) Specifies the host name of the system.

`default_route=`*ip_address* or `NONE`

(Optional) Specifies the IP address of the default router. If you want the installation program to detect the router by using the ICMP router discovery protocol, omit this keyword.

Note – If the installation program cannot detect the router, you are prompted for the router information during the installation.

`ip_address=`*ip_address*

(Optional) Specifies the IP address of the system.

`netmask=`*netmask*

(Optional) Specifies the netmask value for the system.

`protocol_ipv6=`*yes_or_no*

(Optional) Instructs the installation program to configure the system to either use IPv6 or to not use IPv6.

Note – To perform an unattended custom JumpStart installation, you must specify a value for the `protocol_ipv6` keyword.

For WAN boot installations, you must set the value of `protocol_ipv6=no`.

Include any combination or none of the `hostname`, `ip_address`, and `netmask` keywords, as needed. If you do not use any of these keywords, omit the curly braces ({}).

EXAMPLE 4–5 Configuring a Single Interface By Using DHCP With the `network_interface` Keyword

The following example instructs the installation program to use DHCP to configure the `eri0` network interface. IPv6 support is not enabled.

```
network_interface=eri0 {dhcp protocol_ipv6=no}
```

EXAMPLE 4–6 Configuring a Single Interface By Specifying Configuration Information With the `network_interface` Keyword

The following example configures the interface `eri0` with the following settings.

- The host name is set to host1.
- The IP address is set to 172.31.88.100.
- The netmask is set to 255.255.255.0.
- IPv6 support is not enabled on the interface.

```
network_interface=eri0 {hostname=host1 ip_address=172.31.88.100
                        netmask=255.255.255.0 protocol_ipv6=no}
```

Syntax for Configuring Multiple Interfaces

You can configure multiple network interfaces in your `sysidcfg` file. For each interface that you want to configure, include a `network_interface` entry in the `sysidcfg` file.

You can use the `network_interface` keyword to configure multiple interfaces in the following ways.

- **With DHCP** – You can use a DHCP server on your network to configure a network interface. For more information on how to use a DHCP server during your installation, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.

  To use the DHCP server to configure a network interface on the system, use the following syntax for the `network_interface` keyword.

  ```
  network_interface=value {primary
                               dhcp protocol_ipv6=yes-or-no}
  ```

  value
  Instructs the installation program to configure a specific interface, such as `hme0` or `eri1`.

  `primary`
  (Optional) Specifies *value* as the primary interface.

  `protocol_ipv6=`*yes-or-no*
  Instructs the installation program to configure the system to either use IPv6 or to not use IPv6.

  > **Note –** For WAN boot installations, you must set the value of `protocol_ipv6=no`.

- **Without DHCP** – If you do not want to use DHCP to configure the network interface, you can specify the configuration information in the `sysidcfg` file. To instruct the installation program to configure multiple interfaces without using DHCP, use the following syntax.

```
network_interface=value {primary hostname=host_name
                          default_route=ip_address or NONE
                          ip_address=ip_address
                          netmask=netmask
                          protocol_ipv6=yes_or_no}
```

value

Instructs the installation program to configure a specific interface, such as `hme0` or `eri1`.

`primary`

(Optional) Specifies *value* as the primary interface.

`hostname=`*host_name*

(Optional) Specifies the host name of the system.

`default_route=`*ip_address* or NONE

(Optional) Specifies the IP address of the default router. If you want the installation program to detect the router by using the ICMP router discovery protocol, omit this keyword.

If you configure multiple interfaces in the `sysidcfg` file, set `default_route=NONE` for each secondary interface that does not use a static default route.

Note – If the installation program cannot detect the router, you are prompted for the router information during the installation.

`ip_address=`*ip_address*

(Optional) Specifies the IP address of the system.

`netmask=`*netmask*

(Optional) Specifies the netmask value for the system.

`protocol_ipv6=`*yes_or_no*

(Optional) Instructs the installation program to configure the system to either use IPv6 or to not use IPv6.

Note – To perform an unattended custom JumpStart installation, you must specify a value for the `protocol_ipv6` keyword.

For WAN boot installations, you must set the value of `protocol_ipv6=no`.

Include any combination or none of the `hostname`, `ip_address`, and `netmask` keywords, as needed. If you do not use any of these keywords, omit the curly braces ({}).

In the same `sysidcfg` file, you can use DHCP to configure certain interfaces, while also specifying the configuration information for other interfaces in the `sysidcfg` file.

EXAMPLE 4–7 Configuring Multiple Interfaces With the `network_interface` Keyword

In the following example, the network interfaces eri0 and eri1 are configured in the following way.

- `eri0` is configured by using the DHCP server. IPv6 support is not enabled on `eri0`.
- `eri1` is the primary network interface. The host name is set to host1, and the IP address is set to 172.31.88.100. The netmask is set to 255.255.255.0. IPv6 support is not enabled on `eri1`.

```
network_interface=eri0 {dhcp protocol_ipv6=no}
network_interface=eri1 {primary hostname=host1
                        ip_address=172.146.88.100
                        netmask=255.255.255.0
                        protocol_ipv6=no}
```

`root_password` Keyword

You can specify the root password to the system in the `sysidcfg` file. To specify the root password, use the `root_password` keyword with the following syntax.

`root_password=`*encrypted-password*

encrypted-password is the encrypted password as it appears in the `/etc/shadow` file.

security_policy Keyword

You can use the security_policy keyword in your sysidcfg file to configure your system to use the Kerberos network authentication protocol. If you want to configure the system to use Kerberos, use the following syntax.

```
security_policy=kerberos {default_realm=FQDN
                          admin_server=FQDN kdc=FQDN1, FQDN2, FQDN3}
```

FQDN specifies the fully qualified domain name of the Kerberos default realm, the administration server, or key distribution center (KDC). You must specify at least one, but no more than three, key distribution centers.

If you do not want to set the security policy for the system, set security_policy=NONE.

For more information about the Kerberos network authentication protocol, see *System Administration Guide: Security Services*.

EXAMPLE 4–8 Configuring the System to Use Kerberos With the security_policy Keyword

The following example configures the system to use Kerberos with the following information.

- The Kerberos default realm is example.COM.
- The Kerberos administration server is krbadmin.example.COM.
- The two key distribution centers are kdc1.example.COM and kdc2.example.COM.

```
security_policy=kerberos
                 {default_realm=example.COM
                  admin_server=krbadmin.example.COM
                  kdc=kdc1.example.COM,
                  kdc2.example.COM}
```

system_locale Keyword

You can use the system_locale keyword to specify the language in which to display the install program and desktop. Use the following syntax to specify a locale.

```
system_locale=locale
```

locale specifies the language that you want the system to use to display the installation panels and screens. For a list of valid locale values, see the /usr/lib/locale directory or *International Language Environments Guide*.

terminal Keyword

You can use the terminal keyword to specify the terminal type for the system. Use the following syntax to specify the terminal type.

```
terminal=terminal_type
```

terminal_type specifies the terminal type for the system. For a list of valid terminal values, see the subdirectories in the /usr/share/lib/terminfo directory.

timezone Keyword

You can set the time zone for the system with the timezone keyword. Use the following syntax.

```
timezone=timezone
```

In the previous example, *timezone* specifies the time zone value for the system. The directories and files in the /usr/share/lib/zoneinfo directory provide the valid time zone values. The *timezone* value is the name of the path relative to the /usr/share/lib/zoneinfo directory. You can also specify any valid Olson time zone.

EXAMPLE 4–9 Configuring the System Time Zone With the timezone Keyword

In the following example, the system time zone is set to mountain standard time in the United States.

```
timezone=US/Mountain
```

The installation program configures the system to use the time zone information in /usr/share/lib/zoneinfo/US/Mountain.

timeserver Keyword

You can use the timeserver keyword to specify the system that sets the date and time on the system you want to install.

Note – Do not set timeserver=*hostname* or *ip-address* if you are running a name service.

Choose one of the following methods to set the timeserver keyword.

- To configure the system to serve as its own time server, set `timeserver=localhost`. If you specify `localhost` as the time server, the system's time is assumed to be correct.
- To specify another system as the time server, specify either the host name or the IP address of the time server with the `timeserver` keyword. Use the following syntax.

  `timeserver=`*hostname* `or` *ip-address*

  hostname is the host name of the time server system. *ip-address* specifies the IP address of the time server.

x86: `monitor` Keyword

For x86 based systems, you can configure the monitor information with the `monitor` keyword. Use the following syntax with the `monitor` keyword.

`monitor=`*monitor_type*

To set the value for the `monitor` keyword, run the `kdmconfig -d` command on the system you want to install. Copy the line of output that includes the `monitor` keyword, and include this line in the `sysidcfg` file.

For more information, see the `kdmconfig`(1M)

x86: `keyboard` Keyword

For x86 based systems, you can configure the keyboard language and layout information with the `keyboard` keyword. Use the following syntax with the `keyboard` keyword.

`keyboard=`*keyboard_language* {`layout=`*value*}

To set the value for the `keyboard` keyword, run the `kdmconfig -d` command on the system you want to install. Copy the line of output that includes the `keyboard` keyword, and include this line in the `sysidcfg` file.

For more information, see the `kdmconfig`(1M)

x86: `display` Keyword

For x86 based systems, you can configure the following information with the `display` keyword.

- Graphics card
- Screen size

- Color depth
- Display resolution

Use the following syntax with the `display` keyword.

```
display=graphics_card {size=screen_size
                        depth=color_depth
                        resolution=screen_resolution}
```

To set the appropriate values for the `display` keyword, run the `kdmconfig -d` command on the system you want to install. Copy the line of output that includes the `display` keyword, and include this line in the `sysidcfg` file.

For more information, see the `kdmconfig`(1M)

x86: `pointer` Keyword

For x86 based systems, you can configure the following mouse information with the `pointer` keyword.

- Pointing device
- Number of buttons
- IRQ level

Use the following syntax with the `pointer` keyword.

```
pointer=pointing_device {nbuttons=number_buttons irq=value}
```

To set the value for the `pointer` keyword, run the `kdmconfig -d` command on the system you want to install. Copy the line of output that includes the `pointer` keyword, and include this line in the `sysidcfg` file.

For more information, see the `kdmconfig`(1M)

▼ To Create a `sysidcfg` Configuration File

Steps

1. **Create a file called `sysidcfg` in a text editor.**
2. **Type the `sysidcfg` keywords you want.**
3. **Save the `sysidcfg` file.**

Note – If you create more than one `sysidcfg` file, you must save each file in a separate directory or on a separate diskette.

4. **Make the `sysidcfg` file available to clients through the following:**
   - A shared NFS file system. Use `add_install_client`(1M) with the `-p` option to set up the system to install from the network.
   - The root (/) directory on a UFS diskette or PCFS diskette.

Example 4–10 SPARC: `sysidcfg` File

The following is an example of a `sysidcfg` file for a SPARC based system. The host name, IP address, and netmask of this system has been preconfigured by editing the name service. Because all of the system configuration information is preconfigured in this file, you can use a custom JumpStart profile to perform a custom JumpStart installation.

```
system_locale=en_US
timezone=US/Central
terminal=sun-cmd
timeserver=localhost
name_service=NIS {domain_name=marquee.central.example.com
                  name_server=nmsvr2(172.31.112.3)}
root_password=m4QPOWNY
network_interface=hme0 {hostname=host1
                       default_route=172.31.88.1
                       ip_address=172.31.88.210
                       netmask=255.255.0.0
                       protocol_ipv6=no}
security_policy=kerberos {default_realm=example.COM
                          admin_server=krbadmin.example.COM
                          kdc=kdc1.example.COM,
                          kdc2.example.COM}
```

Example 4–11 x86: `sysidcfg` File

The following sample `sysidcfg` file is for a group of x86 based systems that all use the same type of keyboard, graphics cards, and pointing devices. The device information (`keyboard`, `display`, and `pointer`) was obtained by running the `kdmconfig`(1M) command with the `-d` option. If the following example `sysidcfg` file is used, a prompt that asks you to select a language (`system_locale`) is displayed before installation can proceed.

```
keyboard=ATKBD {layout=US-English}
display=ati {size=15-inch}
pointer=MS-S
timezone=US/Central
timeserver=timehost1
terminal=ibm-pc
name_service=NIS {domain_name=marquee.central.example.com
                  name_server=nmsvr2(172.25.112.3)}
root_password=URFUni9
```

Example 4–12 sysidcfg File for Configuring Multiple Interfaces

In the following sample sysidcfg file, configuration information is specified for both the eri0 and eri1 network interfaces. The eri0 interface is configured as the primary network interface, and eri1 is configured as a secondary network interface.

```
timezone=US/Pacific
system_locale=C
terminal=xterms
timeserver=localhost
network_interface=eri0 {primary
                        hostname=host1
                        ip_address=192.168.2.7
                        netmask=255.255.255.0
                        protocol_ipv6=no
                        default_route=192.168.2.1}

network_interface=eri1 {hostname=host1-b
                        ip_address=192.168.3.8
                        netmask=255.255.255.0
                        protocol_ipv6=no
                        default_route=NONE}
root_password=JE2C35JGZi4B2
security_policy=none
name_service=NIS {domain_name=domain.example.com
                  name_server=nis-server(192.168.2.200)}
```

More Information: Continuing the Installation

If you plan to use the sysidcfg file in an installation over the network, you need to set up an installation server and add the system as an installation client. For more information, see Chapter 5.

If you plan to use the sysidcfg file in a WAN boot installation, you need to perform additional tasks. For more information, see Chapter 9.

If you plan to use the sysidcfg file in a custom JumpStart installation, you need to create a profile and a rules.ok file. For more information, see Chapter 3, "Custom JumpStart (Overview)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

See Also For more information about the sysidcfg file, see the man page sysidcfg(4).

Preconfiguring With the Name Service

The following table provides a high-level overview of the name service databases that you need to edit and populate to preconfigure system information.

System Information to Preconfigure	Name Service Database
Host name and Internet Protocol (IP) address	`hosts`
Date and time	`hosts`. Specify the `timehost` alias next to the host name of the system that will provide the date and time for the systems that are being installed.
Time zone	`timezone`
Netmask	`netmasks`

You cannot preconfigure the locale for a system with the DNS or LDAP name service. If you use the NIS or NIS+ name service, follow the procedure for your name service to preconfigure the locale for a system:

- "To Preconfigure the Locale Using NIS" on page 75
- "To Preconfigure the Locale Using NIS+" on page 77

▼ To Preconfigure the Locale Using NIS

Steps

1. **Become superuser on the name server.**

2. **Change `/var/yp/Makefile` to add the locale map.**

   a. **Insert this shell procedure after the last *variable*`.time` shell procedure.**

```
locale.time:  $(DIR)/locale
        -@if [ -f $(DIR)/locale ]; then \
                sed -e "/^#/d" -e s/#.*$$// $(DIR)/locale \
                | awk ’{for (i = 2; i<=NF; i++) print $$i, $$0}’ \
                | $(MAKEDBM) - $(YPDBDIR)/$(DOM)/locale.byname; \
                touch locale.time; \
                echo "updated locale"; \
                if [ ! $(NOPUSH) ]; then \
                        $(YPPUSH) locale.byname; \
                        echo "pushed locale"; \
                else \
                : ; \
                fi \
        else \
                echo "couldn't find $(DIR)/locale"; \
        fi
```

b. **Find the string `all:` and, at the end of the list of variables, insert the word `locale`.**

```
all: passwd group hosts ethers networks rpc services protocols \
    netgroup bootparams aliases publickey netid netmasks c2secure \
    timezone auto.master auto.home locale
```

c. **Near the end of the file, after the last entry of its type, insert the string `locale: locale.time` on a new line.**

```
passwd: passwd.time
group: group.time
hosts: hosts.time
ethers: ethers.time
networks: networks.time
rpc: rpc.time
services: services.time
protocols: protocols.time
netgroup: netgroup.time
bootparams: bootparams.time
aliases: aliases.time
publickey: publickey.time
netid: netid.time
passwd.adjunct: passwd.adjunct.time
group.adjunct: group.adjunct.time
netmasks: netmasks.time
timezone: timezone.time
auto.master: auto.master.time
auto.home: auto.home.time
locale: locale.time
```

d. **Save the file.**

3. **Create the file `/etc/locale` and make one entry for each domain or specific system:**

   locale domain_name

   Or

   locale system_name

   Note – *International Language Environments Guide* contains a list of valid locales.

   For example, the following entry specifies that French is the default language that is used in the `example.com` domain:

   ```
   fr example.com
   ```

And the following entry specifies that Belgian French is the default locale that is used by a system named myhost:

```
fr_BE myhost
```

Note – Locales are available on the Solaris 10 DVD or Solaris 10 Software - 1 CD.

---

4. **Make the maps:**

   ```
   # cd /var/yp; make
   ```

   Systems that are specified by domain or individually in the locale map are now set up to use the default locale. The default locale that you specified is used during installation and by the desktop after the system is rebooted.

More Information

Continuing the Installation

If you plan to use the NIS name service in an installation over the network, you need to set up an installation server and add the system as an installation client. For more information, see Chapter 5.

If you plan to use the NIS name service in a custom JumpStart installation, you need to create a profile and a rules.ok file. For more information, see Chapter 3, “Custom JumpStart (Overview),” in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

See Also

For more information about the NIS name service, see Part III, “NIS Setup and Administration,” in *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

▼ To Preconfigure the Locale Using NIS+

The following procedure assumes the NIS+ domain is set up. Setting up the NIS+ domain is documented in the *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

Steps

1. **Log in to a name server as superuser or as a user in the NIS+ administration group.**

2. **Create the locale table:**

   ```
   # nistbladm -D access=og=rmcd,nw=r -c locale_tbl name=SI,nogw=
   locale=,nogw= comment=,nogw= locale.org_dir.`nisdefaults -d`
   ```

3. **Add needed entries to the locale.**

   ```
   # nistbladm -a name=namelocale=locale comment=comment
   locale.org_dir.`nisdefaults -d`
   ```

name Either the domain name or a specific system name for which you want to preconfigure a default locale.

locale The locale you want to install on the system and use on the desktop after the system is rebooted. *International Language Environments Guide* contains a list of valid locales.

comment The comment field. Use double quotation marks to begin and end comments that are longer than one word.

Note – Locales are available on the Solaris 10 DVD or Solaris 10 Software - 1 CD.

Systems that are specified by domain or individually in the `locale` table are now set up to use the default locale. The default locale you specified is used during installation and by the desktop after the system is rebooted.

More Information

Continuing the Installation

If you plan to use the NIS+ name service in an installation over the network, you need to set up an installation server and add the system as an installation client. For more information, see Chapter 5.

If you plan to use the NIS+ name service in a custom JumpStart installation, you need to create a profile and a `rules.ok` file. For more information, see Chapter 3, "Custom JumpStart (Overview)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

See Also For more information about the NIS+ name service, see *System Administration Guide: Naming and Directory Services (NIS+)*.

Preconfiguring System Configuration Information With the DHCP Service (Tasks)

The Dynamic Host Configuration Protocol (DHCP) enables host systems in a TCP/IP network to be configured automatically for the network as they boot. DHCP uses a client and server mechanism. Servers store and manage configuration information for clients, and provide that information on a client's request. The information includes the client's IP address and information about network services available to the client.

A primary benefit of DHCP is its ability to manage IP address assignments through leasing. Leasing allows IP addresses to be reclaimed when not in use and reassigned to other clients. This ability enables a site to use a smaller pool of IP address than would be needed if all clients were assigned a permanent address.

You can use DHCP to install the Solaris OS on certain client systems on your network. Only `sun4u` based systems and x86 based systems that meet the hardware requirements for running the Solaris OS can use this feature.

The following task map shows the high-level tasks that must be performed to enable clients to obtain installation parameters by using DHCP.

TABLE 4–3 Task Map: Preconfiguring System Configuration Information With the DHCP Service

Task	Description	Instructions
Set up an install server.	Set up a Solaris server to support clients that must install the Solaris OS from the network.	Chapter 5
Set up client systems for Solaris installation over the network by using DHCP.	Use `add_install_client -d` to add DHCP network installation support for a class of client (of a certain machine type, for example) or a particular client ID.	Using Solaris DVD: "Adding Systems to Be Installed From the Network With a DVD Image" on page 108 Using Solaris CD: "Adding Systems to Be Installed From the Network With a CD Image" on page 143 `add_install_client`(1M)
Prepare your network to use the DHCP service.	Decide how you want to configure your DHCP server.	Chapter 12, "Planning for DHCP Service (Tasks)," in *System Administration Guide: IP Services*
Configure the DHCP server.	Use DHCP Manager to configure your DHCP server	Chapter 13, "Configuring the DHCP Service (Tasks)," in *System Administration Guide: IP Services*
Create DHCP options for installation parameters and macros that include the options.	Use DHCP Manager or `dhtadm` to create new Vendor options and macros that the DHCP server can use to pass installation information to the clients.	"Creating DHCP Options and Macros for Solaris Installation Parameters" on page 80

Creating DHCP Options and Macros for Solaris Installation Parameters

When you add clients with the `add_install_client -d` script on the install server, the script reports DHCP configuration information to standard output. This information can be used when you create the options and macros that are needed to pass network installation information to clients.

To install DHCP clients from the network, you must create Vendor category options to pass information that is needed to install the Solaris OS. Table 4–4 shows the options you must create and the properties that are needed to create them.

You can customize the macros in your DHCP service to perform the following types of installations.

- **Class-specific installations** - You can instruct the DHCP service to perform a network installation for all clients of a specific class. For example, you can define a DHCP macro that performs the same installation on all Sun Blade systems on the network. Use the output of the `add_install_client -d` command to set up a class-specific installation.
- **Client-specific installations** - You can instruct the DHCP service to perform a network installation for a client with a specific Ethernet address. For example, you can define a DHCP macro that performs a specific installation on the client with the Ethernet address 00:07:e9:04:4a:bf. Use the output of the `add_install_client -d -e` *ethernet_address* command to set up a client-specific installation.

For more information on setting up clients to install from the network, see the following procedures.

- For network installations that use DVD media, see "Adding Systems to Be Installed From the Network With a DVD Image" on page 108.
- For network installations that use CD media, see "Adding Systems to Be Installed From the Network With a CD Image" on page 143.

The vendor client classes that are listed in the following table determine what classes of client can use the option. Vendor client classes that are listed here are examples only. You should specify client classes that indicate the actual clients in your network that you need to install from the network. See "Working With DHCP Options (Task Map)" in *System Administration Guide: IP Services* for information about how to determine a client's vendor client class.

For detailed information on DHCP options, see "DHCP Option Information" in *System Administration Guide: IP Services*.

TABLE 4–4 Values for Creating Vendor Category Options for Solaris Clients

Name	Code	Data Type	Granularity	Maximum	Vendor Client Classes *	Description
The following Vendor category options are required to enable a DHCP server to support Solaris installation clients. The options are used in the Solaris client's startup scripts.						

`SrootIP4`	2	IP address	1	1	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						IP address of root server

`SrootNM`	3	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						Host name of root server

`SrootPTH`	4	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						Path to the client's root directory on the root server

`SinstIP4`	10	IP address	1	1	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						IP address of JumpStart install server

`SinstNM`	11	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						Host name of install server

`SinstPTH`	12	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						Path to installation image on install server

The following options can be used by the client startup scripts, but are not required by the scripts.						

`SrootOpt`	1	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						NFS mount options for the client's root file system

`SbootFIL`	7	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						Path to the client's boot file

`SbootRS`	9	NUMBER	2	1	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						NFS read size used by standalone boot program when loading the kernel

`SsysidCF`	13	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						Path to `sysidcfg` file, in the format *server:/path*

`SjumpsCF`	14	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						Path to JumpStart configuration file in the format *server:/path*

TABLE 4–4 Values for Creating Vendor Category Options for Solaris Clients *(Continued)*

Name	Code	Data Type	Granularity	Maximum	Vendor Client Classes *	Description

`SbootURI`	16	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						Path to the standalone boot file or path to the WAN boot file. For the standalone boot file, use the following format.
						`tftp://inetboot.sun4u`
						For the WAN boot file, the format is
						`http://`*host.domain/path-to-file*
						This option can be used to override `BootFile` and `siaddr` settings in order to retrieve a standalone boot file. Supported protocols: tftp (inetboot), http (wanboot). For example, use the following format. `tftp://inetboot.sun4u`

`SHTTPproxy`	17	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						IP address and port number of the proxy server that is used on your network. This option is needed only when a client is booting across a WAN, and the local network uses a proxy server. For example, use the following format: `198.162.10.5:8080`

The following options are not currently used by the Solaris client startup scripts. You can use them only if you edit the startup scripts.						

`SswapIP4`	5	IP address	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						IP address of swap server

`SswapPTH`	6	ASCII text	1	0	`SUNW.Sun-Blade-1000,`
					`SUNW.Sun-Fire-880,`
					`SUNW.i86pc`
						Path to the client's swap file on the swap server

TABLE 4–4 Values for Creating Vendor Category Options for Solaris Clients *(Continued)*

Name	Code	Data Type	Granularity	Maximum	Vendor Client Classes *	Description
Stz	8	ASCII text	1	0	SUNW.Sun-Blade-1000, SUNW.Sun-Fire-880, SUNW.i86pc	Time zone for client
Sterm	15	ASCII text	1	0	SUNW.Sun-Blade-1000, SUNW.Sun-Fire-880, SUNW.i86pc	Terminal type

When you have created the options, you can create macros that include those options. The following table lists sample macros you can create to support Solaris installation for clients.

TABLE 4–5 Sample Macros to Support Network Installation Clients

Macro Name	Contains These Options and Macros
Solaris	SrootIP4, SrootNM, SinstIP4, SinstNM
sparc	SrootPTH, SinstPTH
sun4u	Solaris and sparc macros
i86pc	Solaris macro, SrootPTH, SinstPTH, SbootFIL
SUNW.i86pc	i86pc macro
SUNW.Sun-Blade-1000	sun4u macro, SbootFIL
SUNW.Sun-Fire-880	sun4u macro, SbootFIL
xxx.xxx.xxx.xxx network address macros	BootSrvA option could be added to existing network address macros. The value of BootSrvA should indicate the tftboot server.

The macro names that are listed in the previous table match the Vendor client classes of the clients that must install from the network. These names are examples of clients you might have on your network. See "Working With DHCP Options (Task Map)" in *System Administration Guide: IP Services* for information about determining a client's vendor client class.

You can create these options and macros by using the following methods.

- Write a script that creates the options and macros by using the dhtadm command. See "Writing a Script That Uses dhtadm to Create Options and Macros" on page 84 for information bout how to write scripts that create these options and macros.
- Create the options and macros in DHCP Manager. See "Using DHCP Manager to Create Install Options and Macros" on page 86 for instructions about how to create options and macros in DHCP Manager.

Note that the sum total of the values assigned to all the options in a macro must not exceed 255 bytes, including the option codes and length information. This limit is dictated by the DHCP protocol. Generally, you should pass the minimum amount of vendor information needed. You should use short path names in options that require path names. If you create symbolic links to long paths, you can pass the shorter link names.

Writing a Script That Uses dhtadm to Create Options and Macros

You can create a Korn shell script by adapting the example in Example 4–13 to create all the options listed in Table 4–4 and some useful macros. Be sure to change all IP addresses and values contained in quotes to the correct IP addresses, server names, and paths for your network. You should also edit the `Vendor=` key to indicate the class of clients you have. Use the information that `add_install_client -d` reports to obtain the data that you need to adapt the script.

EXAMPLE 4–13 Sample Script to Support Network Installation

```
# Load the Solaris vendor specific options. We'll start out supporting
# the Sun-Blade-1000, Sun-Fire-880, and i86 platforms. Changing -A to -M would replace
# the current values, rather than add them.
dhtadm -A -s SrootOpt -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,1,ASCII,1,0'
dhtadm -A -s SrootIP4 -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,2,IP,1,1'
dhtadm -A -s SrootNM -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,3,ASCII,1,0'
dhtadm -A -s SrootPTH -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,4,ASCII,1,0'
dhtadm -A -s SswapIP4 -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,5,IP,1,0'
dhtadm -A -s SswapPTH -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,6,ASCII,1,0'
dhtadm -A -s SbootFIL -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,7,ASCII,1,0'
dhtadm -A -s Stz -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,8,ASCII,1,0'
dhtadm -A -s SbootRS -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,9,NUMBER,2,1'
dhtadm -A -s SinstIP4 -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,10,IP,1,1'
dhtadm -A -s SinstNM -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,11,ASCII,1,0'
dhtadm -A -s SinstPTH -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,12,ASCII,1,0'
dhtadm -A -s SsysidCF -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,13,ASCII,1,0'
dhtadm -A -s SjumpsCF -d \
'Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,14,ASCII,1,0'
dhtadm -A -s Sterm -d \
```

EXAMPLE 4–13 Sample Script to Support Network Installation *(Continued)*

```
’Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,15,ASCII,1,0’
dhtadm -A -s SbootURI -d \
’Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,16,ASCII,1,0’
dhtadm -A -s SHTTPproxy -d \
’Vendor=SUNW.Sun-Blade-1000 SUNW.Sun-Fire-880 SUNW.i86pc,17,ASCII,1,0’
# Load some useful Macro definitions.
# Define all Solaris-generic options under this macro named Solaris.
dhtadm -A -m Solaris -d \
’:SrootIP4=10.21.0.2:SrootNM="blue2":SinstIP4=10.21.0.2:SinstNM="red5":’
# Define all sparc-platform specific options under this macro named sparc.
dhtadm -A -m sparc -d \
’:SrootPTH="/export/sparc/root":SinstPTH="/export/sparc/install":’
# Define all sun4u architecture-specific options under this macro named sun4u.
#  (Includes Solaris and sparc macros.)
dhtadm -A -m sun4u -d ’:Include=Solaris:Include=sparc:’
# Solaris on IA32-platform-specific parameters are under this macro named i86pc.
dhtadm -A -m i86pc -d \
’:Include=Solaris:SrootPTH="/export/i86pc/root":SinstPTH="/export/i86pc/install"\
:SbootFIL="/platform/i86pc/kernel/unix":’
# Solaris on IA32 machines are identified by the "SUNW.i86pc" class. All
# clients identifying themselves as members of this class will see these
# parameters in the macro called SUNW.i86pc, which includes the i86pc macro.
dhtadm -A -m SUNW.i86pc -d ’:Include=i86pc:’
# Sun-Blade-1000 platforms identify themselves as part of the
# "SUNW.Sun-Blade-1000" class.
# All clients identifying themselves as members of this class
#  will see these parameters.
dhtadm -A -m SUNW.Sun-Blade-1000 -d \
’:SbootFIL="/platform/sun4u/kernel/sparcv9/unix":\
Include=sun4u:’
# Sun-Fire-880 platforms identify themselves as part of the "SUNW.Sun-Fire-880" class.
# All clients identifying themselves as members of this class will see these parameters.
dhtadm -A -m SUNW.Sun-Fire-880 -d \
’:SbootFIL="/platform/sun4u/kernel/sparcv9/unix":Include=sun4u:’
# Add our boot server IP to each of the network macros for our topology served by our
# DHCP server. Our boot server happens to be the same machine running our DHCP server.
dhtadm -M -m 10.20.64.64 -e BootSrvA=10.21.0.2
dhtadm -M -m 10.20.64.0 -e BootSrvA=10.21.0.2
dhtadm -M -m 10.20.64.128 -e BootSrvA=10.21.0.2
dhtadm -M -m 10.21.0.0 -e BootSrvA=10.21.0.2
dhtadm -M -m 10.22.0.0    -e BootSrvA=10.21.0.2
# Make sure we return host names to our clients.
dhtadm -M -m DHCP-servername -e Hostname=_NULL_VALUE_
# The client with this MAC address is a diskless client. Override the root settings
# which at the network scope setup for Install with our client’s root directory.
dhtadm -A -m 0800201AC25E -d \
’:SrootIP4=10.23.128.2:SrootNM="orange-svr-2":SrootPTH="/export/root/10.23.128.12":’
```

As superuser, execute `dhtadm` in batch mode. Specify the name of the script to add the options and macros to your `dhcptab`. For example, if your script is named `netinstalloptions`, type the following command.

```
# dhtadm -B netinstalloptions
```

Clients that have vendor client classes that are listed in the `Vendor=` string can now use DHCP to install over the network.

For more information about how to use the `dhtadm` command, see `dhtadm`(1M). For more information about the `dhcptab` file, see `dhcptab`(4).

Using DHCP Manager to Create Install Options and Macros

You can use DHCP Manager to create the options that are listed in Table 4–4 and the macros that are listed in Table 4–5.

▼ How to Create Options to Support Solaris Installation (DHCP Manager)

This procedure assumes that you have already configured your DHCP server. If you have not configured your DHCP server, see Chapter 12, "Planning for DHCP Service (Tasks)," in *System Administration Guide: IP Services*.

Steps

1. **Become superuser on the DHCP server system.**
2. **Start the DHCP Manager.**

   ```
   # /usr/sadm/admin/bin/dhcpmgr &
   ```

   The DHCP Manager window is displayed.
3. **Select the Options tab in DHCP Manager.**
4. **Choose Create from the Edit menu.**

   The Create Option dialog box opens.
5. **Type the option name for the first option, then type values appropriate for that option.**

   Use Table 4–4 to check the option names and values for options you must create. Notice that the vendor client classes are only suggested values. You should create classes to indicate the actual client types that need to obtain Solaris installation parameters from the DHCP service. See "Working With DHCP Options (Task Map)" in *System Administration Guide: IP Services* for information about how to determine a client's vendor client class.
6. **Click OK when you have entered all the values.**
7. **In the Options tab, select the option you just created.**

8. **Select Duplicate from the Edit menu.**

   The Duplicate Option dialog box opens.

9. **Type the name of another option, then modify other values appropriately.**

   The values for code, data type, granularity, and maximum are most likely to need modification. See Table 4–4 for the values.

10. **Repeat Step 7 through Step 9 until you have created all the options.**

    You can now create macros to pass the options to network installation clients, as explained in the following procedure.

Note – You do not need to add these options to a Solaris client's `/etc/dhcp/inittab` file because they are already included in that file.

▼ How to Create Macros to Support Solaris Installation (DHCP Manager)

This procedure assumes that you have already configured your DHCP server. If you have not configured your DHCP server, see Chapter 12, "Planning for DHCP Service (Tasks)," in *System Administration Guide: IP Services*.

Steps

1. **Select the Macros tab in DHCP Manager.**

2. **Choose Create from the Edit menu.**

   The Create Macro dialog box opens.

3. **Type the name of a macro.**

   See Table 4–5 for macro names you might use.

4. **Click the Select button.**

   The Select Option dialog box opens.

5. **Select Vendor in the Category list.**

   The Vendor options you created are listed.

6. **Select an option you want to add to the macro and click OK.**

7. **Type a value for the option.**

   See Table 4–4 for the option's data type and refer to the information that `add_install_client -d` reports.

8. **Repeat Step 6 through Step 7 for each option you want to include.**

   To include another macro, type **`Include`** as the option name and type the macro name as the option value.

9. **Click OK when the macro is complete.**

More Information

Continuing the Installation

If you plan to use DHCP in an installation over the network, you need to set up an installation server and add the system as an installation client. For more information, see Chapter 5.

If you plan to use DHCP in a WAN boot installation, you need to perform additional tasks. For more information, see Chapter 9.

If you plan to use DHCP in a custom JumpStart installation, you need to create a profile and a `rules.ok` file. For more information, see Chapter 3, "Custom JumpStart (Overview)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

See Also For more information about DHCP, see Part III, "DHCP," in *System Administration Guide: IP Services*.

SPARC: Preconfiguring Power Management Information

You can use the *Power Management* software that is provided in the Solaris OS to automatically save the state of a system and turn it off after it is idle for 30 minutes. When you install the Solaris 10 OS on a system that complies with version 2 of the EPA's Energy Star guidelines, for example a sun4u system, the Power Management software is installed by default. If you install with the Solaris installation program GUI, the installation program prompts you to enable or disable the Power Management software. The Solaris text installer prompts you to enable or disable the Power Management software after the installation is complete and the system reboots.

Note – If your system has Energy Star version 3 or later, you are not prompted for this information.

If you are performing interactive installations, you cannot preconfigure the Power Management information and avoid the prompt. However, by using a custom JumpStart installation, you can preconfigure the Power Management information by using a finish script to create an `/autoshutdown` or `/noautoshutdown` file on the system. When the system reboots, the `/autoshutdown` file enables Power Management and the `/noautoshutdown` file disables Power Management.

For example, the following line in a finish script enables the Power Management software and prevents the display of the prompt after the system reboots.

```
touch /a/autoshutdown
```

Finish scripts are described in “Creating Finish Scripts” in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

PART II Installing Over a Local Area Network

This part describes how to install a system that is on your local area network (LAN).

CHAPTER 5

Preparing to Install From the Network (Overview)

This chapter provides an introduction on how to set up your local area network and systems to install the Solaris software from the network instead of from DVD or CD media.

For information on how to install a client over a wide area network, see Chapter 9.

Planning for a Network Installation Introduction

This section provides you with information you need before you can perform an installation from the network. Network installations enable you to install the Solaris software from a system, called an install server, that has access to the Solaris 10 disc images. You copy the contents of the Solaris 10 DVD or CD media to the install server's hard disk. Then, you can install the Solaris software from the network by using any of the Solaris installation methods.

Required Servers for Network Installation

To install the Solaris OS from the network, the systems to be installed require the following servers to be present on the network.

- **Install server –** A networked system that contains the Solaris 10 disc images from which you can install Solaris 10 software on other systems on the network. You create an install server by copying the images from the following media:
  - Solaris 10 DVD
  - Solaris 10 Software CDs

After you copy the image from the Solaris 10 Software CDs, you can also copy the image from the Solaris 10 Languages CD as necessary for your installation requirements.

You can enable a single install server to provide disc images for different Solaris releases and for multiple platforms by copying the images on to the install server's hard disk. For example, a single install server could contain the disc images for the SPARC platform and x86 platform.

For details about how to create an install server, refer to one of the following sections.

- "SPARC: To Create a SPARC Install Server With SPARC or x86 DVD Media" on page 99
- "x86: To Create an x86 Install Server With SPARC or x86 DVD Media" on page 103
- "SPARC: To Create a SPARC Install Server With SPARC CD Media" on page 121
- "Creating a Cross-Platform Install Server for CD Media" on page 131

- **Boot server –** A server system that provides client systems on the same network subnet with the information that they need to boot in order to install the OS. A boot server and install server are typically the same system. However, if the system on which the Solaris 10 software is to be installed is located in a different subnet than the install server and you are not using DHCP, a boot server is required on that subnet.

  A single boot server can provide Solaris 10 boot software for multiple releases, including the Solaris 10 boot software for different platforms. For example, a SPARC boot server can provide the Solaris 9 and Solaris 10 boot software for SPARC based systems. The same SPARC boot server can also provide the Solaris 10 boot software for x86 based systems.

  Note – When using DHCP, you do not need to create a separate boot server. For more information, see "Using DHCP to Provide Network Installation Parameters" on page 95.

  For details about how to create a boot server, refer to one of the following sections:

  - "Creating a Boot Server on a Subnet With a DVD Image" on page 106
  - "Creating a Boot Server on a Subnet With a CD Image" on page 141

- **(Optional) Name server –** A system that manages a distributed network database, such as DNS, NIS, NIS+, or LDAP, that contains information about systems on the network.

  For details about how to create a name server, refer to *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

Note – The install server and name server can be the same or different systems.

Figure 5–1 illustrates the servers that are typically used for network installation.

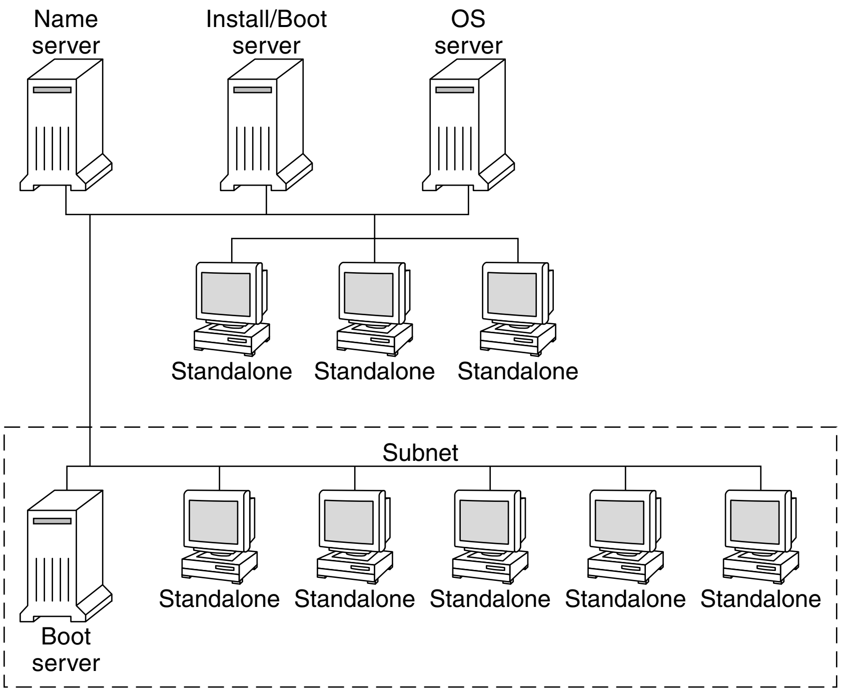

FIGURE 5–1 Network Installation Servers

Using DHCP to Provide Network Installation Parameters

Dynamic Host Configuration Protocol (DHCP) provides the network parameters that are necessary for installation. When using DHCP, you do not need to create a separate boot server. After you have created the install server, you add clients to the network with the `add_install_client` command and the `-d` option. The `-d` option enables you to set up client systems for Solaris installation from the network by using DHCP.

For information on DHCP options for installation parameters, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.

CHAPTER 6

Preparing to Install From the Network With DVD Media (Tasks)

This chapter describes how to use DVD media to set up your network and systems to install the Solaris software from the network. Network installations enable you to install the Solaris software from a system that has access to the Solaris 10 disc images, called an install server, to other systems on the network. You copy the contents of the Solaris 10 DVD media to the install server's hard disk. Then, you can install the Solaris software from the network by using any of the Solaris installation methods. This chapter covers the following topics:

- "Task Map: Preparing to Install From the Network With DVD Media" on page 97
- "Creating an Install Server With DVD Media" on page 98
- "Creating a Boot Server on a Subnet With a DVD Image" on page 106
- "Adding Systems to Be Installed From the Network With a DVD Image" on page 108
- "Booting and Installing the System From the Network With a DVD Image" on page 114

Task Map: Preparing to Install From the Network With DVD Media

TABLE 6–1 Task Map: Setting Up an Install Server With DVD Media

Task	Description	For Instructions
Create an install server.	Use the `setup_install_server`(1M)command to copy the Solaris 10 DVD to the install server's hard disk.	"Creating an Install Server With DVD Media" on page 98

TABLE 6–1 Task Map: Setting Up an Install Server With DVD Media *(Continued)*

Task	Description	For Instructions
(Optional) Create boot servers.	If you want to install systems from the network that are not on the same subnet as the install server, you must create a boot server on the subnet to boot the systems. Use the `setup_install_server` command with the `-b` option to set up a boot server. If you are using Dynamic Host Configuration Protocol (DHCP), a boot server is not necessary.	"Creating a Boot Server on a Subnet With a DVD Image" on page 106
Add systems to be installed from the network.	Use the `add_install_client` command to set up each system that you want to install from the network. Each system that you want to install needs to find the install server, the boot server if required, and configuration information on the network.	"Adding Systems to Be Installed From the Network With a DVD Image" on page 108
Install the system over the network.	Begin the installation by booting the system from the network.	"Booting and Installing the System From the Network With a DVD Image" on page 114

Creating an Install Server With DVD Media

The install server contains the installation image needed to install systems from the network. You must create an install server to install the Solaris software on a system from the network. You do not always need to set up a boot server.

- If you are using DHCP to set installation parameters or your install server and client are on the same subnet, you do not need a boot server.
- If your install server and your client are not on the same subnet and you are not using DHCP, you must create separate boot servers for each subnet. You could create an install server for each subnet. However, install servers require more disk space.

Note – If you want use the Solaris DVD media to set up an install server on a system that is running the Solaris 7 OS, you must first apply one of the following patches.

- Solaris 7 *SPARC Platform Edition* operating environment - Patch ID 107259-03
- Solaris 7 *Intel Platform Edition* operating environment - Patch ID 107260-03

▼ SPARC: To Create a SPARC Install Server With SPARC or x86 DVD Media

---

Note – SPARC: You cannot use a system that is running a SunOS version that was released prior to the Solaris 2.3 release.

Note – This procedure assumes that the system is running the Volume Manager. If you are not using the Volume Manager to manage media, refer to *System Administration Guide: Devices and File Systems* for detailed information about managing removable media without the Volume Manager.

Steps

1. **On the SPARC system that is to become the install server, become superuser.**

   The system must include a DVD-ROM drive and be part of the site's network and name service. If you use a name service, the system must already be in a service, such as NIS, NIS+, DNS, or LDAP. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **Insert the Solaris 10 DVD in the SPARC system's drive.**

3. **Create a directory to contain the DVD image.**

   ```
   # mkdir -p install_dir_path
   ```

4. **Change to the `Tools` directory on the mounted disc.**

   - For SPARC DVD media, type:

     ```
     # cd /cdrom/cdrom0/s0/Solaris_10/Tools
     ```

   - For x86 DVD media, type:

     ```
     # cd /cdrom/cdrom0/Solaris_10/Tools
     ```

   In the previous examples, **`cdrom0`** is the path to the drive that contains the Solaris OS DVD media.

5. **Copy the DVD image in the drive to the install server's hard disk.**

   ```
   # ./setup_install_server install_dir_path
   ```

   install_dir_path Specifies the directory where the DVD image is to be copied

   ---

   Note – The `setup_install_server` command indicates whether you have enough disk space available for the Solaris 10 Software disc images. To determine available disk space, use the `df -kl` command.

   ---

6. **Decide if you need to make the install server available for mounting.**

   - **If the install server is on the same subnet as the system to be installed or you are using DHCP, you do not need to create a boot server. Proceed to Step 7.**
   - **If the install server is not on the same subnet as the system to be installed and you are not using DHCP, complete the following steps.**

     a. **Verify that the path to the install server's image is shared appropriately.**

     ```
     # share | grep install_dir_path
     ```

install_dir_path	Specifies the path to the installation image where the DVD image was copied

     - **If the path to the install server's directory is displayed and anon=0 is displayed in the options, proceed to Step 7.**
     - **If the path to the install server's directory is not displayed or you do not have anon=0 in the options, continue.**

     b. **Make the install server available to the boot server by adding this entry to the `/etc/dfs/dfstab` file.**

     ```
     share -F nfs -o ro,anon=0 -d "install server directory" install_dir_path
     ```

     c. **Verify that the `nfsd` daemon is running.**

     - **If the install server is running the Solaris 10 OS, or compatible version, type the following command.**

       ```
       # svcs -l svc:/network/nfs/server:default
       ```

       If the `nfsd` daemon is online, continue to Step d. If the `nfsd` daemon is not online, start it.

       ```
       # svcadm enable svc:/network/nfs/server
       ```

     - **If the install server is running the Solaris 9 OS, or compatible version, type the following command.**

       ```
       # ps -ef | grep nfsd
       ```

       If the `nfsd` daemon is running, continue to Step d. If the `nfsd` daemon is not running, start it.

       ```
       # /etc/init.d/nfs.server start
       ```

     d. **Share the install server.**

     ```
     # shareall
     ```

7. **Change directories to root (/).**

   ```
   # cd /
   ```

8. **Eject the Solaris 10 DVD.**

9. **Decide if you want to patch the files that are located in the miniroot (/*install_dir_path*/Solaris_10/Tools/Boot) on the net install image that was created by `setup_install_server`. Patching a file might be necessary if a boot image has problems.**

   - **If no, continue.**
   - **If yes, use the `patchadd -C` command to patch the files that are located in the miniroot.**

   Caution – Don't use the `patchadd -C` command unless you have read the Patch README instructions or have contacted your local Sun support office.

10. **Decide if you need to create a boot server.**

    - **If you are using DHCP or the install server is on the same subnet as the system to be installed, you do not need to create a boot server. Proceed to "Adding Systems to Be Installed From the Network With a DVD Image" on page 108.**
    - **If you are *not* using DHCP and the install server and the client are on a different subnet, you must create a boot server. Proceed to "Creating a Boot Server on a Subnet With a DVD Image" on page 106.**

Example 6–1 SPARC: Creating a SPARC Install Server With a SPARC DVD

The following example illustrates how to create an install server by copying the Solaris 10 DVD to the install server's /export/home/dvdsparc directory. This example assumes that the install server is running the Solaris 10 OS.

```
# mkdir -p /export/home/dvdsparc
# cd /cdrom/cdrom0/s0/Solaris_10/Tools
# ./setup_install_server /export/home/dvdsparc
```

If you need a separate boot server, type these commands:

Add the following path to the /etc/dfs/dfstab file:

```
share -F nfs -o ro,anon=0 -d "install server directory" /export/home/dvdsparc
```

Check if the nfsd daemon is online. If the nfsd daemon is not online, start it and share it.

```
# svcs -l svc:/network/nfs/server:default
# svcadm enable svc:/network/nfs/server
# shareall
# cd /
```

Example 6–2 x86: Creating a SPARC Install Server With an x86 DVD

The following example illustrates how to create an install server by copying the Solaris 10 DVD to the install server's /export/home/dvdx86 directory. This example assumes that the install server is running the Solaris 10 OS.

```
# mkdir -p /export/home/dvdx86
# cd /cdrom/cdrom0/Solaris_10/Tools
# ./setup_install_server /export/home/dvdx86
```

Add the following path to the /etc/dfs/dfstab file:

```
share -F nfs -o ro,anon=0 -d "install server directory" /export/home/dvdx86
```

Check if the nfsd daemon is online. If the nfsd daemon is not online, start it and share it.

```
# svcs -l svc:/network/nfs/server:default
# svcadm enable svc:/network/nfs/server
# shareall
# cd /
```

More Information Continuing the Installation

After you set up the install server, you must add the client as an installation client. For information about how to add client systems to install over the network, see "To Add Systems to Be Installed From the Network With add_install_client (DVD)" on page 109.

If you are not using DHCP, and your client system is on a different subnet than your install server, you must create a boot server. For more information, see "Creating a Boot Server on a Subnet With a DVD Image" on page 106.

See Also

For additional information about the setup_install_server and the add_to_install_server commands, see install_scripts(1M).

▼ x86: To Create an x86 Install Server With SPARC or x86 DVD Media

---

Note – This procedure assumes that the system is running the Volume Manager. If you are not using the Volume Manager to manage media, refer to *System Administration Guide: Devices and File Systems* for detailed information about managing removable media without the Volume Manager.

Steps

1. **On the x86 system that is to become the install server, become superuser.**

   The system must include a DVD-ROM drive and be part of the site's network and name service. If you use a name service, the system must also be in the NIS, NIS+, DNS, or LDAP name service. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **Insert the Solaris 10 DVD into the system's drive.**

3. **Create a directory to contain the boot image.**

   ```
   # mkdir -p install_dir_path
   ```

   install_dir_path Specifies the directory where the DVD image is to be copied

4. **Change to the `Tools` directory on the mounted disc:**

   - For x86 DVD media, type:

     ```
     # cd /cdrom/cdrom0/s2/Solaris_10/Tools
     ```

   - For SPARC DVD media, type:

     ```
     # cd /cdrom/cdrom0/Solaris_10/Tools
     ```

   In the previous examples, **`cdrom0`** is the path to the drive that contains the Solaris OS DVD media.

5. **Copy the disc in the drive to the install server's hard disk by using the `setup_install_server` command:**

   ```
   # ./setup_install_server install_dir_path
   ```

   install_dir_path Specifies the directory where the DVD image is to be copied

   ---

   Note – The `setup_install_server` command indicates whether you have enough disk space available for the Solaris 10 Software disc images. To determine available disk space, use the `df -kl` command.

   ---

6. **Decide if you need to make the install server available for mounting.**

- **If the install server is on the same subnet as the system to be installed or you are using DHCP, you do not need to create a boot server. Proceed to Step 7.**

- **If the install server is not on the same subnet as the system to be installed and you are not using DHCP, complete the following steps.**

  a. **Verify that the path to the install server's image is shared appropriately.**

  ```
  # share | grep install_dir_path
  ```

  install_dir_path Specifies the installation image where the DVD image was copied

  - **If the path to the install server's directory is displayed and anon=0 is displayed in the options, proceed to Step 7.**

  - **If the path to the install server's directory is not displayed or you do not have anon=0 in the options, continue.**

  b. **Make the install server available to the boot server by adding this entry to the `/etc/dfs/dfstab` file.**

  ```
  share -F nfs -o ro,anon=0 -d "install server directory" install_dir_path
  ```

  c. **Verify that the `nfsd` daemon is running.**

  - **If the install server is running the Solaris 10 OS, or compatible version, type the following command.**

    ```
    # svcs -l svc:/network/nfs/server:default
    ```

    If the `nfsd` daemon is online, continue to Step d. If the `nfsd` daemon is not online, start it.

    ```
    # svcadm enable svc:/network/nfs/server
    ```

  - **If the install server is running the Solaris 9 OS, or compatible version, type the following command.**

    ```
    # ps -ef | grep nfsd
    ```

    If the `nfsd` daemon is running, continue to Step d. If the `nfsd` daemon is not running, start it.

    ```
    # /etc/init.d/nfs.server start
    ```

  d. **Share the install server.**

  ```
  # shareall
  ```

7. **Change directories to root (/).**

   ```
   # cd /
   ```

8. **Eject the Solaris 10 DVD.**

9. **Decide if you want to patch the files that are located in the miniroot (`Solaris_10/Tools/Boot`) on the net install image that was created by `setup_install_server`.**
   - **If no, continue.**
   - **If yes, use the `patchadd -C` command to patch the files that are located in the miniroot.**

10. **Decide if you need to create a boot server.**
    - **If the install server is on the same subnet as the system to be installed or you are using DHCP, you do not need to create a boot server. See "Adding Systems to Be Installed From the Network With a DVD Image" on page 108.**
    - **If the install server is not on the same subnet as the system to be installed and you are not using DHCP, you must create a boot server. For detailed instructions on how to create a boot server, refer to "Creating a Boot Server on a Subnet With a DVD Image" on page 106.**

Example 6–3 x86: Creating an x86 Install Server With an x86 DVD

The following example illustrates how to create an x86 install server by copying the Solaris 10 Operating System for x86 Platforms DVD to the install server's `/export/home/dvdx86` directory. This example assumes that the install server is running the Solaris 10 OS.

```
# mkdir -p /export/home/dvdx86
# cd /cdrom/cdrom0/s2/Solaris_10/Tools
# ./setup_install_server /export/home/dvdx86
```

Add the following path to the `/etc/dfs/dfstab` file:

```
share -F nfs -o ro,anon=0 -d "install server directory" /export/home/dvdx86
```

Check if the `nfsd` daemon is online. If the `nfsd` daemon is not online, start it and share it.

```
# svcs -l svc:/network/nfs/server:default
# svcadm enable svc:/network/nfs/server
# shareall
# cd /
```

Example 6–4 Creating an x86 Install Server With a SPARC DVD

The following example illustrates how to create an x86 install server by copying the Solaris 10 Operating System for SPARC Platforms DVD to the install server's `/export/home/dvdsparc` directory. This example assumes that the install server is running the Solaris 10 OS.

```
# mkdir -p /export/home/dvdscparc
# cd /cdrom/cdrom0/Solaris_10/Tools
# ./setup_install_server /export/home/dvdsparc
```

Add the following path to the /etc/dfs/dfstab file:

```
share -F nfs -o ro,anon=0 -d "install server directory" /export/home/dvdsparc
```

Check if the nfsd daemon is online. If the nfsd daemon is not online, start it and share it.

```
# svcs -l svc:/network/nfs/server:default
# svcadm enable svc:/network/nfs/server
# shareall
# cd /
```

More Information

Continuing the Installation

After you set up the install server, you must add the client as an installation client. For information about how to add client systems to install over the network, see "To Add Systems to Be Installed From the Network With add_install_client (DVD)" on page 109.

If you are not using PXE, and your client system is on a different subnet than your install server, you must create a boot server. For more information, see "Creating a Boot Server on a Subnet With a CD Image" on page 141.

See Also For additional information about the setup_install_server and the add_to_install_server commands, see install_scripts(1M).

Creating a Boot Server on a Subnet With a DVD Image

You must create an install server to install the Solaris software on a system from the network. You do not always need to set up a boot server. A boot server contains enough of the boot software to boot systems from the network, and then the install server completes the installation of the Solaris software.

- If you are using DHCP to set installation parameters or your install server or client is on the same subnet as the install server, you do not need a boot server. Proceed to "Adding Systems to Be Installed From the Network With a DVD Image" on page 108.

- If your install server and your client are not on the same subnet and you are not using DHCP, you must create separate boot servers for each subnet. You could create an install server for each subnet; however, install servers require more disk space.

▼ To Create a Boot Server on a Subnet With a DVD Image

Steps

1. **On the system you intend to make the boot server for the subnet, log in and become superuser.**

   The system must have access to a remote Solaris 10 disc image, which is normally the install server. If you use a name service, the system should also be in a name service. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **Mount the Solaris 10 DVD from the install server.**

   ```
   # mount -F nfs -o ro server_name:path /mnt
   ```

   server_name:*path* Is the install server name and absolute path to the disc image

3. **Create a directory for the boot image.**

   ```
   # mkdir -p boot_dir_path
   ```

   boot_dir_path Specifies the directory where the boot software is to be copied

4. **Change to the `Tools` directory on the Solaris 10 DVD image.**

   ```
   # cd /mnt/Solaris_10/Tools
   ```

5. **Copy the boot software to the boot server.**

   ```
   # ./setup_install_server -b boot_dir_path
   ```

   `-b` Specifies to set up the system as a boot server

   boot_dir_path Specifies the directory where the boot software is to be copied

   Note – The `setup_install_server` command indicates whether you have enough disk space available for the images. To determine available disk space, use the `df -kl` command.

6. **Change directories to root (/).**

   ```
   # cd /
   ```

7. **Unmount the installation image.**

   ```
   # umount /mnt
   ```

   You are now ready to set up systems to be installed from the network. See "Adding Systems to Be Installed From the Network With a DVD Image" on page 108.

Example 6–5 Creating a Boot Server on a Subnet (DVD)

The following example illustrates how to create a boot server on a subnet. These commands copy the boot software from the Solaris 10 DVD image to /export/home/dvdsparc on the local disk of a boot server named crystal.

```
# mount -F nfs -o ro crystal:/export/home/dvdsparc /mnt
# mkdir -p  /export/home/dvdsparc
# cd /mnt/Solaris_10/Tools
# ./setup_install_server -b /export/home/dvdsparc
# cd /
# umount /mnt
```

More Information Continuing the Installation

After you set up the boot server, you must add the client as an installation client. For information about how to add client systems to install over the network, see "Adding Systems to Be Installed From the Network With a DVD Image" on page 108.

See Also For additional information about the setup_install_server command, see install_scripts(1M).

Adding Systems to Be Installed From the Network With a DVD Image

After you create an install server and, if necessary, a boot server, you must set up each system that you want to install from the network.

Use the following add_install_client procedure for setting up install servers and clients. Also, see the example procedures for the following:

- If you are using DHCP to set installation parameters for a SPARC client, see Example 6–6.

- If your install server and client are on the same subnet, see Example 6–7.
- If your install server and your client are not on the same subnet and you are not using DHCP, see Example 6–8.
- If you are using DHCP to set installation parameters for x86 clients, see Example 6–9.
- If you want to use a specific serial port to display output during the installation of an x86 based system, see Example 6–10.
- If you want to set up an x86 client to use a specific network interface during the installation, see Example 6–11.

For more options to use with this command, see the man page, `add_install_client`(1M).

▼ To Add Systems to Be Installed From the Network With `add_install_client` (DVD)

After you create an install server, you must set up each system that you want to install from the network.

Use the following `add_install_client` procedure for set up an x86 client to install from the network.

Before You Begin

If you have a boot server, make sure you have shared the install server installation image and started the appropriate services. See "To Create a SPARC Install Server With SPARC or x86 DVD Media" Step 6.

Each system that you want to install needs to find the following items.

- Install server
- Boot server if it is required
- `sysidcfg` file if you use a `sysidcfg` file to preconfigure system information
- Name server if you use a name service to preconfigure system information
- The profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method

Steps

1. **On the install server or boot server, become superuser.**
2. **If you use the NIS, NIS+, DNS, or LDAP name service, verify that the following information about the system to be installed has been added to the name service.**
   - Host name
   - IP address

- Ethernet address

For more information on name services, see *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

3. **Change to the `Tools` directory on the Solaris 10 DVD image:**

```
# cd /install_dir_path/Solaris_10/Tools
```

install_dir_path Specifies the path to the `Tools` directory

4. **Set up the client system so it can be installed from the network.**

```
# ./add_install_client -d -s install_server:install_dir_path \
-c jumpstart_server:jumpstart_dir_path  -p sysid_server:path \
-t boot_image_path -b "boot-property=value" \
-e ethernet_address client_name platform_group
```

`-d`

Specifies that the client is to use DHCP to obtain the network install parameters. If you use the `-d` only, the `add_install_client` command sets up the installation information for client systems of the same class, for example, all SPARC client machines. To set up the installation information for a specific client, use the `-d` with the `-e` option.

For x86 clients, use this option to boot the systems from the network by using PXE network boot.

For more information about class-specific installations by using DHCP, see "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 80.

`-s` *install_server:install_dir_path*

Specifies the name and path to the install server.

- *install_server* is the host name of the install server.
- *install_dir_path* is the absolute path to the Solaris 10 DVD image.

`-c` *jumpstart_server*:*jumpstart_dir_path*

Specifies a JumpStart directory for custom JumpStart installations. *jumpstart_server* is the host name of the server on which the JumpStart directory is located. *jumpstart_dir_path* is the absolute path to the JumpStart directory.

`-p` *sysid_server*:*path*

Specifies the path to the `sysidcfg` file for preconfiguring system information. *sysid_server* is either a valid host name or an IP address for the server that contains the file. *path* is the absolute path to the directory containing the `sysidcfg` file.

`-t` *boot_image_path*

Specifies the path to an alternate boot image if you want to use a boot image other than the one in the `Tools` directory on the Solaris 10 net installation image, CD, or DVD.

-b *"boot-property=value"*
: **x86 based systems only:** Enables you to set the value of a boot property variable that you want to use to boot the client from the network. The -b option must be used with the -e option.

  See the eeprom(1M) man page for descriptions of boot properties.

-e *ethernet_address*
: Specifies the Ethernet address of the client that you want to install. This option enables you to set up the installation information to use for a specific client.

  For more information about client-specific installations by using DHCP, see "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 80.

client_name
: Is the name of the system to be installed from the network. This name is *not* the host name of the install server.

platform_group
: Is the platform group of the system to be installed. For more information, see "Platform Names and Groups" on page 35.

Example 6–6 SPARC: Adding a SPARC Install Client on a SPARC Install Server When Using DHCP (DVD)

The following example illustrates how to add an install client when you are using DHCP to set installation parameters on the network. The install client is named basil, which is an Ultra™ 5 system. The file system /export/home/dvdsparc/Solaris_10/Tools contains the add_install_client command.

For more information on how to use DHCP to set installation parameters for network installations, see Preconfiguring System Configuration Information With the DHCP Service (Tasks).

```
sparc_install_server# cd /export/home/dvdsparc/Solaris_10/Tools
sparc_install_server# ./add_install_client -d basil sun4u
```

Example 6–7 Adding an Install Client That Is On the Same Subnet As Its Server (DVD)

The following example illustrates how to add an install client that is on the same subnet as the install server. The install client is named basil, which is an Ultra 5 system. The file system /export/home/dvdsparc/ contains the add_install_client command.

```
install_server# cd /export/home/dvdsparc/Solaris_10/Tools
install_server# ./add_install_client basil sun4u
```

Example 6–8 Adding an Install Client to a Boot Server (DVD)

The following example illustrates how to add an install client to a boot server. The install client is named rose, which is an Ultra 5 system. Run the command on the boot server. The -s option is used to specify an install server that is named rosemary, which contains a Solaris 10 Operating System for SPARC Platforms DVD image in /export/home/dvdsparc.

```
boot_server# cd /export/home/dvdsparc/Solaris_10/Tools
boot_server# ./add_install_client -s rosemary:/export/home/dvdsparc rose sun4u
```

Example 6–9 x86: Adding an x86 Install Client on an x86 Install Server When Using DHCP (DVD)

The following example illustrates how to add an x86 install client to an install server when you are using DHCP to set installation parameters on the network. The -d option is used to specify that clients are to use the DHCP protocol for configuration. If you plan to use PXE network boot, you must use the DHCP protocol. The DHCP class name SUNW.i86pc indicates that this command applies to all Solaris x86 network boot clients, not just a single client. The -s option is used to specify that the clients are to be installed from the install server that is named rosemary. This server contains a Solaris 10 Operating System for x86 Platforms DVD image in /export/home/dvdx86.

For more information on how to use DHCP to set installation parameters for network installations, see Preconfiguring System Configuration Information With the DHCP Service (Tasks).

```
x86_install_server# cd /export/boot/dvdx86/Solaris_10/Tools
x86_install_server# ./add_install_client -d -s rosemary:/export/home/dvdx86 \
SUNW.i86pc i86pc
```

Example 6–10 x86: Specifying a Serial Console to Use During a Network Installation (DVD)

The following example illustrates how to add an x86 install client to an install server and specify a serial console to use during the installation. This example sets up the install client in the following manner.

- The -d option indicates that the client is set up to use DHCP to set installation parameters.
- The -e option indicates that this installation will occur only on the client with the Ethernet address 00:07:e9:04:4a:bf.
- The first and second uses of the -b option instruct the installation program to use the serial port ttya as an input and an output device.

```
install server# cd /export/boot/dvdx86/Solaris_10/Tools
install server# ./add_install_client -d -e "00:07:e9:04:4a:bf" \
-b "input-device=ttya" -b "output-device=ttya" i86pc
```

For a complete description of the boot property variables and values you can use with the `-b` option, see the `eeprom`(1M) man page.

Example 6–11 x86: Specifying a Boot Device to Use During a Network Installation (DVD)

The following example illustrates how to add an x86 install client to an install server and specify a boot device to use during the installation. If you specify the boot device when you set up the install client, you are not prompted for this information by the Device Configuration Assistant during the installation.

This example sets up the install client in the following manner.

- The `-d` option indicates that the client is set up to use DHCP to set installation parameters.
- The `-e` option indicates that this installation will occur only on the client with the Ethernet address 00:07:e9:04:4a:bf.
- The first and second uses of the `-b` option instruct the installation program to use the serial port `ttya` as an input and an output device.
- The third use of the `-b` option instructs the installation program to use a specific boot device during the installation.

  Note – The value of the boot device path varies based on your hardware.

- The i86pc platform name indicates that the client is an x86 based system.

```
install server# cd /export/boot/dvdx86/Solaris_10/Tools
install server# ./add_install_client -d -e "00:07:e9:04:4a:bf" \
-b "input-device=ttya" -b "output-device=ttya" \
-b "bootpath=/pci@0,0/pci108e,16a8@8" i86pc
```

For a complete description of the boot property variables and values you can use with the `-b` option, see the `eeprom`(1M) man page.

More Information Continuing the Installation

After you add your system as an installation client, you are ready to install your system from the network. For information, see "Booting and Installing the System From the Network With a DVD Image" on page 114.

See Also For additional information about the `add_install_client` command, see `install_scripts`(1M).

Booting and Installing the System From the Network With a DVD Image

After you add the system as an installation client, you can install the client from the network. This section describes the following tasks.

- "SPARC: To Boot the Client Over the Network" on page 114
- "x86: To Boot the Client Over the Network" on page 115

▼ SPARC: To Boot the Client Over the Network

Before You Begin

This procedure assumes that you have completed the following tasks.

- Set up an install server. For instructions about how to create an install server from CD media, see "x86: To Create an x86 Install Server" on page 289.
- Set up a boot server or a DHCP server, if necessary. If the system you want to install is on a different subnet than the installation server, you must set up a boot server, or use a DHCP server. For instructions about how to set up a DHCP server to support network installations, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.
- Gathered or preconfigured the information you need to install. You can perform this task in one or more of the following ways.
  - Gather the information in Checklist for Installation.
  - Create a `sysidcfg` file if you use a `sysidcfg` file to preconfigure system information. For information about how to create a `sysidcfg` file, see "Preconfiguring With the `sysidcfg` File" on page 57.
  - Set up a name server if you use a name service to preconfigure system information. For information about how to preconfigure information with a name service, see "Preconfiguring With the Name Service" on page 75.
  - Create a profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method. For information about how to set up a custom JumpStart installation, see Chapter 4, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

Steps

1. **Turn on the client system.**

   If the system is currently running, bring the system to run level 0.

   The `ok` prompt is displayed.

2. **Boot the system from the network.**

- **To install with the Solaris interactive installation GUI, type the following command.**

  ```
  ok boot net - install
  ```

- **To install with the Solaris interactive text installer in a desktop session, type the following command.**

  ```
  ok boot net - text
  ```

- **To install with the Solaris interactive text installer in a console session, type the following command.**

  ```
  ok boot net - nowin
  ```

The system boots from the network.

3. **If you are prompted, answer the system configuration questions.**

   - If you preconfigured all of the system information, the installation program does not prompt you to enter any configuration information. See Chapter 4 for more information.
   - If you did not preconfigure all the system information, use the "Checklist for Installation" on page 41 to help you answer the configuration questions.

   If you are using the GUI, after you confirm the system configuration information, the Welcome to Solaris dialog box appears.

See Also For information about how to complete an interactive installation with the Solaris installation GUI, see "To Install or Upgrade With the Solaris Installation Program" in *Solaris 10 Installation Guide: Basic Installations*.

▼ x86: To Boot the Client Over the Network

To install the system over the network, you must instruct the client system to boot over the network. Enable network boot on the client system by using the BIOS setup program in the system BIOS, the network adapter BIOS, or both. On some systems, you must also adjust the boot device priority list so that network boot is attempted before booting from other devices. See the manufacturer's documentation for each setup program, or watch for setup program instructions during boot.

Before You Begin This procedure assumes that you have completed the following tasks.

- Set up an install server. For instructions about how to create an install server from CD media, see "x86: To Create an x86 Install Server" on page 289.
- Set up a boot server or a DHCP server, if necessary. If the system you want to install is on a different subnet than the installation server, you must set up a boot server, or use a DHCP server. For instructions about how to set up a DHCP server to support network installations, see "Preconfiguring System Configuration

Information With the DHCP Service (Tasks)" on page 78.

- Gathered or preconfigured the information you need to install. You can perform this task in one or more of the following ways.
  - Gather the information in Checklist for Installation.
  - Create a `sysidcfg` file if you use a `sysidcfg` file to preconfigure system information. For information about how to create a `sysidcfg` file, see "Preconfiguring With the `sysidcfg` File" on page 57.
  - Set up a name server if you use a name service to preconfigure system information. For information about how to preconfigure information with a name service, see "Preconfiguring With the Name Service" on page 75.
  - Create a profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method. For information about how to set up a custom JumpStart installation, see Chapter 4, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

This procedure also assumes that your system can boot from the network. If your system cannot boot from the network, you must create a boot diskette to install over the network. See "x86: Copying the Boot Software to a Diskette" on page 285 for information about how to create a boot diskette.

Steps

1. **Turn on the system.**
2. **Type the appropriate keystroke combination to enter the system BIOS.**
3. **In the system BIOS, instruct the system to boot from the network.**

   See your hardware documentation for information about how to set the boot priority in the BIOS.
4. **Exit the BIOS.**

   The system boots from the network.
5. **When prompted, select an installation type.**
   - **To install with the Solaris interactive installation GUI, type 1 and Enter.**
   - **To perform a custom JumpStart installation, type 2 and Enter.**
   - **To install with the Solaris interactive text installer in a desktop session, type 3 and Enter.**
   - **To install with the Solaris interactive text installer in a console session, type 4 and Enter.**

   The installation program begins.
6. **If you are prompted, answer the system configuration questions.**

- If you preconfigured all of the system information, the installation program does not prompt you to enter any configuration information. See Chapter 4 for more information.
- If you did not preconfigure all the system information, use the "Checklist for Installation" on page 41 to help you answer the configuration questions.

If you are using the installation GUI, after you confirm the system configuration information, the Welcome to Solaris dialog box appears.

7. **After the system boots and installs over the network, instruct the system to boot from the disk drive on subsequent boots.**

See Also For information about how to complete an interactive installation with the Solaris installation GUI, see "To Install or Upgrade With the Solaris Installation Program" in *Solaris 10 Installation Guide: Basic Installations*.

CHAPTER 7

Preparing to Install From the Network With CD Media (Tasks)

This chapter describes how to use CD media to set up your network and systems to install the Solaris software from the network. Network installations enable you to install the Solaris software from a system that has access to the Solaris 10 disc images, called an install server, to other systems on the network. You copy the contents of the CD media to the install server's hard disk. Then, you can install the Solaris software from the network by using any of the Solaris installation methods. This chapter covers the following topics:

- "Task Map: Preparing to Install From the Network With CD Media" on page 120
- "SPARC: Creating a SPARC Install Server With CD Media" on page 121
- "x86: Creating an x86 Install Server With CD Media" on page 126
- "Creating a Cross-Platform Install Server for CD Media" on page 131
- "Creating a Boot Server on a Subnet With a CD Image" on page 141
- "Adding Systems to Be Installed From the Network With a CD Image" on page 143
- "Booting and Installing the System From the Network With a CD Image" on page 148

Task Map: Preparing to Install From the Network With CD Media

TABLE 7–1 Task Map: Setting Up an Install Server With CD Media

Task	Description	For Instructions
Create an install server.	Use the `setup_install_server`(1M) command to copy the Solaris 10 Software - 1 CD to the install server's hard disk. Use the `add_to_install_server`(1M) command to copy additional Solaris 10 Software CDs and the Solaris 10 Languages CD to the install server's hard disk.	■ "SPARC: Creating a SPARC Install Server With CD Media" on page 121 ■ "x86: To Create an x86 Install Server With x86 CD Media" on page 126 ■ "Creating a Cross-Platform Install Server for CD Media" on page 131
(Optional) Create boot servers.	If you want to install systems from the network that are not on the same subnet as the install server, you must create a boot server on the subnet to boot the systems. If you are using Dynamic Host Configuration Protocol (DHCP), a boot server is not necessary.	"Creating a Boot Server on a Subnet With a CD Image" on page 141
Add systems to be installed from the network.	Use the `add_install_client` command to set up each system that you want to install from the network. Each system that you want to install needs to find the install server, the boot server if required, and configuration information on the network.	"Adding Systems to Be Installed From the Network With a CD Image" on page 143
Install the system over the network.	Begin the installation by booting the system from the network.	"Booting and Installing the System From the Network With a CD Image" on page 148

SPARC: Creating a SPARC Install Server With CD Media

The install server contains the installation image needed to install systems from the network. You must create an install server to install the Solaris software on a system from the network. You do not always need to set up a separate boot server.

- If you are using DHCP to set installation parameters or your install server and client are on the same subnet, you do not need a separate boot server.
- If your install server and your client are not on the same subnet and you are not using DHCP, you must create separate boot servers for each subnet. You could create an install server for each subnet; however, install servers require more disk space.

▼ SPARC: To Create a SPARC Install Server With SPARC CD Media

This procedure creates a SPARC install server with SPARC CD media.

If you want to create an install server by using media of a platform different from the install server, for example, a SPARC system with x86 CD media, see "Creating a Cross-Platform Install Server for CD Media" on page 131.

Note – This procedure assumes that the system is running the Volume Manager. If you are not using the Volume Manager to manage media, refer to *System Administration Guide: Devices and File Systems* for detailed information about managing removable media without the Volume Manager.

Steps

1. **On the system that is to become the install server, become superuser.**

   The system must include a CD-ROM drive and be part of the site's network and name service. If you use a name service, the system must already be in a name service, such as NIS, NIS+, DNS, or LDAP. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **Insert the Solaris 10 Software - 1 CD in the system's drive.**

3. **Create a directory for the CD image.**

   ```
   # mkdir -p install_dir_path
   ```

   install_dir_path Specifies the directory where the CD image is to be copied

4. **Change to the `Tools` directory on the mounted disc.**

   ```
   # cd /cdrom/cdrom0/s0/Solaris_10/Tools
   ```

   In the previous example, **`cdrom0`** is the path to the drive that contains the Solaris OS CD media.

5. **Copy the image in the drive to the install server's hard disk.**

   ```
   # ./setup_install_server install_dir_path
   ```

   install_dir_path Specifies the directory where the CD image is to be copied

   ---

   Note – The `setup_install_server` command indicates whether you have enough disk space available for the Solaris 10 Software disc images. To determine available disk space, use the `df -kl` command.

   ---

6. **Decide if you need to make the install server available for mounting.**

   - **If the install server is on the same subnet as the system to be installed or you are using DHCP, you do not need to create a boot server. Proceed to Step 7.**
   - **If the install server is not on the same subnet as the system to be installed and you are not using DHCP, complete the following steps.**

     a. **Verify that the path to the install server's image is shared appropriately.**

     ```
     # share | grep install_dir_path
     ```

     install_dir_path Specifies the path to the installation image where the CD image was copied

     - **If the path to the install server's directory is displayed and anon=0 is displayed in the options, proceed to Step 7.**
     - **If the path to the install server's directory is not displayed or you do not have anon=0 in the options, continue.**

     b. **Make the install server available by adding this entry to the `/etc/dfs/dfstab` file.**

     ```
     share -F nfs -o ro,anon=0 -d "install server directory" install_dir_path
     ```

     c. **Verify that the `nfsd` daemon is running.**

     - **If the install server is running the Solaris 10 OS, or compatible version, type the following command.**

       ```
       # svcs -l svc:/network/nfs/server:default
       ```

If the nfsd daemon is online, continue to Step d. If the nfsd daemon is not online, start it.

```
# svcadm enable svc:/network/nfs/server
```

- **If the install server is running the Solaris 9 OS, or compatible version, type the following command.**

  ```
  # ps -ef | grep nfsd
  ```

  If the nfsd daemon is running, continue to Step d. If the nfsd daemon is not running, start it.

  ```
  # /etc/init.d/nfs.server start
  ```

d. **Share the install server.**

```
# shareall
```

7. **Change directories to root (/).**

```
# cd /
```

8. **Eject the Solaris 10 Software - 1 CD.**

9. **Insert the Solaris 10 Software - 2 CD in the system's CD-ROM drive.**

10. **Change to the Tools directory on the mounted CD.**

```
# cd /cdrom/cdrom0/Solaris_10/Tools
```

11. **Copy the CD in the CD-ROM drive to the install server's hard disk.**

```
# ./add_to_install_server install_dir_path
```

install_dir_path Specifies the directory where the CD image is to be copied

12. **Change directories to root (/).**

```
# cd /
```

13. **Eject the Solaris 10 Software - 2 CD.**

14. **Repeat Step 9 through Step 13 for each Solaris 10 Software CD that you want to install.**

15. **Insert the Solaris 10 Languages CD in the system's CD-ROM drive.**

16. **Change to the Tools directory on the mounted CD.**

```
# cd /cdrom/cdrom0/Tools
```

17. **Copy the CD in the CD-ROM drive to the install server's hard disk.**

```
# ./add_to_install_server install_dir_path
```

install_dir_path Specifies the directory where the CD image is to be copied

18. **Change directories to root (/).**

```
# cd /
```

19. **Decide if you want to patch the files that are located in the miniroot (/*install_dir_path*/Solaris_10/Tools/Boot) on the net install image that was created by setup_install_server. Patching a file might be necessary if a boot image has problems.**

    - **If no, continue.**
    - **If yes, use the patchadd -C command to patch the files that are located in the miniroot.**

    Caution – Don't use the patchadd -C command unless you have read the Patch README instructions or have contacted your local Sun support office.

20. **Decide if you need to create a boot server.**

    - **If you are using DHCP or the install server is on the same subnet as the system to be installed, you do not need to create a boot server. Proceed to "Adding Systems to Be Installed From the Network With a CD Image" on page 143.**
    - **If you are *not* using DHCP and the install server and the client are on a different subnet, you must create a boot server. Proceed to "Creating a Boot Server on a Subnet With a CD Image" on page 141.**

Example 7–1 SPARC: Creating a SPARC Install Server With SPARC CD Media

The following example illustrates how to create an install server by copying the following CDs to the install server's /export/home/cdsparc directory. This example assumes that the install server is running the Solaris 10 OS.

- Solaris 10 Software for SPARC Platforms CDs
- Solaris 10 Languages for SPARC Platforms CD

Insert the Solaris 10 Software for SPARC Platforms - 1 CD in the system's CD-ROM drive.

```
# mkdir -p /export/home/cdsparc
# cd /cdrom/cdrom0/s0/Solaris_10/Tools
# ./setup_install_server /export/home/cdsparc
```

- If you have a separate boot server, add these steps.
  1. Add the following path to the /etc/dfs/dfstab file.

```
share -F nfs -o ro,anon=0 -d "install server directory" \
/export/home/cdsparc
```

2. Check if the nfsd daemon is online. If the nfsd daemon is not online, start it and share it.

   ```
   # svcs -l svc:/network/nfs/server:default
   # svcadm enable svc:/network/nfs/server
   # shareall
   ```

3. Continue with the following steps.

- If you do not need a boot server or have completed the steps for a separate boot server, continue.

```
# cd /
```

Eject the Solaris 10 Software for SPARC Platforms - 1 CD. Insert the Solaris 10 Software for SPARC Platforms - 2 CD in the CD-ROM drive.

```
# cd /cdrom/cdrom0/Solaris_10/Tools
# ./add_to_install_server /export/home/cdsparc
# cd /
```

Repeat the previous commands for each Solaris 10 Software CD that you want to install.

Insert the Solaris 10 Languages for SPARC Platforms CD in the CD-ROM drive.

```
# cd /cdrom/cdrom0/Tools
# ./add_to_install_server /export/home/cdsparc
```

More Information

Continuing the Installation

After you set up the install server, you must add the client as an installation client. For information about how to add client systems to install over the network, see "Adding Systems to Be Installed From the Network With a CD Image" on page 143.

If you are not using DHCP, and your client system is on a different subnet than your install server, you must create a boot server. For more information, see "Creating a Boot Server on a Subnet With a CD Image" on page 141.

See Also For additional information about the setup_install_server and the add_to_install_server commands, see install_scripts(1M).

x86: Creating an x86 Install Server With CD Media

The install server contains the installation image needed to install systems from the network. You must create an install server to install the Solaris software on a system from the network. You do not always need to set up a separate boot server.

- If you are using DHCP to set installation parameters or your install server and client are on the same subnet, you do not need a separate boot server.
- If your install server and your client are not on the same subnet and you are not using DHCP, you must create separate boot servers for each subnet. You could create an install server for each subnet; however, install servers require more disk space.

▼ x86: To Create an x86 Install Server With x86 CD Media

This procedure creates an x86 install server with x86 CD media.

If you want to create an install server by using media of a platform different from the install server, for example, an x86 system with SPARC CD media, see "Creating a Cross-Platform Install Server for CD Media" on page 131.

Note – This procedure assumes that the system is running the Volume Manager. If you are not using the Volume Manager to manage media, refer to *System Administration Guide: Devices and File Systems* for detailed information about managing removable media without the Volume Manager.

Steps

1. **On the system that is to become the install server, become superuser.**

   The system must include a CD-ROM drive and be part of the site's network and name service. If you use a name service, the system must already be in a name service, such as NIS, NIS+, DNS, or LDAP. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **Insert the Solaris 10 Software - 1 CD in the system's drive.**

3. **Create a directory for the CD image.**

   ```
   # mkdir -p install_dir_path
   ```

install_dir_path Specifies the directory where the CD image is to be copied

4. **Change to the `Tools` directory on the mounted disc.**

   ```
   # cd /cdrom/cdrom0/s2/Solaris_10/Tools
   ```

   In the previous example, **`cdrom0`** is the path to the drive that contains the Solaris OS CD media.

5. **Copy the image in the drive to the install server's hard disk.**

   ```
   # ./setup_install_server install_dir_path
   ```

   install_dir_path Specifies the directory where the CD image is to be copied

   ---

   Note – The `setup_install_server` command indicates whether you have enough disk space available for the Solaris 10 Software disc images. To determine available disk space, use the `df -kl` command.

   ---

6. **Decide if you need to make the install server available for mounting.**

   - **If the install server is on the same subnet as the system to be installed or you are using DHCP, you do not need to create a boot server. Proceed to Step 7.**

   - **If the install server is not on the same subnet as the system to be installed and you are not using DHCP, complete the following steps.**

     a. **Verify that the path to the install server's image is shared appropriately.**

     ```
     # share | grep install_dir_path
     ```

     install_dir_path Specifies the path to the installation image where the CD image was copied

     - **If the path to the install server's directory is displayed and anon=0 is displayed in the options, proceed to Step 7.**

     - **If the path to the install server's directory is not displayed or you do not have anon=0 in the options, continue.**

     b. **Make the install server available by adding this entry to the `/etc/dfs/dfstab` file.**

     ```
     share -F nfs -o ro,anon=0 -d "install server directory" install_dir_path
     ```

     c. **Verify that the `nfsd` daemon is running.**

     - **If the install server is running the Solaris 10 OS, or compatible version, type the following command.**

       ```
       # svcs -l svc:/network/nfs/server:default
       ```

If the `nfsd` daemon is online, continue to Step d. If the `nfsd` daemon is not online, start it.

```
# svcadm enable svc:/network/nfs/server
```

- **If the install server is running the Solaris 9 OS, or compatible version, type the following command.**

  ```
  # ps -ef | grep nfsd
  ```

  If the `nfsd` daemon is running, continue to Step d. If the `nfsd` daemon is not running, start it.

  ```
  # /etc/init.d/nfs.server start
  ```

d. **Share the install server.**

```
# shareall
```

7. **Change directories to root (/).**

```
# cd /
```

8. **Eject the Solaris 10 Software - 1 CD.**

9. **Insert the Solaris 10 Software - 2 CD in the system's CD-ROM drive.**

10. **Change to the `Tools` directory on the mounted CD:**

```
# cd /cdrom/cdrom0/Solaris_10/Tools
```

11. **Copy the CD in the CD-ROM drive to the install server's hard disk.**

```
# ./add_to_install_server install_dir_path
```

install_dir_path Specifies the directory where the CD image is to be copied

12. **Change directories to root (/).**

```
# cd /
```

13. **Eject the Solaris 10 Software - 2 CD.**

14. **Repeat Step 9 through Step 13 for each Solaris 10 Software CD that you want to install.**

15. **Insert the Solaris 10 Languages CD in the system's CD-ROM drive.**

16. **Change to the `Tools` directory on the mounted CD:**

```
# cd /cdrom/cdrom0/Tools
```

17. **Copy the CD in the CD-ROM drive to the install server's hard disk.**

```
# ./add_to_install_server install_dir_path
```

install_dir_path Specifies the directory where the CD image is to be copied

18. **Change directories to root (/).**

```
# cd /
```

19. **Decide if you want to patch the files that are located in the miniroot (/*install_dir_path*/Solaris_10/Tools/Boot) on the net install image that was created by setup_install_server. Patching a file might be necessary if a boot image has problems.**

    - **If no, continue.**
    - **If yes, use the patchadd -C command to patch the files that are located in the miniroot.**

> **Caution –** Don't use the patchadd -C command unless you have read the Patch README instructions or have contacted your local Sun support office.

20. **Decide if you need to create a boot server.**

    - **If you are using DHCP or the install server is on the same subnet as the system to be installed, you do not need to create a boot server. Proceed to "Adding Systems to Be Installed From the Network With a CD Image" on page 143.**
    - **If you are *not* using DHCP and the install server and the client are on a different subnet, you must create a boot server. Proceed to "Creating a Boot Server on a Subnet With a CD Image" on page 141.**

Example 7–2 x86: Creating an x86 Install Server With x86 CD Media

The following example illustrates how to create an install server by copying the following CDs to the install server's /export/home/cdx86 directory. This example assumes that the install server is running the Solaris 10 OS.

- Solaris 10 Software for x86 Platforms CDs
- Solaris 10 Languages for x86 Platforms CD

Insert the Solaris 10 Software for x86 Platforms - 1 CD in the system's CD-ROM drive.

```
# mkdir -p /export/home/cdx86
# cd /cdrom/cdrom0/s2/Solaris_10/Tools
# ./setup_install_server /export/home/cdx86
```

- If you have a separate boot server, add these steps.
    1. Add the following path to the /etc/dfs/dfstab file:

```
share -F nfs -o ro,anon=0 -d "install server directory" \
/export/home/cdx86
```

2. Check if the nfsd daemon is online. If the nfsd daemon is not online, start it and share it.

   ```
   # svcs -l svc:/network/nfs/server:default
   # svcadm enable svc:/network/nfs/server
   # shareall
   ```

3. Continue with the following steps.

- If you do not need a boot server or have completed the steps for a separate boot server, continue with the following steps.

```
# cd /
```

Eject the Solaris 10 Software for x86 Platforms - 1 CD. Insert the Solaris 10 Software for x86 Platforms - 2 CD in the CD-ROM drive.

```
# cd /cdrom/cdrom0/Solaris_10/Tools
# ./add_to_install_server /export/home/cdx86
# cd /
```

Repeat the previous commands for each Solaris 10 Software CD that you want to install.

Insert the Solaris 10 Languages for x86 Platforms CD in the CD-ROM drive.

```
# cd /cdrom/cdrom0/Tools
# ./add_to_install_server /export/home/cdx86
```

More Information

Continuing the Installation

After you set up the install server, you must add the client as an installation client. For information about how to add client systems to install over the network, see "Adding Systems to Be Installed From the Network With a CD Image" on page 143.

If you are not using DHCP, and your client system is on a different subnet than your install server, you must create a boot server. For more information, see "Creating a Boot Server on a Subnet With a CD Image" on page 141.

See Also

For additional information about the setup_install_server and the add_to_install_server commands, see install_scripts(1M).

Creating a Cross-Platform Install Server for CD Media

If you need to use a CD of a platform different from the install server, you cannot read the CD in the install server. You need a remote system to read the CD. For example, if you are setting up a SPARC install server and need to use x86 CD media, you need a remote x86 system to read the CDs.

▼ To Create an x86 Install Server on a SPARC System With x86 CD Media

Use this procedure to create an x86 install server on a SPARC system with x86 CD media.

In this procedure, *SPARC-system* is the SPARC system that is to be the install server and *remote-x86-system* is the remote x86 system to be used with the x86 CD media.

Before You Begin

You need the following items to perform this task.

- A SPARC system
- An x86 system with a CD-ROM drive
- A set of CDs for the remote x86 system
  - Solaris 10 Software for x86 Platforms CDs
  - Solaris 10 Languages for x86 Platforms CD

Note – This procedure assumes that the system is running the Volume Manager. If you are not using the Volume Manager to manage media, refer to *System Administration Guide: Devices and File Systems* for detailed information about managing removable media without the Volume Manager.

Steps

1. **On the remote x86 system, become superuser.**

   The system must include a CD-ROM drive and be part of the site's network and name service. If you use a name service, the system must also be in the NIS, NIS+, DNS, or LDAP name service. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **On the remote x86 system, insert the Solaris 10 Software for x86 Platforms - 1 CD into the system's drive.**

3. **On the remote x86 system, add the following entries to the `/etc/dfs/dfstab` file.**

   ```
   share -F nfs -o ro,anon=0 /cdrom/cdrom0/s0
   share -F nfs -o ro,anon=0 /cdrom/cdrom0/s2
   ```

4. **On the remote x86 system, start the NFS daemon.**

   - **If the install server is running the Solaris 10 OS, or compatible version, type the following command.**

     ```
     remote-x86-system# svcadm enable svc:/network/nfs/server
     ```

   - **If the install server is running the Solaris 9 OS, or compatible version, type the following command.**

     ```
     remote-x86-system# /etc/init.d/nfs.server start
     ```

5. **On the remote x86 system, verify that the CD is available to other systems by using the share command.**

   ```
   remote-x86-system# share
   -   /cdrom/sol_10_x86/s0 ro,anon=0 " "
   -   /cdrom/sol_10_x86/s2 ro,anon=0 " "
   ```

   In the previous sample output, `sol_10_x86` refers to the Solaris 10 OS on x86 based systems. This text string varies for each version of the Solaris OS.

6. **On the SPARC system that is to be the x86 install server, become superuser.**

7. **On the SPARC system, access the x86 CD by creating two directories for the appropriate mount points, one for the miniroot and one for the product.**

   ```
   SPARC-system# mkdir directory_name_s0
   ```

   ```
   SPARC-system# mkdir  directory_name_s2
   ```

   directory_name_s0 Is the name of the directory to contain the miniroot from slice 0

   directory_name_s2 Is the name of the directory to contain the product from slice 2

8. **Verify that the CD is properly exported on the remote x86 system.**

   ```
   SPARC-system# showmount -e remote-x86-system
   export list for remote-x86-system:
   /cdrom/sol_10_x86/s0 (everyone)
   /cdrom/sol_10_x86/s2 (everyone)
   ```

9. **On the SPARC system, mount the remote x86 CD image.**

```
SPARC-system# mount remote_x86_system_name:/cdrom/sol_10_x86/s0 directory_name_s0
```

```
SPARC-system# mount remote_x86_system_name:/cdrom/sol_10_x86/s2 directory_name_s2
```

10. **On the SPARC system, change to the `Tools` directory on the mounted disc:**

    ```
    SPARC-system# cd /directory_name_s2/Solaris_10/Tools
    ```

11. **On the SPARC system, copy the disc in the drive to the install server's hard disk in the directory you've created by using the `setup_install_server` command:**

    ```
    SPARC-system# ./setup_install_server -t directory_name_s0  install_dir_path
    ```

`-t`	Specifies the path to a boot image if you want to use a boot image other than the one in the `Tools` directory on the Solaris 10 Software - 2 CD.
directory_name_s0	Is the name of the directory that contains the miniroot from slice 0.
install_dir_path	Specifies the directory where the disc image is to be copied. The directory must be empty.

    Note – The `setup_install_server` command indicates whether you have enough disk space available for the Solaris 10 Software disc images. To determine available disk space, use the `df -kl` command.

12. **On the SPARC system, change to the top directory.**

    ```
    SPARC-system# cd /
    ```

13. **On the SPARC system, unmount both directories.**

    ```
    SPARC-system# unmount directory_name_s0
    ```

    ```
    SPARC-system# unmount directory_name_s2
    ```

14. **On the x86 system, unshare both CD-ROM slices.**

    ```
    remote x86  system# unshare /cdrom/sol_10_x86/s0
    ```

    ```
    remote x86  system# unshare /cdrom/sol_10_x86/s2
    ```

15. **On the x86 system, eject the Solaris 10 Software for x86 Platforms - 1 CD.**

16. **Insert the Solaris 10 Software for x86 Platforms - 2 CD into the SPARC system's CD-ROM drive.**

17. **On the SPARC system, change to the `Tools` directory on the mounted CD:**

    ```
    SPARC-system# cd /cdrom/cdrom0/Solaris_10/Tools
    ```

18. **On the SPARC system, copy the CD to the install server's hard disk:**

    ```
    SPARC-system# ./add_to_install_server install_dir_path
    ```

install_dir_path	Specifies the directory where the CD image is to be copied

19. **Eject the Solaris 10 Software for x86 Platforms - 2 CD.**

20. **Repeat Step 16 through Step 19 for each Solaris 10 Software CD you want to install.**

21. **On the SPARC system, insert the Solaris 10 Languages for x86 Platforms CD into the SPARC system's CD-ROM drive and mount the CD.**

22. **On the SPARC system, change to the `Tools` directory on the mounted CD:**

    ```
    SPARC-system# cd /cdrom/cdrom0/Tools
    ```

23. **On the SPARC system, copy the CD to the install server's hard disk:**

    ```
    SPARC-system# ./add_to_install_server install_dir_path
    ```

install_dir_path	Specifies the directory where the CD image is to be copied

24. **Decide if you want to patch the files that are located in the miniroot (`Solaris_10/Tools/Boot`) on the net installation image that was created by `setup_install_server`.**

    - **If no, proceed to the next step.**

    - **If yes, use the `patchadd -C` command to patch the files that are located in the miniroot.**

    Caution – Don't use the `patchadd -C` command unless you have read the `Patch README` instructions or have contacted your local Sun support office.

    ---

25. **Decide if you need to create a boot server.**

    - **If the install server is on the same subnet as the system to be installed or you are using DHCP, you do not need to create a boot server. See "Adding Systems to Be Installed From the Network With a CD Image" on page 143.**

    - **If the install server is not on the same subnet as the system to be installed and you are not using DHCP, you must create a boot server. For detailed instructions on how to create a boot server, refer to "To Create a Boot Server on a Subnet With a CD Image" on page 141.**

Example 7–3 Creating an x86 Install Server on a SPARC System With x86 CD Media

The following example illustrates how to create an x86 install server on a SPARC system that is named `rosemary`. The following x86 CDs are copied from a remote x86 system that is named `tadpole` to the SPARC install server's `/export/home/cdx86` directory.

- Solaris 10 Software for x86 Platforms CDs
- Solaris 10 Languages for x86 Platforms CD

This example assumes that the install server is running the Solaris 10 OS.

On the remote x86 system:

```
tadpole (remote-x86-system)# share -F nfs -o ro,anon=0 /cdrom/cdrom0/s0
tadpole (remote-x86-system)# share -F nfs -o ro,anon=0 /cdrom/cdrom0/s2
tadpole (remote-x86-system)# svcadm enable svc:/network/nfs/server
```

On the SPARC system:

```
rosemary (SPARC-system)# mkdir /x86S0
rosemary (SPARC-system)# mkdir /x86S2
rosemary (SPARC-system)# mount tadpole:/cdrom/sol_10_x86/s0 /x86S0
rosemary (SPARC-system)# mount tadpole:/cdrom/sol_10_x86/s0 /x86S2
rosemary (SPARC-system)# cd /x86S2/Solaris_10/Tools
rosemary (SPARC-system)# ./setup_install_server -t /x86S0 /export/home/cdx86
rosemary (SPARC-system)# cd /
rosemary (SPARC-system)# unmount /x86S0
rosemary (SPARC-system)# unmount /x86S2
```

On the remote x86 system:

```
tadpole (remote-x86-system) unshare /cdrom/cdrom0/s0
tadpole (remote-x86-system) unshare  /cdrom/cdrom0/s2
```

On the SPARC system:

```
rosemary (SPARC-system)# cd /cdrom/cdrom0/Solaris_10/Tools
rosemary (SPARC-system)# ./add_to_install_server /export/home/cdx86
```

Repeat the previous commands for each Solaris 10 Software CD that you want to install.

```
rosemary (SPARC-system)# cd /cdrom/cdrom0/Tools
rosemary (SPARC-system)# ./add_to_install_server /export/home/cdx86
```

In this example, each CD is inserted and automatically mounted before each of the commands. After each command, the CD is removed.

More Information

Continuing the Installation

After you set up the install server, you must add the client as an installation client. For information about how to add client systems to install over the network, see "Adding Systems to Be Installed From the Network With a CD Image" on page 143.

If you are not using DHCP, and your client system is on a different subnet than your install server, you must create a boot server. For more information, see "Creating a Boot Server on a Subnet With a CD Image" on page 141.

See Also

For additional information about the `setup_install_server` and the `add_to_install_server` commands, see `install_scripts`(1M).

▼ To Create a SPARC Install Server on an x86 System With SPARC CD Media

Use this procedure to create a SPARC install server on an x86 system with SPARC CD media.

In this procedure, *x86-system* is the x86 system that is to be the install server and *remote-SPARC-system* is the remote SPARC system to be used with the SPARC CD media.

Before You Begin

You need the following items to perform this task.

- An x86 system
- A SPARC system with a CD-ROM drive
- A set of CDs for the remote SPARC system
  - Solaris 10 Software for SPARC Platforms CDs
  - Solaris 10 Languages for SPARC Platforms CD

Note – This procedure assumes that the system is running the Volume Manager. If you are not using the Volume Manager to manage media, refer to *System Administration Guide: Devices and File Systems* for detailed information about managing removable media without the Volume Manager.

Steps

1. **On the remote SPARC system, become superuser.**

   The system must include a CD-ROM drive and be part of the site's network and name service. If you use a name service, the system must also be in the NIS, NIS+, DNS, or LDAP name service. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **On the remote SPARC system, insert the Solaris 10 Software for SPARC Platforms - 1 CD into the system's drive.**

3. **On the remote SPARC system, add the following entries to the `/etc/dfs/dfstab` file.**

   ```
   share -F nfs -o ro,anon=0 /cdrom/cdrom0/s0
   share -F nfs -o ro,anon=0 /cdrom/cdrom0/s1
   ```

4. **On the remote SPARC system, start the NFS daemon.**

   - **If the install server is running the Solaris 10 OS, or compatible version, type the following command.**

     ```
     remote-SPARC-system# svcadm enable svc:/network/nfs/server
     ```

   - **If the install server is running the Solaris 9 OS, or compatible version, type the following command.**

     ```
     remote-SPARC-system# /etc/init.d/nfs.server start
     ```

5. **On the remote SPARC system, verify that the CD is available to other systems by using the share command.**

   ```
   remote-SPARC-system# share
   -   /cdrom/cdrom0/s0 ro,anon=0 " "
   -   /cdrom/cdrom0/s1 ro,anon=0 " "
   ```

6. **On the x86 system that is to be the SPARC install server, become superuser.**

7. **On the x86 system, access the SPARC CD by creating two directories for the appropriate mount points, one for the miniroot and one for the product.**

   ```
   x86-system# mkdir directory_name_s0
   ```

   ```
   x86-system# mkdir  directory_name_s1
   ```

   directory_name_s0 Is the name of the directory to contain the product from slice 0

   directory_name_s1 Is the name of the directory to contain the miniroot from slice 1

8. **Verify that the CD is properly exported on the remote x86 system.**

   ```
   x86-system# showmount -e remote-SPARC-system
   export list for remote-SPARC-system:
   /cdrom/sol_10_sparc/s0 (everyone)
   /cdrom/sol_10_sparc/s1 (everyone)
   ```

9. **On the x86 system, mount the remote SPARC CD image.**

```
x86-system# mount remote_SPARC_system_name:/cdrom/cdrom0/s0 directory_name_s0
```

```
x86-system# mount remote_SPARC_system_name:/cdrom/cdrom0/s1 directory_name_s1
```

10. **On the x86 system, change to the `Tools` directory on the mounted disc:**

    ```
    x86-system# cd /directory_name_s0/Solaris_10/Tools
    ```

11. **On the x86 system, copy the disc in the drive to the install server's hard disk in the directory you've created by using the `setup_install_server` command:**

    ```
    x86-system# ./setup_install_server -t directory_name_s1 install_dir_path
    ```

    `-t` Specifies the path to a boot image if you want to use a boot image other than the one in the `Tools` directory on the Solaris 10 Software - 2 CD.

    directory_name_s1 Is the name of the directory that contains the miniroot from slice 1.

    install_dir_path Specifies the directory where the disc image is to be copied. The directory must be empty.

    Note – The `setup_install_server` command indicates whether you have enough disk space available for the Solaris 10 Software disc images. To determine available disk space, use the `df -kl` command.

    ---

12. **On the x86 system, change to the top directory.**

    ```
    x86-system# cd /
    ```

13. **On the x86 system, unmount both directories.**

    ```
    x86-system# unmount directory_name_s0
    ```

    ```
    x86-system# unmount directory_name_s1
    ```

14. **On the SPARC system, unshare both CD-ROM slices.**

    ```
    remote-SPARC-system# unshare /cdrom/cdrom0/s0
    ```

    ```
    remote-SPARC-system# unshare /cdrom/cdrom0/s2
    ```

15. **On the SPARC system, eject the Solaris 10 Software for SPARC Platforms - 1 CD.**

16. **Insert the Solaris 10 Software for SPARC Platforms - 2 CD into the x86 system's CD-ROM drive.**

17. **On the x86 system, change to the `Tools` directory on the mounted CD:**

    ```
    x86-system# cd /cdrom/cdrom0/Solaris_10/Tools
    ```

18. **On the x86 system, copy the CD to the install server's hard disk:**

    ```
    x86-system# ./add_to_install_server install_dir_path
    ```

install_dir_path Specifies the directory where the CD image is to be copied

19. **Eject the Solaris 10 Software for SPARC Platforms - 2 CD.**

20. **Repeat Step 16 through Step 19 for each Solaris 10 Software CD you want to install.**

21. **On the x86 system, insert the Solaris 10 Languages for SPARC Platforms CD into the x86 system's CD-ROM drive and mount the CD.**

22. **On the x86 system, change to the `Tools` directory on the mounted CD:**

```
x86-system# cd /cdrom/cdrom0/Tools
```

23. **On the x86 system, copy the CD to the install server's hard disk:**

```
x86-system# ./add_to_install_server install_dir_path
```

install_dir_path Specifies the directory where the CD image is to be copied

24. **Decide if you want to patch the files that are located in the miniroot (`Solaris_10/Tools/Boot`) on the net install image that was created by `setup_install_server`.**

    - **If no, proceed to the next step.**
    - **If yes, use the `patchadd -C` command to patch the files that are located in the miniroot.**

Caution – Don't use the `patchadd -C` unless you have read the `Patch` README instructions or have contacted your local Sun support office.

25. **Decide if you need to create a boot server.**

    - **If the install server is on the same subnet as the system to be installed or you are using DHCP, you do not need to create a boot server. See "Adding Systems to Be Installed From the Network With a CD Image" on page 143.**
    - **If the install server is not on the same subnet as the system to be installed and you are not using DHCP, you must create a boot server. For detailed instructions on how to create a boot server, refer to "To Create a Boot Server on a Subnet With a CD Image" on page 141.**

Example 7–4 Creating a SPARC Install Server on an x86 System With SPARC CD Media

The following example illustrates how to create a SPARC install server on an x86 system that is named `richards`. The following SPARC CDs are copied from a remote SPARC system that is named `simpson` to the x86 install server's `/export/home/cdsparc` directory.

- Solaris 10 Software for SPARC Platforms CDs
- Solaris 10 Languages for SPARC Platforms CD

This example assumes that the install server is running the Solaris 10 OS.

On the remote SPARC system:

```
simpson (remote-SPARC-system)# share -F nfs -o ro,anon=0 /cdrom/cdrom0/s0
simpson (remote-SPARC-system)# share -F nfs -o ro,anon=0 /cdrom/cdrom0/s1
simpson (remote-SPARC-system)# svcadm enable svc:/network/nfs/server
```

On the x86 system:

```
richards (x86-system)# mkdir /sparcS0
richards (x86-system)# mkdir /sparcS1
richards (x86-system)# mount simpson:/cdrom/cdrom0/s0 /sparcS0
richards (x86-system)# mount simpson:/cdrom/cdrom0/s1 /sparcS1
richards (x86-system)# cd /sparcS0/Solaris_10/Tools
richards (x86-system)# ./setup_install_server -t /sparcS0 /export/home/cdsparc
richards (x86-system)# cd /
richards (x86-system)# unmount /sparcS0
richards (x86-system)# unmount /sparcS1
```

On the remote SPARC system:

```
simpson (remote-SPARC-system) unshare /cdrom/cdrom0/s0
simpson (remote-SPARC-system) unshare  /cdrom/cdrom0/s1
```

On the x86 system:

```
richards (x86-system)# cd /cdrom/cdrom0/Solaris_10/Tools
richards (x86-system)# ./add_to_install_server /export/home/cdsparc
```

Repeat the previous commands for each Solaris 10 Software for x86 Platforms CD that you want to install.

```
richards (x86-system)# cd /cdrom/cdrom0/Tools
richards (x86-system)# ./add_to_install_server /export/home/cdsparc
```

In this example, each CD is inserted and automatically mounted before each of the commands. After each command, the CD is removed.

More Information

Continuing the Installation

After you set up the install server, you must add the client as an installation client. For information about how to add client systems to install over the network, see "Adding Systems to Be Installed From the Network With a CD Image" on page 143.

If you are not using DHCP, and your client system is on a different subnet than your install server, you must create a boot server. For more information, see "Creating a Boot Server on a Subnet With a CD Image" on page 141.

See Also For additional information about the `setup_install_server` and the `add_to_install_server` commands, see `install_scripts`(1M).

Creating a Boot Server on a Subnet With a CD Image

You must create an install server to install the Solaris software on a system from the network. You do not always need to set up a boot server. A boot server contains enough of the boot software to boot systems from the network, and then the install server completes the installation of the Solaris software.

- If you are using DHCP to set installation parameters or your install server and client are on the same subnet, you do not need a boot server. Proceed to "Adding Systems to Be Installed From the Network With a CD Image" on page 143.
- If your install server and your client are not on the same subnet and you are not using DHCP, you must create separate boot servers for each subnet. You could create an install server for each subnet; however, install servers require more disk space.

▼ To Create a Boot Server on a Subnet With a CD Image

Steps

1. **On the system you intend to make the boot server for the subnet, log in and become superuser.**

   The system must include a local CD-ROM drive or have access to the remote Solaris 10 disc images, which are normally on the install server. If you use a name service, the system should be in the name service. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **Mount the Solaris 10 Software - 1 CD image from the install server.**

   ```
   # mount -F nfs -o ro server_name:path /mnt
   ```

   server_name:*path* Is the install server name and absolute path to the disc image

3. **Create a directory for the boot image.**

   ```
   # mkdir -p boot_dir_path
   ```

   boot_dir_path Specifies the directory where the boot software is to be copied

4. **Change to the `Tools` directory on the Solaris 10 Software - 1 CD image.**

   ```
   # cd /mnt/Solaris_10/Tools
   ```

5. **Copy the boot software to the boot server.**

   ```
   # ./setup_install_server -b boot_dir_path
   ```

   `-b` Specifies to set up the system as a boot server

   boot_dir_path Specifies the directory where the boot software is to be copied

   Note – The `setup_install_server` command indicates whether you have enough disk space available for the images. To determine available disk space, use the `df -kl` command.

6. **Change directories to root (/).**

   ```
   # cd /
   ```

7. **Unmount the installation image.**

   ```
   # umount /mnt
   ```

Example 7–5 Creating a Boot Server on a Subnet With CD Media

The following example illustrates how to create a boot server on a subnet. These commands copy the boot software from the Solaris 10 Software for SPARC Platforms - 1 CD image to `/export/install/boot` on the system's local disk.

```
# mount -F nfs -o ro crystal:/export/install/boot /mnt
# mkdir -p /export/install/boot
# cd /mnt/Solaris_10/Tools
# ./setup_install_server -b /export/install/boot
# cd /
# umount /mnt
```

In this example, the disc is inserted and automatically mounted before the command. After the command, the disc is removed.

More Information

Continuing the Installation

After you set up the boot server, you must add the client as an installation client. For information about how to add client systems to install over the network, see "Adding Systems to Be Installed From the Network With a CD Image" on page 143.

See Also

For additional information about the setup_install_server command, see install_scripts(1M).

Adding Systems to Be Installed From the Network With a CD Image

After you create an install server and, if necessary, a boot server, you must set up each system that you want to install from the network. Each system that you want to install needs to find the following:

- An install server
- A boot server if it is required
- The sysidcfg file if you use a sysidcfg file to preconfigure system information
- A name server if you use a name service to preconfigure system information
- The profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method

Use the following add_install_client procedure for setting up install servers and clients. Also, see the example procedures for the following:

- If you are using DHCP to set installation parameters, see Example 7–6.
- If your install server and client are on the same subnet, see Example 7–7.
- If your install server and your client are not on the same subnet and you are not using DHCP, see Example 7–8.
- If you are using DHCP to set installation parameters for x86 clients, see Example 7–9.
- If you want to use a specific serial port to display output during the installation of an x86 based system, see Example 7–10.
- If you want to set up an x86 client to use a specific network interface during the installation, see Example 7–11.

For more options to use with this command, see the man page, add_install_client(1M).

▼ To Add Systems to Be Installed From the Network With `add_install_client` (CDs)

If you have a boot server, make sure you have shared the install server installation image. See the procedure “To Create an Install Server,” Step 6.

Steps

1. **On the install server or boot server, become superuser.**

2. **If you use the NIS, NIS+, DNS, or LDAP name service, verify that the following information about the system to be installed has been added to the name service:**

   - Host name
   - IP address
   - Ethernet address

   For more information on name services, see *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

3. **Change to the `Tools` directory on the Solaris 10 CD image on the install server:**

   ```
   # cd /install_dir_path/Solaris_10/Tools
   ```

   install_dir_path Specifies the path to the `Tools` directory

4. **Set up the client system to be installed from the network.**

   ```
   # ./add_install_client -d -s install_server:install_dir_path \
   -c jumpstart_server:jumpstart_dir_path    -p sysid_server:path \
     -t boot_image_path -b "network_boot_variable=value" \
   -e ethernet_address client_name platform_group
   ```

   `-d`
   Specifies that the client is to use DHCP to obtain the network install parameters. If you use the `-d` only, the `add_install_client` command sets up the installation information for client systems of the same class, for example, all SPARC client machines. To set up the installation information for a specific client, use the `-d` with the `-e` option.

   For x86 clients, use this option to boot the systems from the network by using PXE network boot.

   For more information about class-specific installations by using DHCP, see “Creating DHCP Options and Macros for Solaris Installation Parameters” on page 80.

   `-s` *install_server:install_dir_path*
   Specifies the name and path to the install server.

   - *install_server* is the host name of the install server
   - *install_dir_path* is the absolute path to the Solaris 10 CD image

`-c` *jumpstart_server*`:`*jumpstart_dir_path*
Specifies a JumpStart directory for custom JumpStart installations. *jumpstart_server* is the host name of the server on which the JumpStart directory is located. *jumpstart_dir_path* is the absolute path to the JumpStart directory.

`-p` *sysid_server*`:`*path*
Specifies the path to the `sysidcfg` file for preconfiguring system information. *sysid_server* is either a valid host name or an IP address for the server that contains the file. *path* is the absolute path to the directory containing the `sysidcfg` file.

`-t` *boot_image_path*
Specifies the path to an alternate boot image if you want to use a boot image other than the one in the Tools directory on the Solaris 10 net installation image, CD, or DVD.

`-b` *"boot-property=value"*
x86 based systems only: Enables you to set the value of a boot property variable that you want to use to boot the client from the network. The `-b` must be used with the `-e` option.

See the `eeprom`(1M) man page for descriptions of boot properties.

`-e` *ethernet_address*
Specifies the Ethernet address of the client that you want to install. This option enables you to set up the installation information to use for a specific client.

For more information about client-specific installations by using DHCP, see "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 80.

client_name
Is the name of the system to be installed from the network. This name is *not* the host name of the install server.

platform_group
Is the platform group of the system to be installed. A detailed list of platform groups appears in "Platform Names and Groups" on page 35.

Example 7–6 SPARC: Adding a SPARC Install Client on a SPARC Install Server When Using DHCP (CDs)

The following example illustrates how to add an install client when you are using DHCP to set installation parameters on the network. The install client is named `basil`, which is an Ultra 5 system. The file system `/export/home/cdsparc/Solaris_10/Tools` contains the `add_install_client` command.

For more information on how to use DHCP to set installation parameters for network installations, see Preconfiguring System Configuration Information With the DHCP Service (Tasks).

```
sparc_install_server# cd /export/home/cdsparc/Solaris_10/Tools
sparc_install_server# ./add_install_client -d basil sun4u
```

Example 7–7 Adding an Install Client That Is on the Same Subnet as Its Server (CDs)

The following example illustrates how to add an install client that is on the same subnet as the install server. The install client is named basil, which is an Ultra 5 system. The file system /export/home/cdsparc/Solaris_10/Tools contains the add_install_client command.

```
install_server# cd /export/home/cdsparc/Solaris_10/Tools
install_server# ./add_install_client basil sun4u
```

Example 7–8 Adding an Install Client to a Boot Server (CDs)

The following example illustrates how to add an install client to a boot server. The install client is named rose, which is an Ultra 5 system. Run the command on the boot server. The -s option is used to specify an install server that is named rosemary, which contains a Solaris 10 CD image in /export/home/cdsparc.

```
boot_server# cd /export/home/cdsparc/Solaris_10/Tools
boot_server# ./add_install_client -s rosemary:/export/home/cdsparc rose sun4u
```

Example 7–9 x86: Adding an x86 Install Client on an x86 Install Server When Using DHCP (CDs)

The following example illustrates how to add an x86 install client to an install server when you are using DHCP to set installation parameters on the network. The -d option is used to specify that clients are to use the DHCP protocol for configuration. If you plan to use PXE network boot, you must use the DHCP protocol. The DHCP class name SUNW.i86pc indicates that this command applies to all Solaris x86 network boot clients, not just a single client. The -s option is used to specify that the clients are to be installed from the install server that is named rosemary. This server contains a Solaris 10 Software for x86 Platforms - 1 CD image in /export/home/cdx86.

For more information on how to use DHCP to set installation parameters for network installations, see Preconfiguring System Configuration Information With the DHCP Service (Tasks).

```
install server# cd /export/boot/Solaris_10/Tools
install server# ./add_install_client -d -s rosemary:/export/home/cdx86 SUNW.\
i86pc i86pc
```

Example 7–10 x86: Specifying a Serial Console to Use During a Network Installation (CDs)

The following example illustrates how to add an x86 install client to an install server and specify a serial console to use during the installation. This example sets up the install client in the following manner.

- The -d option indicates that the client is set up to use DHCP to set installation parameters.
- The -e option indicates that this installation will occur only on the client with the Ethernet address 00:07:e9:04:4a:bf.
- The first and second uses of the -b option instruct the installation program to use the serial port ttya as an input and an output device.

```
install server# cd /export/boot/Solaris_10/Tools
install server# ./add_install_client -d -e "00:07:e9:04:4a:bf" \
-b "input-device=ttya" -b "output-device=ttya" i86pc
```

For a complete description of the boot property variables and values you can use with the -b option, see the eeprom(1M) man page.

Example 7–11 x86: Specifying a Boot Device to Use During a Network Installation (CDs)

The following example illustrates how to add an x86 install client to an install server and specify a boot device to use during the installation. If you specify the boot device when you set up the install client, you are not prompted for this information by the Device Configuration Assistant during the installation.

This example sets up the install client in the following manner.

- The -d option indicates that the client is set up to use DHCP to set installation parameters.
- The -e option indicates that this installation will occur only on the client with the Ethernet address 00:07:e9:04:4a:bf.
- The first and second uses of the -b option instruct the installation program to use the serial port ttya as an input and an output device.
- The third use of the -b option instructs the installation program to use a specific boot device during the installation.

  Note – The value of the boot device path varies based on your hardware.

- The i86pc platform name indicates that the client is an x86 based system.

```
install server# cd /export/boot/Solaris_10/Tools
install server# ./add_install_client -d -e "00:07:e9:04:4a:bf" \
-b "input-device=ttya" -b "output-device=ttya" \
-b "bootpath=/pci@0,0/pci108e,16a8@8" i86pc
```

For a complete description of the boot property variables and values you can use with the -b option, see the eeprom(1M) man page.

More Information

Continuing the Installation

After you add your system as an installation client, you are ready to install your system from the network. For information, see "Booting and Installing the System From the Network With a CD Image" on page 148.

See Also For additional information about the `add_install_client` command, see `install_scripts`(1M).

Booting and Installing the System From the Network With a CD Image

After you add the system as an installation client, you can install the client from the network. This section describes the following tasks.

- "SPARC: To Boot the Client Over the Network" on page 148
- "x86: To Boot the Client Over the Network" on page 149

▼ SPARC: To Boot the Client Over the Network

Before You Begin This procedure assumes that you have completed the following tasks.

- Set up an install server. For instructions about how to create an install server from CD media, see "x86: To Create an x86 Install Server" on page 289.
- Set up a boot server or a DHCP server, if necessary. If the system you want to install is on a different subnet than the installation server, you must set up a boot server, or use a DHCP server. For instructions about how to set up a DHCP server to support network installations, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.
- Gathered or preconfigured the information you need to install. You can perform this task in one or more of the following ways.
  - Gather the information in Checklist for Installation.
  - Create a `sysidcfg` file if you use a `sysidcfg` file to preconfigure system information. For information about how to create a `sysidcfg` file, see "Preconfiguring With the `sysidcfg` File" on page 57.
  - Set up a name server if you use a name service to preconfigure system information. For information about how to preconfigure information with a name service, see "Preconfiguring With the Name Service" on page 75.
  - Create a profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method. For information about how to set up a custom JumpStart installation, see Chapter 4, "Preparing Custom JumpStart

Installations (Tasks)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

Steps

1. **Turn on the client system.**

   If the system is currently running, bring the system to run level 0.

   The `ok` prompt is displayed.

2. **Boot the system from the network.**

   - **To install with the Solaris interactive installation GUI, type the following command.**

     ```
     ok boot net - install
     ```

   - **To install with the Solaris interactive text installer in a desktop session, type the following command.**

     ```
     ok boot net - text
     ```

   - **To install with the Solaris interactive text installer in a console session, type the following command.**

     ```
     ok boot net - nowin
     ```

   The system boots from the network.

3. **If you are prompted, answer the system configuration questions.**

   - If you preconfigured all of the system information, the installation program does not prompt you to enter any configuration information. See Chapter 4 for more information.
   - If you did not preconfigure all the system information, use the "Checklist for Installation" on page 41 to help you answer the configuration questions.

   If you are using the GUI, after you confirm the system configuration information, the Welcome to Solaris dialog box appears.

See Also For information about how to complete an interactive installation with the Solaris installation GUI, see "To Install or Upgrade With the Solaris Installation Program" in *Solaris 10 Installation Guide: Basic Installations*.

▼ x86: To Boot the Client Over the Network

To install the system over the network, you must instruct the client system to boot over the network. Enable network boot on the client system by using the BIOS setup program in the system BIOS, the network adapter BIOS, or both. On some systems, you must also adjust the boot device priority list so that network boot is attempted before booting from other devices. See the manufacturer's documentation for each setup program, or watch for setup program instructions during boot.

Before You Begin

This procedure assumes that you have completed the following tasks.

- Set up an install server. For instructions about how to create an install server from CD media, see "x86: To Create an x86 Install Server" on page 289.
- Set up a boot server or a DHCP server, if necessary. If the system you want to install is on a different subnet than the installation server, you must set up a boot server, or use a DHCP server. For instructions about how to set up a DHCP server to support network installations, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.
- Gathered or preconfigured the information you need to install. You can perform this task in one or more of the following ways.
  - Gather the information in Checklist for Installation.
  - Create a `sysidcfg` file if you use a `sysidcfg` file to preconfigure system information. For information about how to create a `sysidcfg` file, see "Preconfiguring With the `sysidcfg` File" on page 57.
  - Set up a name server if you use a name service to preconfigure system information. For information about how to preconfigure information with a name service, see "Preconfiguring With the Name Service" on page 75.
  - Create a profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method. For information about how to set up a custom JumpStart installation, see Chapter 4, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

This procedure also assumes that your system can boot from the network. If your system cannot boot from the network, you must create a boot diskette to install over the network. See "x86: Copying the Boot Software to a Diskette" on page 285 for information about how to create a boot diskette.

Steps

1. **Turn on the system.**
2. **Type the appropriate keystroke combination to enter the system BIOS.**
3. **In the system BIOS, instruct the system to boot from the network.**

   See your hardware documentation for information about how to set the boot priority in the BIOS.
4. **Exit the BIOS.**

   The system boots from the network.
5. **When prompted, select an installation type.**
   - **To install with the Solaris interactive installation GUI, type 1 and Enter.**
   - **To perform a custom JumpStart installation, type 2 and Enter.**

- **To install with the Solaris interactive text installer in a desktop session, type 3 and Enter.**
- **To install with the Solaris interactive text installer in a console session, type 4 and Enter.**

The installation program begins.

6. **If you are prompted, answer the system configuration questions.**
   - If you preconfigured all of the system information, the installation program does not prompt you to enter any configuration information. See Chapter 4 for more information.
   - If you did not preconfigure all the system information, use the "Checklist for Installation" on page 41 to help you answer the configuration questions.

   If you are using the installation GUI, after you confirm the system configuration information, the Welcome to Solaris dialog box appears.

7. **After the system boots and installs over the network, instruct the system to boot from the disk drive on subsequent boots.**

See Also For information about how to complete an interactive installation with the Solaris installation GUI, see "To Install or Upgrade With the Solaris Installation Program" in *Solaris 10 Installation Guide: Basic Installations*.

CHAPTER 8

Preparing to Install From the Network (Command Reference)

This chapter lists the commands used to set up network installations.

Network Installation Commands

This table describes the commands you use to install Solaris software over the network. The table also indicates to which platform the commands apply.

Command	Platform	Description
`add_install_client`	All	A command that adds network installation information about a system to an install server or boot server from the network. The `add_install_client`(1M) man page contains more information.
`setup_install_server`	All	A script that copies the Solaris 10 DVD or CDs to an install server's local disk or copies the boot software to a boot server. The `setup_install_server`(1M) man page contains more information.
(CD media only) `add_to_install_server`	All	A script that copies additional packages within a product tree on the CDs to the local disk on an existing install server. The `add_to_install_server`(1M) man page contains more information.
`mount`	All	A command that enables the mounting of file systems and shows the mounted file systems, including the file system on the Solaris 10 DVD or Solaris 10 Software and Solaris 10 Languages CDs. The `mount`(1M) man page contains more information.

Command	Platform	Description
`showmount -e`	All	A command that lists all the shared file systems that are located on a remote host. The `showmount`(1M) man page contains more information.
`uname -i`	All	A command for determining a system's platform name, for example, SUNW,Ultra-5_10, or i86pc. You might need the system's platform name when you install the Solaris software. The `uname`(1) man page contains more information.
`patchadd -C` *net_install_image*	All	A command to add patches to the files that are located in the miniroot, `Solaris_10 /Tools/Boot`, on a net installation image of a DVD or CD that is created by `setup_install_server`. This facility enables you to patch Solaris installation commands and other miniroot-specific commands. *net_install_image* is the absolute path name of the net installation image. The `patchadd`(1M) man page contains more information. **Caution –** Don't use the `patchadd -C` command unless you have read the `Patch README` instructions or have contacted your local Sun support office.
`reset`	SPARC	An Open Boot PROM command for resetting the system and rebooting the machine. Or, if you boot and see a series of error messages about I/O interrupts, press the Stop and A keys at the same time, and then type `reset` at the `ok` or `>` PROM prompt.
`banner`	SPARC	An Open Boot PROM command that displays system information, such as model name, Ethernet address, and memory installed. You can issue this command only at the `ok` or `>` PROM prompt.

PART III Installing Over a Wide Area Network

This part describes how to use the WAN boot installation method to install a system over a wide area network (WAN).

CHAPTER 9

WAN Boot (Overview)

This chapter provides an overview of the WAN boot installation method. This chapter describes the following topics.

- "What Is WAN Boot?" on page 157
- "When to Use WAN Boot" on page 158
- "How WAN Boot Works (Overview)" on page 159
- "Security Configurations Supported by WAN Boot (Overview)" on page 163

What Is WAN Boot?

The WAN boot installation method enables you to boot and install software over a wide area network (WAN) by using HTTP. By using WAN boot, you can install the Solaris OS on SPARC based systems over a large public network where the network infrastructure might be untrustworthy. You can use WAN boot with security features to protect data confidentiality and installation image integrity.

The WAN boot installation method enables you to transmit an encrypted Solaris Flash archive over a public network to a remote SPARC based client. The WAN boot programs then install the client system by performing a custom JumpStart installation. To protect the integrity of the installation, you can use private keys to authenticate and encrypt data. You can also transmit your installation data and files over a secure HTTP connection by configuring your systems to use digital certificates.

To perform a WAN boot installation, you install a SPARC based system by downloading the following information from a web server over a HTTP or secure HTTP connection.

- `wanboot` program – The `wanboot` program is the second level boot program that loads the WAN boot miniroot, client configuration files, and installation files. The `wanboot` program performs tasks similar to those that are performed by the `ufsboot` or `inetboot` second level boot programs.

- WAN boot file system – WAN boot uses several different files to configure the client and retrieve data to install the client system. These files are located in the /etc/netboot directory of the web server. The wanboot-cgi program transmits these files to the client as a file system, called the WAN boot file system.
- WAN boot miniroot – The WAN boot miniroot is a version of the Solaris miniroot that has been modified to perform a WAN boot installation. The WAN boot miniroot, like the Solaris miniroot, contains a kernel and just enough software to install the Solaris environment. The WAN boot miniroot contains a subset of the software in the Solaris miniroot.
- Custom JumpStart configuration files – To install the system, WAN boot transmits sysidcfg, rules.ok, and profile files to the client. WAN boot then uses these files to perform a custom JumpStart installation on the client system.
- Solaris Flash archive – A Solaris Flash archive is a collection of files that you copy from a master system. You can then use this archive to install a client system. WAN boot uses the custom JumpStart installation method to install a Solaris Flash archive on the client system. After you install an archive on a client system, the system contains the exact configuration of the master system.

You then install the archive on the client by using the custom JumpStart installation method.

You can protect the transfer of the previously listed information by using keys and digital certificates.

For a more detailed description of the sequence of events in a WAN boot installation, see "How WAN Boot Works (Overview)" on page 159.

When to Use WAN Boot

The WAN boot installation method enables you to install SPARC based systems that are located in geographically remote areas. You might want to use WAN boot to install remote servers or clients that are accessible only over a public network.

If you want to install systems that are located on your local area network (LAN), the WAN boot installation method might require more configuration and administration than necessary. For information about how to install systems over a LAN, see Chapter 5.

How WAN Boot Works (Overview)

WAN boot uses a combination of servers, configuration files, Common Gateway Interface (CGI) programs, and installation files to install a remote SPARC based client. This section describes the general sequence of events in a WAN boot installation.

Sequence of Events in a WAN Boot Installation

Figure 9–1 shows the basic sequence of events in a WAN boot installation. In this figure, a SPARC based client retrieves configuration data and installation files from a web server and an install server over a WAN.

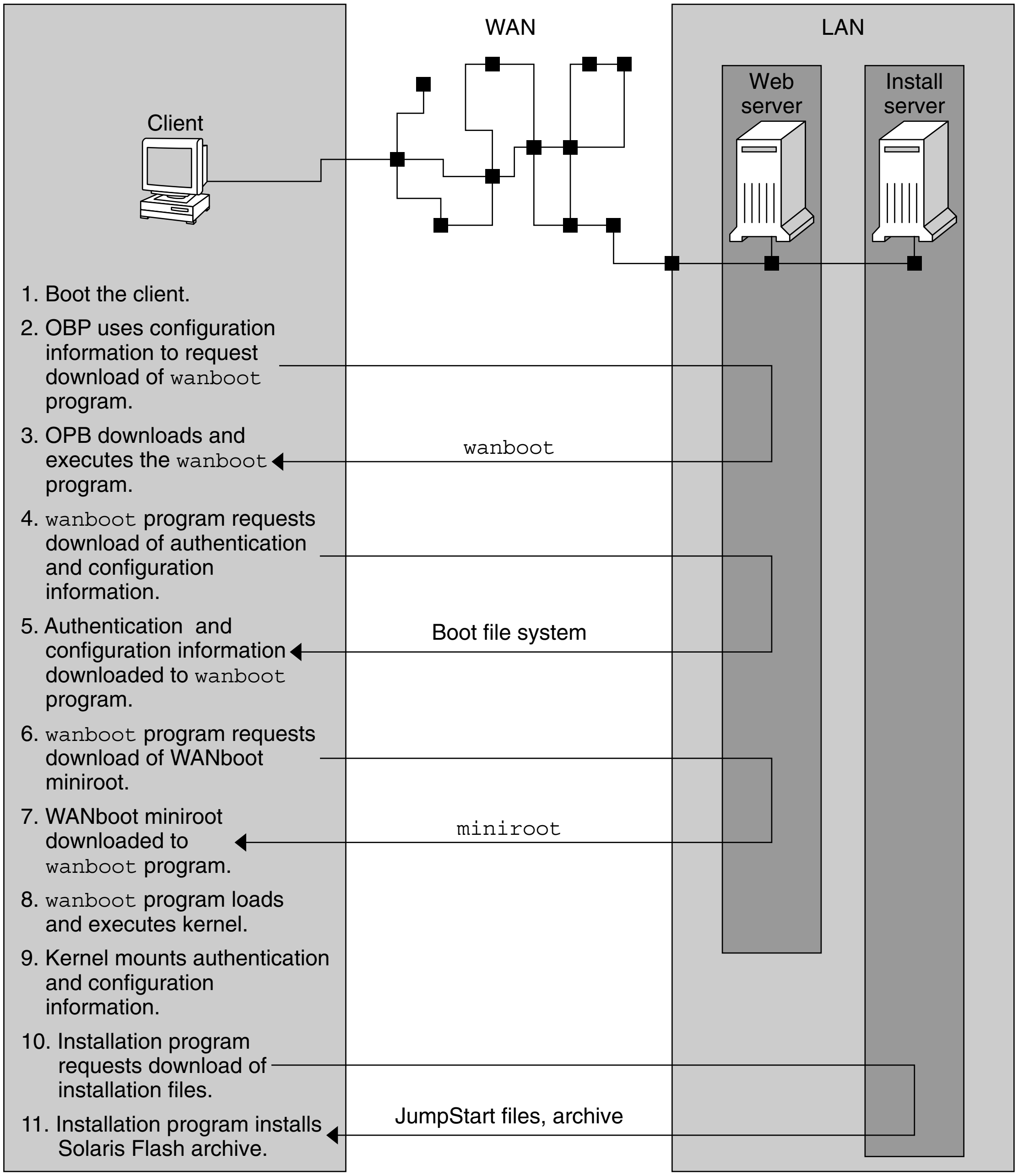
WAN
LAN
Client
Web server
Install server
1. Boot the client.
2. OBP uses configuration information to request download of wanboot program.
3. OPB downloads and executes the wanboot program.
wanboot
4. wanboot program requests download of authentication and configuration information.
5. Authentication and configuration information downloaded to wanboot program.
Boot file system
6. wanboot program requests download of WANboot miniroot.
7. WANboot miniroot downloaded to wanboot program.
miniroot
8. wanboot program loads and executes kernel.
9. Kernel mounts authentication and configuration information.
10. Installation program requests download of installation files.
11. Installation program installs Solaris Flash archive.
JumpStart files, archive

FIGURE 9–1 Sequence of Events in a WAN Boot Installation

1. You boot the client in one of the following ways.

   - Boot from the network by setting network interface variables in the Open Boot PROM (OBP).
   - Boot from the network with the DHCP option.
   - Boot from a local CD-ROM.
2. The client OBP obtains configuration information from one of the following sources.
   - From boot argument values that are typed at the command line by the user
   - From the DHCP server, if the network uses DHCP
3. The client OBP requests the WAN boot second level boot program (`wanboot`).

   The client OBP downloads the `wanboot` program from the following sources.
   - From a special web server, called the WAN boot server, by using the Hyper Text Transfer Protocol (HTTP)
   - From a local CD-ROM (not shown in the figure)
4. The `wanboot` program requests the client configuration information from the WAN boot server.
5. The `wanboot` program downloads configuration files that are transmitted by the `wanboot-cgi` program from the WAN boot server. The configuration files are transmitted to the client as the WAN boot file system.
6. The `wanboot` program requests the download of the WAN boot miniroot from the WAN boot server.
7. The `wanboot` program downloads the WAN boot miniroot from the WAN boot server by using HTTP or secure HTTP.
8. The `wanboot` program loads and executes the UNIX kernel from the WAN boot miniroot.
9. The UNIX kernel locates and mounts the WAN boot file system for use by the Solaris installation program.
10. The installation program requests the download of a Solaris Flash archive and custom JumpStart files from an install server.

    The installation program downloads the archive and custom JumpStart files over an HTTP or HTTPS connection.
11. The installation program performs a custom JumpStart installation to install the Solaris Flash archive on the client.

Protecting Data During a WAN Boot Installation

The WAN boot installation method enables you to use hashing keys, encryption keys, and digital certificates to protect your system data during the installation. This section briefly describes the different data protection methods that are supported by the WAN boot installation method.

Checking the Integrity of Data With a Hashing Key

To protect the data you transmit from the WAN boot server to the client, you can generate a Hashed Message Authentication Code (HMAC) key. You install this hashing key on both the WAN boot server and the client. The WAN boot server uses this key to sign the data to be transmitted to the client. The client then uses this key to verify the integrity of the data that is transmitted by the WAN boot server. After you install a hashing key on a client, the client uses this key for future WAN boot installations.

For instructions about how to use a hashing key, see "(Optional) To Create a Hashing Key and an Encryption Key" on page 197.

Encrypting Data With Encryption Keys

The WAN boot installation method enables you to encrypt the data you transmit from the WAN boot server to the client. You can use WAN boot utilities to create a Triple Data Encryption Standard (3DES) or Advanced Encryption Standard (AES) encryption key. You can then provide this key to both the WAN boot server and the client. WAN boot uses this encryption key to encrypt the data sent from the WAN boot server to the client. The client can then use this key to decrypt the encrypted configuration files and security files that are transmitted during the installation.

Once you install an encryption key on a client, the client uses this key for future WAN boot installations.

Your site might not permit the use of encryption keys. To determine if your site permits encryption, ask your site's security administrator. If your site permits encryption, ask your security administrator which type of encryption key, either 3DES or AES, you should use.

For instructions on how to use encryption keys, see "(Optional) To Create a Hashing Key and an Encryption Key" on page 197.

Protecting Data by Using HTTPS

WAN boot supports the use of HTTP over Secure Sockets Layer (HTTPS) to transfer data between the WAN boot server and the client. By using HTTPS, you can require the server, or both the server and the client, to authenticate themselves during the installation. HTTPS also encrypts the data that is transferred from the server to the client during the installation.

HTTPS uses digital certificates to authenticate systems that exchange data over the network. A digital certificate is a file that identifies a system, either a server or client, as a system to trust during online communication. You can request a digital certificate from an external certificate authority, or create your own certificate and certificate authority.

To enable the client to trust the server and accept data from the server, you must install a digital certificate on the server. You then instruct the client to trust this certificate. You can also require the client to authenticate itself to the servers by providing a digital certificate to the client. You can then instruct the server to accept the certificate's signer when the client presents the certificate during the installation.

To use digital certificates during the installation, you must configure your web server to use HTTPS. See your web server documentation for information about how to use HTTPS.

For information about the requirements to use digital certificates during your WAN boot installation, see "Digital Certificate Requirements" on page 173. For instructions about how to use digital certificates in your WAN boot installation, see "(Optional) To Use Digital Certificates for Server and Client Authentication" on page 195.

Security Configurations Supported by WAN Boot (Overview)

WAN boot supports varying levels of security. You can use a combination of the security features that are supported in WAN boot to meet the needs of your network. A more secure configuration requires more administration, but also protects your system data to a greater extent. For more critical systems, or those systems you want to install over a public network, you might choose the configuration in "Secure WAN Boot Installation Configuration" on page 163. For less critical systems, or systems on semi-private networks, consider the configuration that is described in "Insecure WAN Boot Installation Configuration" on page 164.

This section briefly describes the different configurations you can use to set the level of security for your WAN boot installation. The section also describes the security mechanisms that are required by these configurations.

Secure WAN Boot Installation Configuration

This configuration protects the integrity of the data exchanged between the server and client, and helps keep the contents of the exchange confidential. This configuration uses an HTTPS connection, and uses either the 3DES or AES algorithm to encrypt the client configuration files. This configuration also requires the server to authenticate itself to the client during the installation. A secure WAN boot installation requires the following security features.

- HTTPS enabled on the WAN boot server and the install server
- HMAC SHA1 hashing key on the WAN boot server and the client

- 3DES or AES encryption key for the WAN boot server and the client
- Digital certificate of a certificate authority for the WAN boot server

If you want to also require client authentication during the installation, you must also use the following security features.

- Private key for the WAN boot server
- Digital certificate for the client

For a list of the tasks that are required to install with this configuration, see Table 11–1.

Insecure WAN Boot Installation Configuration

This security configuration requires the least administration effort, but provides the least secure transfer of data from the web server to the client. You do not need to create a hashing key, encryption key, or digital certificates. You do not need to configure your web server to use HTTPS. However, this configuration transfers the installation data and files over an HTTP connection, which leaves your installation vulnerable to interception over the network.

If you want the client to check the integrity of the data that is transmitted, you can use a HMAC SHA1 hashing key with this configuration. However, the Solaris Flash archive is not protected by the hashing key. The archive is transferred insecurely between the server and the client during the installation.

For a list of the tasks that are required to install with this configuration, see Table 11–2.

CHAPTER 10

Preparing to Install With WAN Boot (Planning)

This chapter describes how to prepare your network for a WAN boot installation. This chapter describes the following topics.

- "WAN Boot Requirements and Guidelines" on page 165
- "WAN Boot Security Limitations" on page 174
- "Gathering Information for WAN Boot Installations" on page 174

WAN Boot Requirements and Guidelines

The section describes the system requirements to perform a WAN boot installation.

TABLE 10–1 System Requirements for WAN Boot Installation

System and Description	Requirements
WAN boot server – The WAN boot server is a web server that provides the `wanboot` program, the configuration and security files, and the WAN boot miniroot.	■ Operating system – Solaris 9 12/03 OS, or compatible version ■ Must be configured as web server ■ Web server software must support HTTP 1.1 ■ If you want to use digital certificates, the web server software must support HTTPS

TABLE 10–1 System Requirements for WAN Boot Installation *(Continued)*

System and Description	Requirements
Install server – The install server provides the Solaris Flash archive and custom JumpStart files that are required to install the client.	■ Available disk space – space for each Solaris Flash archive ■ Media drive – CD-ROM or DVD-ROM drive ■ Operating system – Solaris 9 12/03 OS, or compatible version If the install server is a different system than the WAN boot server, the install server must meet these additional requirements. ■ Must be configured as a web server ■ Web server software must support HTTP 1.1 ■ If you want to use digital certificates, the web server software must support HTTPS
Client system – The remote system you want to install over a WAN	■ Memory - Minimum of 256 Mbytes of RAM ■ CPU – UltraSPARC II processor minimum ■ Hard disk – At least 2 Gbytes of hard disk space ■ OBP – WAN boot-enabled PROM If the client does not have the appropriate PROM, the client must have a CD-ROM drive. To determine if your client has a WAN boot-enabled PROM, see "To Check the Client OBP for WAN Boot Support" on page 186.
(Optional) DHCP server – You can use a DHCP server to provide client configuration information.	If you are using a SunOS DHCP server, you must perform one of the following tasks. ■ Upgrade the server to be an EDHCP server. ■ Rename Sun vendor options to satisfy the eight-character limit on options. For more information about the WAN installation-specific Sun vendor options, see "(Optional) Providing Configuration Information With a DHCP Server" on page 214. If the DHCP server is on a different subnet than the client, you must configure a BOOTP relay agent. For more information about how to configure a BOOTP relay agent, see Chapter 13, "Configuring the DHCP Service (Tasks)," in *System Administration Guide: IP Services*.
(Optional) Logging server – By default, all booting and installation log messages are displayed on the client console during a WAN installation. If you want to view these messages on another system, you can specify a system to serve as a logging server.	Must be configured as web server. **Note –** If you use HTTPS during your installation, the logging server must be the same system as the WAN boot server.

TABLE 10–1 System Requirements for WAN Boot Installation *(Continued)*

System and Description	Requirements
(Optional) Proxy server – You can configure the WAN boot feature to use an HTTP proxy during the download of the installation data and files.	If the installation uses HTTPS, the proxy server must be configured to tunnel HTTPS.

Web Server Software Requirements and Guidelines

The web server software you use on your WAN boot server and install server must meet the following requirements.

- Operating system requirements – WAN boot provides a Common Gateway Interface (CGI) program (`wanboot-cgi`) that converts data and files into a specific format that the client machine expects. To perform a WAN boot installation with these scripts, the web server software must run on the Solaris 9 12/03 OS, or compatible version.
- File size limitations – Your web server software might limit the size of the files you can transmit over HTTP. Check your web server documentation to make sure the software can transmit files that are the size of a Solaris Flash archive.
- SSL support – If you want to use HTTPS in your WAN boot installation, the web server software must support SSL version 3.

Server Configuration Options

You can customize the configuration of the servers that are required by WAN boot to meet your network needs. You can host all the servers on one system, or place the servers on multiple systems.

- **Single server** – If you want to centralize the WAN boot data and files on one system, you can host all the servers on the same machine. You can administer all your different servers on one system, and you only need to configure one system as a web server. However, a single server might not be able to support the volume of traffic that is required for a large number of simultaneous WAN boot installations.
- **Multiple servers** – If you want to distribute the installation data and files across your network, you can host these servers on multiple machines. You might set up a central WAN boot server, and configure multiple install servers to host Solaris Flash archives across your network. If you host the install server and logging server on independent machines, you must configure those servers as web servers.

Storing Installation and Configuration Files in the Document Root Directory

The wanboot-cgi program transmits the following files during a WAN boot installation.

- wanboot program
- WAN boot miniroot
- Custom JumpStart files
- Solaris Flash archive

To enable the wanboot-cgi program to transmit these files you must store these files in a directory that is accessible to the web server software. One way to make these files accessible is to place these files in the *document root* on your web server.

The document root, or primary document directory, is the directory on your web server where you store files you want to make available to clients. You can name and configure this directory in your web server software. See your web server documentation for more information about setting up the document root directory on your web server.

You might want to create different subdirectories of the document root directory to store your different installation and configuration files. For example, you might want to create specific subdirectories for each group of clients that you want to install. If you plan to install several different releases of the Solaris OS across your network, you might create subdirectories for each release.

Figure 10–1 shows a basic sample structure for a document root directory. In this example, the WAN boot server and install server are on the same machine. The server is running the Apache web server software.

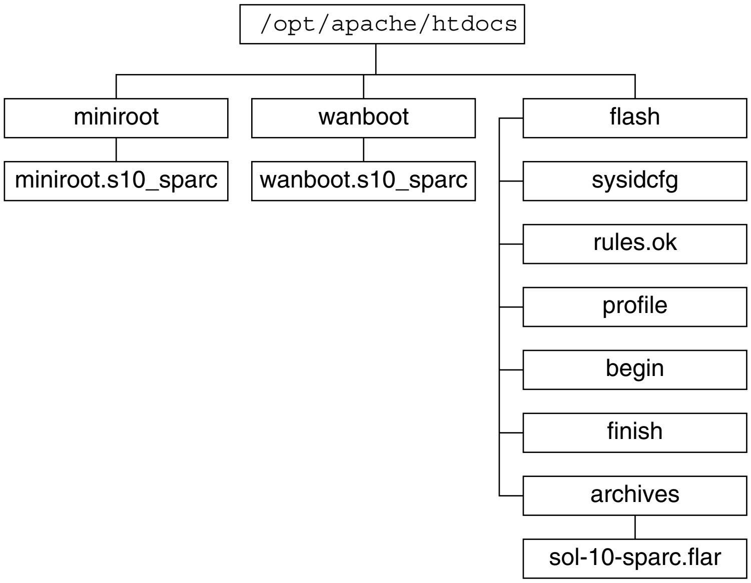

FIGURE 10–1 Sample Structure for Document Root Directory

This sample document directory uses the following structure.

- The /opt/apache/htdocs directory is the document root directory.
- The WAN boot miniroot (miniroot) directory contains the WAN boot miniroot.
- The wanboot directory contains the wanboot program.
- The Solaris Flash (flash) directory contains the custom JumpStart files that are required to install the client and the subdirectory archives. The archives directory contains the Solaris 10 Flash archive.

Note – If the WAN boot server and the install server are different systems, you might want to store the flash directory on the install server. Ensure that these files and directories are accessible to the WAN boot server.

For information about how to create the document root directory, see your web server documentation. For detailed instructions about how to create and store these installation files, see “Creating the Custom JumpStart Installation Files” on page 200.

Storing Configuration and Security Information in the `/etc/netboot` Hierarchy

The `/etc/netboot` directory contains the configuration information, private key, digital certificate, and certificate authority that are required for a WAN boot installation. This section describes the files and directories you can create in the `/etc/netboot` directory to customize your WAN boot installation.

Customizing the Scope of the WAN Boot Installation

During the installation, the `wanboot-cgi` program searches for the client information in the `/etc/netboot` directory on the WAN boot server. The `wanboot-cgi` program converts this information into the WAN boot file system, and then transmits the WAN boot file system to the client. You can create subdirectories within the `/etc/netboot` directory to customize the scope of the WAN installation. Use the following directory structures to define how configuration information is shared among the clients that you want to install.

- **Global configuration** – If you want all the clients on your network to share configuration information, store the files that you want to share in the `/etc/netboot` directory.
- **Network-specific configuration** – If you want only those machines on a specific subnet to share configuration information, store the configuration files that you want to share in a subdirectory of `/etc/netboot`. Have the subdirectory follow this naming convention.

  `/etc/netboot/`*net-ip*

  In this example, *net-ip* is the IP address of the client's subnet. For example, if you want all systems on the subnet with the IP address of 192.168.255.0 to share configuration files, create a `/etc/netboot/192.168.255.0` directory. Then, store the configuration files in this directory.
- **Client-specific configuration** – If you want only a specific client to use the boot file system, store the boot file system files in a subdirectory of `/etc/netboot`. Have the subdirectory follow this naming convention.

  `/etc/netboot/`*net-ip*`/`*client-ID*

  In this example, *net-ip* is the IP address of the subnet. *client-ID* is either the client ID that is assigned by the DHCP server, or a user-specified client ID. For example, if you want a system with the client ID 010003BA152A42 on the subnet 192.168.255.0 to use specific configuration files, create a `/etc/netboot/192.168.255.0/010003BA152A42` directory. Then, store the appropriate files in this directory.

Specifying Security and Configuration Information in the /etc/netboot Directory

You specify the security and configuration information by creating the following files and storing the files in the /etc/netboot directory.

- wanboot.conf – This file specifies the client configuration information for a WAN boot installation.
- System configuration file (system.conf) – This system configuration file specifies the location of the client's sysidcfg file and custom JumpStart files.
- keystore – This file contains the client's HMAC SHA1 hashing key, 3DES or AES encryption key, and SSL private key.
- truststore – This file contains the digital certificates of certificate signing authorities that the client should trust. These trusted certificates instruct the client to trust the server during the installation.
- certstore – This file contains the client's digital certificate.

Note – The certstore file must be located in the client ID directory. See "Customizing the Scope of the WAN Boot Installation" on page 170 for more information about subdirectories of the /etc/netboot directory.

For detailed instructions on how to create and store these files, see the following procedures.

- "To Create the System Configuration File" on page 208
- "To Create the wanboot.conf File" on page 210
- "(Optional) To Create a Hashing Key and an Encryption Key" on page 197
- "(Optional) To Use Digital Certificates for Server and Client Authentication" on page 195

Sharing Security and Configuration Information in the /etc/netboot Directory

To install clients on your network, you might want to share security and configuration files among several different clients, or across entire subnets. You can share these files by distributing your configuration information throughout the /etc/netboot/*net-ip*/*client-ID*, /etc/netboot/*net-ip*, and /etc/netboot directories. The wanboot-cgi program searches these directories for the configuration information that best fits the client, and uses that information during the installation.

The wanboot-cgi program searches for client information in the following order.

1. /etc/netboot/*net-ip*/*client-ID* – The wanboot-cgi program first checks for configuration information that is specific to the client machine. If the /etc/netboot/*net-ip*/*client-ID* directory contains all the client configuration

information, the wanboot-cgi program does not check for configuration information elsewhere in the /etc/netboot directory.

2. /etc/netboot/*net-ip* – If all the required information is not located in the /etc/netboot/*net-ip*/*client-ID* directory, the wanboot-cgi program then checks for subnet configuration information in the /etc/netboot/*net-ip* directory.
3. /etc/netboot – If the remaining information is not located in the /etc/netboot/*net-ip* directory, the wanboot-cgi program then checks for global configuration information in the /etc/netboot directory.

Figure 10–2 demonstrates how you can set up the /etc/netboot directory to customize your WAN boot installations.

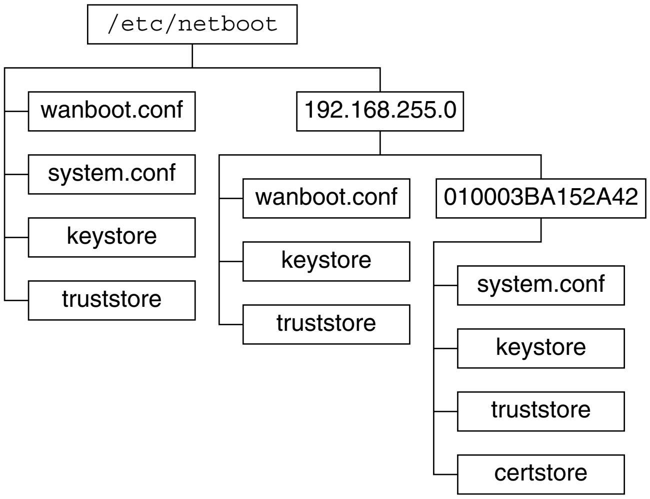

FIGURE 10–2 Sample /etc/netboot Directory

The /etc/netboot directory layout in Figure 10–2 enables you to perform the following WAN boot installations.

- When you install the client 010003BA152A42, the wanboot-cgi program uses the following files in the /etc/netboot/192.168.255.0/010003BA152A42 directory.
  - system.conf
  - keystore
  - truststore
  - certstore

  The wanboot-cgi program then uses the wanboot.conf file in the /etc/netboot/192.168.255.0 directory.
- When you install a client that is located on the 192.168.255.0 subnet, the wanboot-cgi program uses the wanboot.conf, keystore, and truststore files in the /etc/netboot/192.168.255.0 directory. The wanboot-cgi

program then uses the system.conf file in the /etc/netboot directory.

- When you install a client machine that is not located on the 192.168.255.0 subnet, the wanboot-cgi program uses the following files in the /etc/netboot directory.
  - wanboot.conf
  - system.conf
  - keystore
  - truststore

Storing the wanboot-cgi Program

The wanboot-cgi program transmits the data and files from the WAN boot server to the client. You must ensure that this program is in a directory on the WAN boot server that is accessible to the client. One method to make this program accessible to the client is to store this program in the cgi-bin directory of the WAN boot server. You might need to configure your web server software to use the wanboot-cgi program as a CGI program. See your web server documentation for information about CGI program requirements.

Digital Certificate Requirements

If you want to add security to your WAN boot installation, you can use digital certificates to enable server and the client authentication. WAN boot can use a digital certificate to establish the identity of the server or the client during an online transaction. Digital certificates are issued by a certificate authority (CA). These certificates contain a serial number, expiration dates, a copy of the certificate holder's public key, and the certificate authority's digital signature.

If you want to require server or both client and server authentication during your installation, you must install digital certificates on the server. Follow these guidelines when you use digital certificates.

- If you want to use digital certificates, the digital certificates must be formatted as part of a Public-Key Cryptography Standards #12 (PKCS#12) file.
- If you create your own certificates, you must create the certificates as PKCS#12 files.
- If you receive your certificates from third-party certificate authorities, request your certificates in the PKCS#12 format.

For detailed instructions on how to use PKCS#12 certificates during your WAN boot installation, see "(Optional) To Use Digital Certificates for Server and Client Authentication" on page 195.

WAN Boot Security Limitations

While WAN boot provides several different security features, WAN boot does not address these potential insecurities.

- **Denial of service (DoS) attacks** – A denial of service attack can take many forms, with the goal of preventing users from accessing a specific service. A DoS attack can overwhelm a network with large amounts of data, or aggressively consume limited resources. Other DoS attacks manipulate the data that is transmitted between systems in transit. The WAN boot installation method does not protect servers or clients from DoS attacks.
- **Corrupted binaries on the servers** – The WAN boot installation method does not check the integrity of the WAN boot miniroot or the Solaris Flash archive before you perform your installation. Before you perform your installation, check the integrity of your Solaris binaries against the Solaris Fingerprint Database at http://sunsolve.sun.com.
- **Encryption key and hashing key privacy** – If you use encryption keys or a hashing key with WAN boot, you must type the key value on the command line during your installation. Follow the precautions that are necessary for your network to make sure that these key values remain private.
- **Compromise of the network name service** – If you use a name service on your network, check the integrity of your name servers before you perform your WAN boot installation.

Gathering Information for WAN Boot Installations

You need to gather a wide variety of information to configure your network for a WAN boot installation. You might want to write down this information as you prepare to install over a WAN.

Use the following worksheets to record the WAN boot installation information for your network.

- Table 10–2
- Table 10–3

TABLE 10–2 Worksheet for Collecting Server Information

Information Needed	Notes
Install server information ■ Path to the WAN boot miniroot on install server ■ Path to the custom JumpStart files on the install server	
WAN boot server information ■ Path to the `wanboot` program on the WAN boot server ■ URL of the `wanboot-cgi` program on the WAN boot server ■ Path to the client's subdirectory in the `/etc/netboot` hierarchy on the WAN boot server ■ (Optional) File name of the PKCS#12 certificate file ■ (Optional) Host names of any machines other than the WAN boot server that are required for WAN installation ■ (Optional) IP address and TCP port number of the network's proxy server	
Optional server information ■ URL of the `bootlog-cgi` script on logging server ■ IP address and TCP port number of the network's proxy server	

TABLE 10–3 Worksheet for Collecting Client Information

Information	Notes
IP address for the client's subnet	
IP address for the client's router	
IP address of the client	
Subnet mask for the client	
Host name for the client	
MAC address of the client	

CHAPTER 11

Preparing to Install With WAN Boot (Tasks)

This chapter describes the following tasks that are necessary to prepare your network for a WAN boot installation.

- "Preparing to Install Over a Wide Area Network (Task Maps)" on page 177
- "Configuring the WAN Boot Server" on page 181
- "Creating the Custom JumpStart Installation Files" on page 200
- "Creating the Configuration Files" on page 208
- "(Optional) Providing Configuration Information With a DHCP Server" on page 214
- "(Optional) To Configure the WAN Boot Logging Server" on page 193

Preparing to Install Over a Wide Area Network (Task Maps)

The following tables list the tasks you need to perform to prepare for a WAN boot installation.

- For a list of the tasks you need to perform to prepare for a secure WAN boot installation, see Table 11–1.

  For a description of a secure WAN boot installation over HTTPS, see "Secure WAN Boot Installation Configuration" on page 163.
- For a list of the tasks you need to perform to prepare for an insecure WAN boot installation, see Table 11–2.

  For a description of an insecure WAN boot installation, see "Insecure WAN Boot Installation Configuration" on page 164.

To use a DHCP server or a logging server, complete the optional tasks that are listed at the bottom of each table.

TABLE 11–1 Task Map: Preparing to Perform a Secure WAN Boot Installation

Task	Description	For Instructions
Decide what security features you want to use in your installation.	Review the security features and configurations to decide what level of security you want to use in your WAN boot installation.	"Protecting Data During a WAN Boot Installation" on page 161 "Security Configurations Supported by WAN Boot (Overview)" on page 163
Collect WAN boot installation information.	Complete the worksheet to record all the information you need to perform a WAN boot installation.	"Gathering Information for WAN Boot Installations" on page 174
Create the document root directory on the WAN boot server.	Create the document root directory and any subdirectories to serve the configuration and installation files.	"Creating the Document Root Directory" on page 182
Create the WAN boot miniroot.	Use the `setup_install_server` command to create the WAN boot miniroot.	"SPARC: To Create a WAN Boot Miniroot" on page 182
Verify that the client system supports WAN boot.	Check the client OBP for boot argument support of WAN boot.	"To Check the Client OBP for WAN Boot Support" on page 186
Install the `wanboot` program on the WAN boot server.	Copy the `wanboot` program to the document root directory of the WAN boot server.	"Installing the `wanboot` Program on the WAN Boot Server" on page 187
Install the `wanboot-cgi` program on the WAN boot server.	Copy the `wanboot-cgi` program to the WAN boot server's CGI directory.	"To Copy the `wanboot-cgi` Program to the WAN Boot Server" on page 192
(Optional) Set up the logging server.	Configure a dedicated system for displaying boot and installation log messages.	"(Optional) To Configure the WAN Boot Logging Server" on page 193
Set up the `/etc/netboot` hierarchy.	Populate the `/etc/netboot` hierarchy with the configuration and security files that are required for a WAN boot installation.	"Creating the `/etc/netboot` Hierarchy on the WAN Boot Server" on page 189
Configure the web server to use secure HTTP for a more secure WAN boot installation.	Identify the web server requirements that are necessary to perform a WAN installation with HTTPS.	"(Optional) Protecting Data by Using HTTPS" on page 194

TABLE 11–1 Task Map: Preparing to Perform a Secure WAN Boot Installation *(Continued)*

Task	Description	For Instructions
Format digital certificates for a more secure WAN boot installation.	Split PKCS#12 file into a private key and a certificate to use with the WAN installation.	"(Optional) To Use Digital Certificates for Server and Client Authentication" on page 195
Create a hashing key and an encryption key for a more secure WAN boot installation.	Use the `wanbootutil keygen` command to create HMAC SHA1, 3DES, or AES keys.	"(Optional) To Create a Hashing Key and an Encryption Key" on page 197
Create the Solaris Flash archive.	Use the `flar create` command to create an archive of the software that you want to install on the client.	"To Create the Solaris Flash Archive" on page 200
Create the installation files for the custom JumpStart installation.	Use a text editor to create the following files: ■ `sysidcfg` ■ profile ■ `rules.ok` ■ begin scripts ■ finish scripts	"To Create the `sysidcfg` File" on page 202 "To Create the Profile" on page 203 "To Create the `rules` File" on page 205 "(Optional) Creating Begin and Finish Scripts" on page 207
Create the system configuration file.	Set the configuration information in the `system.conf` file.	"To Create the System Configuration File" on page 208
Create the WAN boot configuration file.	Set the configuration information in the `wanboot.conf` file.	"To Create the `wanboot.conf` File" on page 210
(Optional) Configure the DHCP server to support a WAN boot installation.	Set Sun vendor options and macros in the DHCP server.	"Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78

TABLE 11–2 Task Map: Preparing to Perform an Insecure WAN Boot Installation

Task	Description	For Instructions
Decide what security features you want to use in your installation.	Review the security features and configurations to decide what level of security you want to use in your WAN boot installation.	"Protecting Data During a WAN Boot Installation" on page 161 "Security Configurations Supported by WAN Boot (Overview)" on page 163
Collect WAN boot installation information.	Complete the worksheet to record all the information you need to perform a WAN boot installation.	"Gathering Information for WAN Boot Installations" on page 174
Create the document root directory on the WAN boot server.	Create the document root directory and any subdirectories to serve the configuration and installation files.	"Creating the Document Root Directory" on page 182
Create the WAN boot miniroot.	Use the `setup_install_server` command to create the WAN boot miniroot.	"SPARC: To Create a WAN Boot Miniroot" on page 182
Verify that the client system supports WAN boot.	Check the client OBP for boot argument support of WAN boot.	"To Check the Client OBP for WAN Boot Support" on page 186
Install the `wanboot` program on the WAN boot server.	Copy the `wanboot` program to the document root directory of the WAN boot server.	"Installing the `wanboot` Program on the WAN Boot Server" on page 187
Install the `wanboot-cgi` program on the WAN boot server.	Copy the `wanboot-cgi` program to the WAN boot server's CGI directory.	"To Copy the `wanboot-cgi` Program to the WAN Boot Server" on page 192
(Optional) Set up the logging server.	Configure a dedicated system for displaying boot and installation log messages.	"(Optional) To Configure the WAN Boot Logging Server" on page 193
Set up the `/etc/netboot` hierarchy.	Populate the `/etc/netboot` hierarchy with the configuration and security files that are required for a WAN boot installation.	"Creating the `/etc/netboot` Hierarchy on the WAN Boot Server" on page 189

TABLE 11–2 Task Map: Preparing to Perform an Insecure WAN Boot Installation *(Continued)*

Task	Description	For Instructions
(Optional) Create a hashing key.	Use the `wanbootutil keygen` command to create HMAC SHA1 key. For insecure installations that check data integrity, complete this task to create an HMAC SHA1 hashing key.	"(Optional) To Create a Hashing Key and an Encryption Key" on page 197
Create the Solaris Flash archive.	Use the `flar create` command to create an archive of the software that you want to install on the client.	"To Create the Solaris Flash Archive" on page 200
Create the installation files for the custom JumpStart installation.	Use a text editor to create the following files: ■ `sysidcfg` ■ profile ■ `rules.ok` ■ begin scripts ■ finish scripts	"To Create the `sysidcfg` File" on page 202 "To Create the Profile" on page 203 "To Create the `rules` File" on page 205 "(Optional) Creating Begin and Finish Scripts" on page 207
Create the system configuration file.	Set the configuration information in the `system.conf` file.	"To Create the System Configuration File" on page 208
Create the WAN boot configuration file.	Set the configuration information in the `wanboot.conf` file.	"To Create the `wanboot.conf` File" on page 210
(Optional) Configure the DHCP server to support a WAN boot installation.	Set Sun vendor options and macros in the DHCP server.	"Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78

Configuring the WAN Boot Server

The WAN boot server is a web server that provides the boot and configuration data during a WAN boot installation. For a list of the system requirements for the WAN boot server, see Table 10–1.

This section describes the following tasks required to configure the WAN boot server for a WAN boot installation.

- "Creating the Document Root Directory" on page 182
- "Creating the WAN Boot Miniroot" on page 182
- "Installing the `wanboot` Program on the WAN Boot Server" on page 187
- "Creating the `/etc/netboot` Hierarchy on the WAN Boot Server" on page 189
- "Copying the WAN Boot CGI Program to the WAN Boot Server" on page 192
- "(Optional) Protecting Data by Using HTTPS" on page 194

Creating the Document Root Directory

To serve the configuration and installation files, you must make these files accessible to the web server software on the WAN boot server. One method to make these files accessible is to store them in the WAN boot server's document root directory.

If you want to use a document root directory to serve the configuration and installation files, you must create this directory. See your web server documentation for information about how to create the document root directory. For detailed information about how to design your document root directory, see "Storing Installation and Configuration Files in the Document Root Directory" on page 168.

For an example of how to set up this directory, see "Create the Document Root Directory" on page 241.

After you create the document root directory, create the WAN boot miniroot. For instructions, see "Creating the WAN Boot Miniroot" on page 182.

Creating the WAN Boot Miniroot

WAN boot uses a special Solaris miniroot that has been modified to perform a WAN boot installation. The WAN boot miniroot contains a subset of the software in the Solaris miniroot. To perform a WAN boot installation, you must copy the miniroot from the Solaris 10 DVD or the Solaris 10 Software - 1 CD to the WAN boot server. Use the `-w` option to the `setup_install_server` command to copy the WAN boot miniroot from the Solaris software media to your system's hard disk.

▼ SPARC: To Create a WAN Boot Miniroot

This procedure creates a SPARC WAN boot miniroot with SPARC media. If you want to serve a SPARC WAN boot miniroot from an x86–based server, you must create the miniroot on a SPARC machine. After you create the miniroot, copy the miniroot to the document root directory on the x86–based server.

Before You Begin

This procedure assumes that the WAN boot server is running the Volume Manager. If you are not using the Volume Manager, see *System Administration Guide: Devices and File Systems* for information about managing removable media without the Volume Manager.

Steps

1. **Become superuser on the WAN boot server.**

   The system must meet the following requirements.

   - Include a CD-ROM or DVD-ROM drive
   - Be part of the site's network and name service.

     If you use a name service, the system must already be in a name service, such as NIS, NIS+, DNS, or LDAP. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **Insert the Solaris 10 Software - 1 CD or the Solaris 10 DVD in the install server's drive.**

3. **Create a directory for the WAN boot miniroot and Solaris installation image.**

   ```
   # mkdir -p wan-dir-path install-dir-path
   ```

   `-p` Instructs the `mkdir` command to create all the necessary parent directories for the directory you want to create.

   wan-dir-path Specifies the directory where the WAN boot miniroot is to be created on the install server. This directory needs to accommodate miniroots that are typically 250 Mbytes in size.

   install-dir-path Specifies the directory on the install server where the Solaris software image is to be copied. This directory can be removed later in this procedure.

4. **Change to the `Tools` directory on the mounted disc.**

   ```
   # cd /cdrom/cdrom0/s0/Solaris_10/Tools
   ```

   In the previous example, **`cdrom0`** is the path to the drive that contains the Solaris OS media.

5. **Copy the WAN boot miniroot and the Solaris software image to the WAN boot server's hard disk.**

   ```
   # ./setup_install_server -w wan-dir-path install-dir-path
   ```

   wan-dir-path Specifies the directory where the WAN boot miniroot is to be copied

   install-dir-path Specifies the directory where the Solaris software image is to be copied

Note – The `setup_install_server` command indicates whether you have enough disk space available for the Solaris 10 Software disc images. To determine available disk space, use the `df -kl` command.

The `setup_install_server -w` command creates the WAN boot miniroot and a network installation image of the Solaris software.

6. **(Optional) Remove the network installation image.**

   You do not need the Solaris software image to perform a WAN installation with a Solaris Flash archive. You can free up disk space if you do not plan to use the network installation image for other network installations. Type the following command to remove the network installation image.

   ```
   # rm -rf install-dir-path
   ```

7. **Make the WAN boot miniroot available to the WAN boot server in one of the following ways.**

   - **Create a symbolic link to the WAN boot miniroot in the document root directory of the WAN boot server.**

     ```
     # cd /document-root-directory/miniroot
     # ln -s /wan-dir-path/miniroot .
     ```

     document-root-directory/`miniroot` Specifies the directory in the WAN boot server's document root directory where you want to link to the WAN boot miniroot

     /*wan-dir-path*/`miniroot` Specifies the path to the WAN boot miniroot

   - **Move the WAN boot miniroot to the document root directory on the WAN boot server.**

     ```
     # mv /wan-dir-path/miniroot /document-root-directory/miniroot/miniroot-name
     ```

     wan-dir-path/`miniroot` Specifies the path to the WAN boot miniroot.

     /*document-root-directory*/`miniroot/` Specifies the path to the WAN boot miniroot directory in the WAN boot server's document root directory.

     miniroot-name Specifies the name of the WAN boot miniroot. Name the file descriptively, for example `miniroot.s10_sparc`.

Example 11–1 Creating the WAN Boot Miniroot

Use the setup_install_server(1M) with the -w option to copy the WAN boot miniroot and the Solaris software image to the /export/install/Solaris_10 directory of wanserver-1.

Insert the Solaris 10 Software media in the media drive that is attached to wanserver-1. Type the following commands.

```
wanserver-1# mkdir -p /export/install/sol_10_sparc
wanserver-1# cd /cdrom/cdrom0/s0/Solaris_10/Tools
wanserver-1# ./setup_install_server -w /export/install/sol_10_sparc/miniroot \
/export/install/sol_10_sparc
```

Move the WAN boot miniroot to the document root directory (/opt/apache/htdocs/) of the WAN boot server. In this example the name the WAN boot miniroot is set to miniroot.s10_sparc.

```
wanserver-1# mv /export/install/sol_10_sparc/miniroot/miniroot \
/opt/apache/htdocs/miniroot/miniroot.s10_sparc
```

More Information Continuing the WAN Boot Installation

After you create the WAN boot miniroot, verify that the client OpenBoot PROM (OBP) supports WAN boot. For instructions, see “Verifying WAN Boot Support on the Client” on page 185.

See Also For additional information about the setup_install_server command, see install_scripts(1M).

Verifying WAN Boot Support on the Client

To perform an unattended WAN boot installation, the client system’s OpenBoot PROM (OBP) must support WAN boot. If the client’s OBP does not support WAN boot, you can perform a WAN boot installation by providing the necessary programs on a local CD.

You can determine if the client supports WAN boot by checking the client’s OBP configuration variables. Perform the following procedure to check the client for WAN boot support.

▼ To Check the Client OBP for WAN Boot Support

This procedure describes how to determine if the client OBP supports WAN boot.

Steps

1. **Become superuser or assume an equivalent role.**

   Roles contain authorizations and privileged commands. For more information about roles, see "Configuring RBAC (Task Map)" in *System Administration Guide: Security Services*.

2. **Check the OBP configuration variables for WAN boot support.**

   ```
   # eeprom | grep network-boot-arguments
   ```

   - If the variable `network-boot-arguments` is displayed, or if the previous command returns the output `network-boot-arguments: data not available`, the OBP supports WAN boot installations. You do not need to update the OBP before you perform your WAN boot installation.
   - If the previous command does not return any output, the OBP does not support WAN boot installations. You must perform one of the following tasks.
     - Update the client OBP. See your system documentation for information about how to update the OBP.
     - After you complete the preparation tasks and are ready to install the client, perform the WAN boot installation from the Solaris 10 Software CD in a local CD-ROM drive.

       For instructions about how to boot the client from a local CD-ROM drive, see "To Perform a WAN Boot Installation With Local CD Media" on page 234. To continue preparing for the WAN boot installation, see "Creating the `/etc/netboot` Hierarchy on the WAN Boot Server" on page 189.

Example 11–2 Verifying OBP Support for WAN Boot on the Client

The following command shows how to check the client OBP for WAN boot support.

```
# eeprom | grep network-boot-arguments
network-boot-arguments: data not available
```

In this example, the output `network-boot-arguments: data not available` indicates that the client OBP supports WAN boot.

More Information Continuing the WAN Boot Installation

After you verify that the client OBP supports WAN boot, you must copy the `wanboot` program to the WAN boot server. For instructions, see "Installing the `wanboot` Program on the WAN Boot Server" on page 187.

If the client OBP does not support WAN boot, you do not need to copy the wanboot program to the WAN boot server. You must provide the wanboot program to the client on a local CD. To continue the installation, see "Creating the /etc/netboot Hierarchy on the WAN Boot Server" on page 189

See Also For additional information about the setup_install_server command, see Chapter 7.

Installing the wanboot Program on the WAN Boot Server

WAN boot uses a special second-level boot program (wanboot) to install the client. The wanboot program loads the WAN boot miniroot, client configuration files, and installation files that are required to perform a WAN boot installation.

To perform a WAN boot installation, you must provide the wanboot program to the client during the installation. You can provide this program to the client in the following ways.

- If your client's PROM supports WAN boot, you can transmit the program from the WAN boot server to the client. You must install the wanboot program on the WAN boot server.

  To check if your client's PROM supports WAN boot, see "To Check the Client OBP for WAN Boot Support" on page 186.
- If your client's PROM does not support WAN boot, you must provide the program to the client on a local CD. If your client's PROM does not support WAN boot, go to "Creating the /etc/netboot Hierarchy on the WAN Boot Server" on page 189 to continue preparing for your installation.

▼ SPARC: To Install the wanboot Program on the WAN Boot Server

This procedure describes how to copy the wanboot program from Solaris media to the WAN boot server.

This procedure assumes that the WAN boot server is running the Volume Manager. If you are not using the Volume Manager, see *System Administration Guide: Devices and File Systems* for information about managing removable media without the Volume Manager.

Before You Begin Verify that your client system supports WAN boot. See "To Check the Client OBP for WAN Boot Support" on page 186 for more information.

Steps

1. **Become superuser on the install server.**

2. **Insert the Solaris 10 Software - 1 CD or the Solaris 10 DVD in the install server's drive.**

3. **Change to the `sun4u` platform directory on the Solaris 10 Software - 1 CD or the Solaris 10 DVD.**

   ```
   # cd /cdrom/cdrom0/s0/Solaris_10/Tools/Boot/platform/sun4u/
   ```

4. **Copy the `wanboot` program to the install server.**

   ```
   # cp wanboot /document-root-directory/wanboot/wanboot-name
   ```

   document-root-directory Specifies the document root directory of the WAN boot server.

   wanboot-name Specifies the name of the `wanboot` program. Name this file descriptively, for example, `wanboot.s9_sparc`.

5. **Make the `wanboot` program available to the WAN boot server in one of the following ways.**

   - Create a symbolic link to the `wanboot` program in the document root directory of the WAN boot server.

     ```
     # cd /document-root-directory/wanboot
     # ln -s /wan-dir-path/wanboot .
     ```

     document-root-directory/`wanboot` Specifies the directory in the WAN boot server's document root directory where you want to link to the `wanboot` program

     /*wan-dir-path*/`wanboot` Specifies the path to the `wanboot` program

   - Move the WAN boot miniroot to the document root directory on the WAN boot server.

     ```
     # mv /wan-dir-path/wanboot /document-root-directory/wanboot/wanboot-name
     ```

     wan-dir-path/`wanboot` Specifies the path to the `wanboot` program

     /*document-root-directory*/`wanboot/` Specifies the path to the `wanboot` program directory in the WAN boot server's document root directory.

     wanboot-name Specifies the name of the `wanboot` program. Name the file descriptively, for example `wanboot.s10_sparc`.

Example 11–3 Installing the wanboot Program on the WAN Boot Server

To install the wanboot program on the WAN boot server, copy the program from the Solaris 10 Software media to the WAN boot server's document root directory.

Insert the Solaris 10 DVD or the Solaris 10 Software - 1 CD in the media drive that is attached to wanserver-1 and type the following commands.

```
wanserver-1# cd /cdrom/cdrom0/s0/Solaris_10/Tools/Boot/platform/sun4u/
wanserver-1# cp wanboot /opt/apache/htdocs/wanboot/wanboot.s10_sparc
```

In this example, the name of the wanboot program is set to wanboot.s10_sparc.

More Information Continuing the WAN Boot Installation

After you install the wanboot program on the WAN boot server, you must create the /etc/netboot hierarchy on the WAN boot server. For instructions, see "Creating the /etc/netboot Hierarchy on the WAN Boot Server" on page 189.

See Also For overview information about the wanboot program, see "What Is WAN Boot?" on page 157.

Creating the /etc/netboot Hierarchy on the WAN Boot Server

During the installation, WAN boot refers to the contents of the /etc/netboot hierarchy on the web server for instructions about how to perform the installation. This directory contains the configuration information, private key, digital certificate, and certificate authority required for a WAN boot installation. During the installation, the wanboot-cgi program converts this information into the WAN boot file system. The wanboot-cgi program then transmits the WAN boot file system to the client.

You can create subdirectories within the /etc/netboot directory to customize the scope of the WAN installation. Use the following directory structures to define how configuration information is shared among the clients that you want to install.

- **Global configuration** – If you want all the clients on your network to share configuration information, store the files that you want to share in the /etc/netboot directory.
- **Network-specific configuration** – If you want only those machines on a specific subnet to share configuration information, store the configuration files that you want to share in a subdirectory of /etc/netboot. Have the subdirectory follow this naming convention.

  /etc/netboot/*net-ip*

In this example, *net-ip* is the IP address of the client's subnet.

- **Client-specific configuration** – If you want only a specific client to use the boot file system, store the boot file system files in a subdirectory of /etc/netboot. Have the subdirectory follow this naming convention.

  /etc/netboot/*net-ip*/*client-ID*

  In this example, *net-ip* is the IP address of the subnet. *client-ID* is either the client ID that is assigned by the DHCP server, or a user-specified client ID.

For detailed planning information about these configurations, see "Storing Configuration and Security Information in the /etc/netboot Hierarchy" on page 170.

The following procedure describes how to create the /etc/netboot hierarchy.

▼ To Create the /etc/netboot Hierarchy on the WAN Boot Server

Follow these steps to create the /etc/netboot hierarchy.

Steps

1. **Become superuser on the WAN boot server.**

2. **Create the /etc/netboot directory.**

   ```
   # mkdir /etc/netboot
   ```

3. **Change the permissions of the /etc/netboot directory to 700.**

   ```
   # chmod 700 /etc/netboot
   ```

4. **Change the owner of the /etc/netboot directory to the web server owner.**

   ```
   # chown web-server-user:web-server-group /etc/netboot/
   ```

   web-server-user Specifies the user owner of the web server process

   web-server-group Specifies the group owner of the web server process

5. **Exit the superuser role.**

   ```
   # exit
   ```

6. **Assume the user role of the web server owner.**

7. **Create the client subdirectory of the /etc/netboot directory.**

   ```
   # mkdir -p /etc/netboot/net-ip/client-ID
   ```

   -p Instructs the mkdir command to create all the necessary parent directories for the directory you want to create

(Optional) *net-ip*	Specifies the network IP address of the client's subnet.
(Optional) *client-ID*	Specifies the client ID. The client ID can be a user-defined value or the DHCP client ID. The *client-ID* directory must be a subdirectory of the *net-ip* directory.

8. **For each directory in the `/etc/netboot` hierarchy, change the permissions to 700.**

   ```
   # chmod 700 /etc/netboot/dir-name
   ```

dir-name	Specifies the name of a directory in the `/etc/netboot` hierarchy

Example 11–4 Creating the `/etc/netboot` Hierarchy on the WAN Boot Server

The following example shows how to create the `/etc/netboot` hierarchy for the client 010003BA152A42 on subnet 192.168.198.0. In this example, the user `nobody` and the group `admin` own the web server process.

The commands in this example perform the following tasks.

- Create the `/etc/netboot` directory.
- Change the permissions of the `/etc/netboot` directory to 700.
- Change the ownership of the `/etc/netboot` directory to the owner of the web server process.
- Assume the same user role as the web server user.
- Create a subdirectory of `/etc/netboot` that is named after the subnet (192.168.198.0).
- Create a subdirectory of the subnet directory that is named after the client ID.
- Change the permissions of the `/etc/netboot` subdirectories to 700.

```
# cd /
# mkdir /etc/netboot/
# chmod 700 /etc/netboot
# chown nobody:admin /etc/netboot
# exit
server# su nobody
Password:
nobody# mkdir -p /etc/netboot/192.168.198.0/010003BA152A42
nobody# chmod 700 /etc/netboot/192.168.198.0
nobody# chmod 700 /etc/netboot/192.168.198.0/010003BA152A42
```

More Information Continuing the WAN Boot Installation

After you create the `/etc/netboot` hierarchy, you must copy the WAN Boot CGI program to the WAN boot server. For instructions, see "Copying the WAN Boot CGI Program to the WAN Boot Server" on page 192.

See Also For detailed planning information about how to design the /etc/netboot hierarchy, see "Storing Configuration and Security Information in the /etc/netboot Hierarchy" on page 170.

Copying the WAN Boot CGI Program to the WAN Boot Server

The wanboot-cgi program creates the data streams that transmit the following files from the WAN boot server to the client.

- wanboot program
- WAN boot file system
- WAN boot miniroot

The wanboot-cgi program is installed on the system when you install the Solaris 10 software. To enable the WAN boot server to use this program, copy this program to the cgi-bin directory of the WAN boot server.

▼ To Copy the wanboot-cgi Program to the WAN Boot Server

Steps

1. **Become superuser on the WAN boot server.**
2. **Copy the wanboot-cgi program to the WAN boot server.**

```
# cp /usr/lib/inet/wanboot/wanboot-cgi /WAN-server-root/cgi-bin/wanboot-cgi
```

/WAN-server-root Specifies the root directory of the web server software on the WAN boot server

3. **On the WAN boot server, change the permissions of the CGI program to 755.**

```
# chmod 755 /WAN-server-root/cgi-bin/wanboot-cgi
```

More Information

Continuing the WAN Boot Installation

After you copy the WAN boot CGI program to the WAN boot server, you can optionally set up a logging server. For instructions, see "(Optional) To Configure the WAN Boot Logging Server" on page 193.

If you do not want to set up a separate logging server, see "(Optional) Protecting Data by Using HTTPS" on page 194 for instructions about how to set up the security features of a WAN boot installation.

See Also For overview information about the `wanboot-cgi` program, see "What Is WAN Boot?" on page 157.

▼ (Optional) To Configure the WAN Boot Logging Server

By default, all WAN boot logging messages are displayed on the client system. This default behavior enables you to quickly debug any installation issues.

If you want to record boot and installation logging messages on a system other than the client, you must set up a logging server. If you want to use a logging server with HTTPS during the installation, you must configure the WAN boot server as the logging server.

To configure the logging server, follow these steps.

Steps

1. **Copy the `bootlog-cgi` script to the logging server's CGI script directory.**

   ```
   # cp /usr/lib/inet/wanboot/bootlog-cgi \    log-server-root/cgi-bin
   ```

   log-server-root/`cgi-bin` Specifies the `cgi-bin` directory in the logging server's web server directory

2. **Change the permissions of the `bootlog-cgi` script to 755.**

   ```
   # chmod 755 log-server-root/cgi-bin/bootlog-cgi
   ```

3. **Set the value of the `boot_logger` parameter in the `wanboot.conf` file.**

   In the `wanboot.conf` file, specify the URL of the `bootlog-cgi` script on the logging server.

   For more information about setting parameters in the `wanboot.conf` file, see "To Create the `wanboot.conf` File" on page 210.

   During the installation, boot and installation log messages are recorded in the `/tmp` directory of the logging server. The log file is named `bootlog.`*hostname*, where *hostname* is the host name of the client.

Example 11–5 Configuring a Logging Server for WAN Boot Installation Over HTTPS

The following example configures the WAN boot server as a logging server.

```
# cp /usr/lib/inet/wanboot/bootlog-cgi /opt/apache/cgi-bin/
# chmod 755 /opt/apache/cgi-bin/bootlog-cgi
```

More Information

Continuing the WAN Boot Installation

After you set up the logging server, you can optionally set up the WAN boot installation to use digital certificates and security keys. See "(Optional) Protecting Data by Using HTTPS" on page 194 for instructions about how to set up the security features of a WAN boot installation.

(Optional) Protecting Data by Using HTTPS

To protect your data during the transfer from the WAN boot server to the client, you can use HTTP over Secure Sockets Layer (HTTPS). To use the more secure installation configuration that is described in "Secure WAN Boot Installation Configuration" on page 163, you must enable your web server to use HTTPS.

If you do not want to perform a secure WAN boot, skip the procedures in this section. To continue preparing for your less secure installation, see "Creating the Custom JumpStart Installation Files" on page 200.

To enable the web server software on the WAN boot server to use HTTPS, you must perform the following tasks.

- Activate Secure Sockets Layer (SSL) support in your web server software.

  The processes for enabling SSL support and client authentication vary by web server. This document does not describe how to enable these security features on your web server. For information about these features, see the following documentation.

  - For information about activating SSL on the SunONE and iPlanet web servers, see the SunONE and iPlanet documentation collections on http://docs.sun.com.
  - For information about activating SSL on the Apache web server, see the Apache Documentation Project at http://httpd.apache.org/docs-project/.
  - If you are using web server software that is not listed in the previous list, see your web server software documentation.
- Install digital certificates on the WAN boot server.

  For information about using digital certificates with WAN boot, see "(Optional) To Use Digital Certificates for Server and Client Authentication" on page 195.
- Provide a trusted certificate to the client.

  For instructions about how to create a trusted certificate, see "(Optional) To Use Digital Certificates for Server and Client Authentication" on page 195.

- Create a hashing key and an encryption key.

  For instructions about how to create keys, see "(Optional) To Create a Hashing Key and an Encryption Key" on page 197.
- (Optional) Configure the web server software to support client authentication.

  For information about how to configure your web server to support client authentication, see your web server documentation.

This section describes how to use digital certificates and keys in your WAN boot installation.

▼ (Optional) To Use Digital Certificates for Server and Client Authentication

The WAN boot installation method can use PKCS#12 files to perform an installation over HTTPS with server or both client and server authentication. For requirements and guidelines about using PKCS#12 files, see "Digital Certificate Requirements" on page 173.

To use a PKCS#12 file in a WAN boot installation, you perform the following tasks.

- Split the PKCS#12 file into separate SSL private key and trusted certificate files.
- Insert the trusted certificate in the client's `truststore` file in the `/etc/netboot` hierarchy. The trusted certificate instructs the client to trust the server.
- (Optional) Insert the contents of the SSL private key file in the client's `keystore` file in the `/etc/netboot` hierarchy.

The `wanbootutil` command provides options to perform the tasks in the previous list.

If you do not want to perform a secure WAN boot, skip this procedure. To continue preparing for your less secure installation, see "Creating the Custom JumpStart Installation Files" on page 200.

Follow these steps to create a trusted certificate and a client private key.

Before You Begin Before you split a PKCS#12 file, create the appropriate subdirectories of the `/etc/netboot` hierarchy on the WAN boot server.

- For overview information that describes the `/etc/netboot` hierarchy, see "Storing Configuration and Security Information in the `/etc/netboot` Hierarchy" on page 170.
- For instructions about how to create the `/etc/netboot` hierarchy, see "Creating the `/etc/netboot` Hierarchy on the WAN Boot Server" on page 189.

Steps

1. **Assume the same user role as the web server user on the WAN boot server.**

2. **Extract the trusted certificate from the PKCS#12 file. Insert the certificate in the client's `truststore` file in the `/etc/netboot` hierarchy.**

   ```
   # wanbootutil p12split -i p12cert \
   -t /etc/netboot/net-ip/client-ID/truststore
   ```

   `p12split`
   Option to `wanbootutil` command that splits a PKCS#12 file into separate private key and certificate files.

   `-i` *p12cert*
   Specifies the name of the PKCS#12 file to split.

   `-t /etc/netboot/`*net-ip*`/`*client-ID*`/truststore`
   Inserts the certificate in the client's `truststore` file. *net-ip* is the IP address of the client's subnet. *client-ID* can be a user-defined ID or the DHCP client ID.

3. **(Optional) Decide if you want to require client authentication.**

   - **If no, go to "(Optional) To Create a Hashing Key and an Encryption Key" on page 197.**

   - **If yes, continue with the following steps.**

     a. **Insert the client certificate in the client's `certstore`.**

     ```
     # wanbootutil p12split -i p12cert -c \
     /etc/netboot/net-ip/client-ID/certstore -k keyfile
     ```

     `p12split`
     Option to `wanbootutil` command that splits a PKCS#12 file into separate private key and certificate files.

     `-i` *p12cert*
     Specifies the name of the PKCS#12 file to split.

     `-c /etc/netboot/`*net-ip*`/`*client-ID*`/certstore`
     Inserts the client's certificate in the client's `certstore`. *net-ip* is the IP address of the client's subnet. *client-ID* can be a user-defined ID or the DHCP client ID.

     `-k` *keyfile*
     Specifies the name of the client's SSL private key file to create from the split PKCS#12 file.

     b. **Insert the private key in the client's `keystore`.**

     ```
     # wanbootutil keymgmt -i -k keyfile \
     -s /etc/netboot/net-ip/client-ID/keystore -o type=rsa
     ```

     `keymgmt -i`
     Inserts an SSL private key in the client's `keystore`

`-k` *keyfile*
Specifies the name of the client's private key file that was created in the previous step

`-s /etc/netboot/`*net-ip*`/`*client-ID*`/keystore`
Specifies the path to the client's `keystore`

`-o type=rsa`
Specifies the key type as RSA

Example 11–6 Creating a Trusted Certificate for Server Authentication

In the following example, you use a PKCS#12 file to install client 010003BA152A42 on subnet 192.168.198.0. This command sample extracts a certificate from a PKCS#12 file that is named `client.p12`. The command then places the contents of the trusted certificate in the client's `truststore` file.

Before you execute these commands, you must first assume the same user role as the web server user. In this example, the web server user role is `nobody`.

```
server# su nobody
Password:
nobody# wanbootutil p12split -i client.p12 \
-t /etc/netboot/192.168.198.0/010003BA152A42/truststore
nobody# chmod 600 /etc/netboot/192.168.198.0/010003BA152A42/truststore
```

More Information Continuing the WAN Boot Installation

After you create a digital certificate, create a hashing key and an encryption key. For instructions, see"(Optional) To Create a Hashing Key and an Encryption Key" on page 197.

See Also For more information about how to create trusted certificates, see the man page `wanbootutil`(1M).

▼ (Optional) To Create a Hashing Key and an Encryption Key

If you want to use HTTPS to transmit your data, you must create a HMAC SHA1 hashing key and an encryption key. If you plan to install over a semi-private network, you might not want to encrypt the installation data. You can use a HMAC SHA1 hashing key to check the integrity of the `wanboot` program.

By using the `wanbootutil keygen` command, you can generate these keys and store them in the appropriate `/etc/netboot` directory.

If you do not want to perform a secure WAN boot, skip this procedure. To continue preparing for your less secure installation, see "Creating the Custom JumpStart Installation Files" on page 200.

To create a hashing key and an encryption key, follow these steps.

Steps

1. **Assume the same user role as the web server user on the WAN boot server.**
2. **Create the master HMAC SHA1 key.**

   ```
   # wanbootutil keygen -m
   ```

   `keygen -m` Creates the master HMAC SHA1 key for the WAN boot server
3. **Create the HMAC SHA1 hashing key for the client from the master key.**

   ```
   # wanbootutil keygen -c -o [net=net-ip,{cid=client-ID,}]type=sha1
   ```

   `-c` Creates the client's hashing key from the master key.

   `-o` Indicates that additional options are included for the `wanbootutil keygen` command.

   (Optional) `net=`*net-ip* Specifies the IP address for the client's subnet. If you do not use the `net` option, the key is stored in the `/etc/netboot/keystore` file, and can be used by all WAN boot clients.

   (Optional) `cid=`*client-ID* Specifies the client ID. The client ID can be a user-defined ID or the DHCP client ID. The `cid` option must be preceded by a valid `net=` value. If you do not specify the `cid` option with the `net` option, the key is stored in the `/etc/netboot/`*net-ip*`/keystore` file. This key can be used by all WAN boot clients on the *net-ip* subnet.

   `type=sha1` Instructs the `wanbootutil keygen` utility to create a HMAC SHA1 hashing key for the client.
4. **Decide if you need to create an encryption key for the client.**

   You need to create an encryption key to perform a WAN boot installation over HTTPS. Before the client establishes an HTTPS connection with the WAN boot server, the WAN boot server transmits encrypted data and information to the client. The encryption key enables the client to decrypt this information and use this information during the installation.

   - If you are performing a more secure WAN installation over HTTPS with server authentication, continue.
   - If you only want to check the integrity of the `wanboot` program, you do not need to create an encryption key. Go to Step 6.
5. **Create an encryption key for the client.**

   ```
   # wanbootutil keygen -c -o [net=net-ip,{cid=client-ID,}]type=key-type
   ```

   `-c` Creates the client's encryption key.

`-o`
Indicates that additional options are included for the `wanbootutil keygen` command.

(Optional) `net=`*net-ip*
Specifies the network IP address for the client. If you do not use the `net` option, the key is stored in the `/etc/netboot/keystore` file, and can be used by all WAN boot clients.

(Optional) `cid=`*client-ID*
Specifies the client ID. The client ID can be a user-defined ID, or the DHCP client ID. The `cid` option must be preceded by a valid `net=` value. If you do not specify the `cid` option with the `net` option, the key is stored in the `/etc/netboot/`*net-ip*`/keystore` file. This key can be used by all WAN boot clients on the *net-ip* subnet.

`type=`*key-type*
Instructs the `wanbootutil keygen` utility to create an encryption key for the client. *key-type* can have a value of `3des` or `aes`.

6. **Install the keys on the client system.**

   For instructions about how to install keys on the client, see "Installing Keys on the Client" on page 220.

Example 11–7 Creating Required Keys for WAN Boot Installation Over HTTPS

The following example creates a master HMAC SHA1 key for the WAN boot server. This example also creates a HMAC SHA1 hashing key and 3DES encryption key for client 010003BA152A42 on subnet 192.168.198.0.

Before you execute these commands, you must first assume the same user role as the web server user. In this example, the web server user role is `nobody`.

```
server# su nobody
Password:
nobody# wanbootutil keygen -m
nobody# wanbootutil keygen -c -o net=192.168.198.0,cid=010003BA152A42,type=sha1
nobody# wanbootutil keygen -c -o net=192.168.198.0,cid=010003BA152A42,type=3des
```

More Information Continuing the WAN Boot Installation

After you create a hashing and an encryption key, you must create the installation files. For instructions, see "Creating the Custom JumpStart Installation Files" on page 200.

See Also For overview information on hashing keys and encryption keys, see "Protecting Data During a WAN Boot Installation" on page 161.

For more information about how to create hashing and encryption keys, see the man page `wanbootutil`(1M).

Creating the Custom JumpStart Installation Files

WAN boot performs a custom JumpStart installation to install a Solaris Flash archive on the client. The custom JumpStart installation method is a command–line interface that enables you to automatically install several systems, based on profiles that you create. The profiles define specific software installation requirements. You can also incorporate shell scripts to include preinstallation and postinstallation tasks. You choose which profile and scripts to use for installation or upgrade. The custom JumpStart installation method installs or upgrades the system, based on the profile and scripts that you select. Also, you can use a `sysidcfg` file to specify configuration information so that the custom JumpStart installation is completely free of manual intervention.

To prepare the custom JumpStart files for a WAN boot installation, complete the following tasks.

- "To Create the Solaris Flash Archive" on page 200
- "To Create the `sysidcfg` File" on page 202
- "To Create the `rules` File" on page 205
- "To Create the Profile" on page 203
- "(Optional) Creating Begin and Finish Scripts" on page 207

For detailed information on the custom JumpStart installation method, see Chapter 3, "Custom JumpStart (Overview)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

▼ To Create the Solaris Flash Archive

The Solaris Flash installation feature enables you to use a single reference installation of the Solaris OS on a system, which is called the master system. You can then create a Solaris Flash archive, which is a replica image of the master system. You can install the Solaris Flash archive on other systems in the network, creating clone systems.

This section describes how to create a Solaris Flash archive.

Before You Begin Before you create a Solaris Flash archive, you must first install the master system.

- For information about installing a master system, see "Installing the Master System" in *Solaris 10 Installation Guide: Solaris Flash Archives (Creation and Installation)*.
- For detailed information about Solaris Flash archives, see Chapter 1, "Solaris Flash (Overview)," in *Solaris 10 Installation Guide: Solaris Flash Archives (Creation and Installation)*.

Check your web server software documentation to verify that the software can transmit files that are the size of a Solaris Flash archive.

Steps

1. **Boot the master system.**

   Run the master system in as inactive a state as possible. When possible, run the system in single-user mode. If that is not possible, shut down any applications that you want to archive and any applications that require extensive operating system resources.

2. **To create the archive, use the `flar create` command.**

   ```
   # flar create -n name [optional-parameters]   document-root/flash/filename
   ```

   name — The name that you give the archive. The *name* you specify is the value of the `content_name` keyword.

   optional-parameters — You can use several options to the `flar create` command to customize your Solaris Flash archive. For detailed descriptions of these options, see Chapter 5, "Solaris Flash (Reference)," in *Solaris 10 Installation Guide: Solaris Flash Archives (Creation and Installation)*.

   document-root`/flash` — The path to the Solaris Flash subdirectory of the install server's document root directory.

   filename — The name of the archive file.

   To conserve disk space, you might want to use the `-c` option to the `flar create` command to compress the archive. However, a compressed archive can affect the performance of your WAN boot installation. For more information about creating a compressed archive, see the man page `flarcreate`(1M).

   - If the archive creation is successful, the `flar create` command returns an exit code of 0.
   - If the archive creation fails, the `flar create` command returns a nonzero exit code.

Example 11–8 Creating a Solaris Flash Archive for a WAN Boot Installation

In this example, you create your Solaris Flash archive by cloning the WAN boot server system with the host name wanserver. The archive is named sol_10_sparc, and is copied exactly from the master system. The archive is an exact duplicate of the master system. The archive is stored in sol_10_sparc.flar. You save the archive in the flash/archives subdirectory of the document root directory on the WAN boot server.

```
wanserver# flar create -n sol_10_sparc \
/opt/apache/htdocs/flash/archives/sol_10_sparc.flar
```

More Information Continuing the WAN Boot Installation

After you create the Solaris Flash archive, preconfigure the client information in the sysidcfg file. For instructions, see "To Create the sysidcfg File" on page 202.

See Also For detailed instructions about how to create a Solaris Flash archive, see Chapter 3, "Creating Solaris Flash Archives (Tasks)," in *Solaris 10 Installation Guide: Solaris Flash Archives (Creation and Installation)*.

For more information about the flar create command, see the man page flarcreate(1M).

▼ To Create the sysidcfg File

You can specify a set of keywords in the sysidcfg file to preconfigure a system.

To create the sysidcfg file, follow these steps.

Before You Begin Create the Solaris Flash archive. See "To Create the Solaris Flash Archive" on page 200 for detailed instructions.

Steps

1. **Create a file called sysidcfg in a text editor on the install server.**
2. **Type the sysidcfg keywords you want.**

   For detailed information about sysidcfg keywords, see "sysidcfg File Keywords" on page 59.
3. **Save the sysidcfg file in a location that is accessible to the WAN boot server.**

   Save the file to one of the following locations.

   - If the WAN boot server and install server are hosted on the same machine, save this file to the flash subdirectory of the document root directory on the WAN boot server.

- If the WAN boot server and install server are not on the same machine, save this file to the flash subdirectory of the document root directory of the install server.

Example 11–9 sysidcfg File for WAN Boot Installation

The following is an example of a sysidcfg file for a SPARC based system. The host name, IP address, and netmask of this system have been preconfigured by editing the name service.

```
network_interface=primary {hostname=wanclient
                           default_route=192.168.198.1
                           ip_address=192.168.198.210
                           netmask=255.255.255.0
                           protocol_ipv6=no}
timezone=US/Central
system_locale=C
terminal=xterm
timeserver=localhost
name_service=NIS {name_server=matter(192.168.255.255)
                  domain_name=mind.over.example.com
                  }
security_policy=none
```

More Information Continuing the WAN Boot Installation

After you create the sysidcfg file, create a custom JumpStart profile for the client. For instructions, see "To Create the Profile" on page 203.

See Also For more detailed information about sysidcfg keywords and values, see "Preconfiguring With the sysidcfg File" on page 57.

▼ To Create the Profile

A profile is a text file that instructs the custom JumpStart program how to install the Solaris software on a system. A profile defines elements of the installation, for example, the software group to install.

For detailed information about how to create profiles, see "Creating a Profile" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

To create the profile, follow these steps.

Before You Begin Create the sysidcfg file for the client. See "To Create the sysidcfg File" on page 202 for detailed instructions.

Steps

1. **Create a text file on the install server. Name the file descriptively.**

   Ensure that the name of the profile reflects how you intend to use the profile to install the Solaris software on a system. For example, you might name the profiles `basic_install`, `eng_profile`, or `user_profile`.

2. **Add profile keywords and values to the profile.**

   For a list of profile keywords and values, see "Profile Keywords and Values" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

   Profile keywords and their values are case sensitive.

3. **Save the profile in a location that is accessible to the WAN boot server.**

   Save the profile in one of the following locations.

   - If the WAN boot server and install server are hosted on the same machine, save this file to the `flash` subdirectory of the document root directory on the WAN boot server.
   - If the WAN boot server and install server are not on the same machine, save this file to the `flash` subdirectory of the document root directory of the install server.

4. **Ensure that `root` owns the profile and that the permissions are set to 644.**

5. **(Optional) Test the profile.**

   "Testing a Profile" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations* contains information about testing profiles.

Example 11–10 Retrieving a Solaris Flash Archive From a Secure HTTP Server

In the following example, the profile indicates that the custom JumpStart program retrieves the Solaris Flash archive from a secure HTTP server.

```
# profile keywords         profile values
# ----------------         -------------------
install_type               flash_install
archive_location           https://192.168.198.2/sol_10_sparc.flar
partitioning               explicit
filesys                    c0t1d0s0 4000 /
filesys                    c0t1d0s1 512 swap
filesys                    c0t1d0s7 free /export/home
```

The following list describes some of the keywords and values from this example.

`install_type` The profile installs a Solaris Flash archive on the clone system. All files are overwritten as in an initial installation.

`archive_location` The compressed Solaris Flash archive is retrieved from a secure HTTP server.

partitioning	The file system slices are determined by the filesys keywords, value explicit. The size of root (/) is based on the size of the Solaris Flash archive. The size of swap is set to the necessary size and is installed on c0t1d0s1. /export/home is based on the remaining disk space. /export/home is installed on c0t1d0s7.

More Information

Continuing the WAN Boot Installation

After you create a profile, you must create and validate the rules file. For instructions, see "To Create the rules File" on page 205.

See Also

For more information about how to create a profile, see "Creating a Profile" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

For more detailed information about profile keywords and values, see "Profile Keywords and Values" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

▼ To Create the rules File

The rules file is a text file that contains a rule for each group of systems on which you want to install the Solaris OS. Each rule distinguishes a group of systems that are based on one or more system attributes. Each rule also links each group to a profile. A profile is a text file that defines how the Solaris software is to be installed on each system in the group. For example, the following rule specifies that the JumpStart program use the information in the basic_prof profile to install any system with the sun4u platform group.

```
karch sun4u - basic_prof -
```

The rules file is used to create the rules.ok file, which is required for custom JumpStart installations.

For detailed information about how to create a rules file, see "Creating the rules File" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

To create the rules file, follow these steps.

Before You Begin

Create the profile for the client. See "To Create the Profile" on page 203 for detailed instructions.

Steps

1. **On the install server, create a text file that is named rules.**
2. **Add a rule in the rules file for each group of systems you want to install.**

For detailed information about how to create a rules file, see "Creating the rules File" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

3. **Save the `rules` file on the install server.**

4. **Validate the `rules` file.**

   ```
   $ ./check -p path -r file-name
   ```

   `-p` *path* Validates the `rules` by using the `check` script from the Solaris 10 software image instead of the `check` script from the system you are using. *path* is the image on a local disk or a mounted Solaris 10 DVD or a Solaris 10 Software - 1 CD.

   Use this option to run the most recent version of `check` if your system is running a previous version of the Solaris OS.

   `-r` *file_name* Specifies a rules file other than the file that is named `rules`. By using this option, you can test the validity of a rule before you integrate the rule into the `rules` file.

   As the `check` script runs, the script reports the checking of the validity of the `rules` file and each profile. If no errors are encountered, the script reports: `The custom JumpStart configuration is ok`. The `check` script creates the `rules.ok` file.

5. **Save the `rules.ok` file in a location that is accessible to the WAN boot server.**

   Save the file to one of the following locations.

   - If the WAN boot server and install server are hosted on the same machine, save this file to the `flash` subdirectory of the document root directory on the WAN boot server.
   - If the WAN boot server and install server are not on the same machine, save this file to the `flash` subdirectory of the document root directory of the install server.

6. **Ensure that `root` owns the `rules.ok` file and that the permissions are set to 644.**

Example 11–11 Creating and Validating the `rules` File

The custom JumpStart programs use the `rules` file to select the correct installation profile for the `wanclient-1` system. Create a text file that is named `rules`. Then, add keywords and values to this file.

The IP address of the client system is 192.168.198.210, and the netmask is 255.255.255.0. Use the `network` rule keyword to specify the profile that the custom JumpStart programs should use to install the client.

```
network 192.168.198.0 - wanclient_prof -
```

This rules file instructs the custom JumpStart programs to use the wanclient_prof to install the Solaris 10 software on the client.

Name this rule file wanclient_rule.

After you create the profile and the rules file, you run the check script to verify that the files are valid.

```
wanserver# ./check -r wanclient_rule
```

If the check script does not find any errors, the script creates the rules.ok file.

Save the rules.ok file in the /opt/apache/htdocs/flash/ directory.

More Information

Continuing the WAN Boot Installation

After you create the rules.ok file, you can optionally set up begin and finish scripts for your installation. For instructions, see "(Optional) Creating Begin and Finish Scripts" on page 207.

If you do not want to set up begin and finish scripts, see "Creating the Configuration Files" on page 208 to continue the WAN boot installation.

See Also

For more information about how to create a rules file, see "Creating the rules File" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

For more detailed information about rules file keywords and values, see "Rule Keywords and Values" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

(Optional) Creating Begin and Finish Scripts

Begin and finish scripts are user-defined Bourne shell scripts that you specify in the rules file. A begin script performs tasks before the Solaris software is installed on a system. A finish script performs tasks after the Solaris software is installed on a system, but before the system reboots. You can use these scripts only when using custom JumpStart to install Solaris.

You can use begin scripts to create derived profiles. Finish scripts enable you to perform various postinstallation tasks, such as adding files, packages, patches, or additional software.

You must store the begin and finish scripts in the same directory as the sysidcfg, rules.ok, and profile files on the install server.

- For more information about creating begin scripts, see "Creating Begin Scripts" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

- For more information about creating finish scripts, see "Creating Finish Scripts" in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

To continue preparing for your WAN boot installation, see "Creating the Configuration Files" on page 208.

Creating the Configuration Files

WAN boot uses the following files to specify the location of the data and files that are required for a WAN boot installation.

- System configuration file (`system.conf`)
- `wanboot.conf` file

This section describes how to create and store these two files.

▼ To Create the System Configuration File

In the system configuration file, you can direct the WAN boot installation programs to the following files.

- `sysidcfg` file
- `rules.ok` file
- Custom JumpStart profile

WAN boot follows the pointers in the system configuration file to install and configure the client.

The system configuration file is a plain text file, and must be formatted in the following pattern.

setting=value

To use a system configuration file to direct the WAN installation programs to the `sysidcfg`, `rules.ok`, and profile files, follow these steps.

Before You Begin Before you create the system configuration file, you must create the installation files for you WAN boot installation. See "Creating the Custom JumpStart Installation Files" on page 200 for detailed instructions.

Steps 1. **Assume the same user role as the web server user on the WAN boot server.**

2. **Create a text file. Name the file descriptively, for example, `sys-conf.s10-sparc`.**

3. **Add the following entries to the system configuration file.**

   `SsysidCF=`*sysidcfg-file-URL*
   This setting points to the `flash` directory on the install server that contains the `sysidcfg` file. Make sure that this URL matches the path to the `sysidcfg` file that you created in "To Create the `sysidcfg` File" on page 202.

   For WAN installations that use HTTPS, set the value to a valid HTTPS URL.

   `SjumpsCF=`*jumpstart-files-URL*
   This setting points to the Solaris Flash directory on the install server that contains the `rules.ok`file, profile file, and begin and finish scripts. Make sure that this URL matches the path to the custom JumpStart files that you created in "To Create the Profile" on page 203 and "To Create the `rules` File" on page 205.

   For WAN installations that use HTTPS, set the value to a valid HTTPS URL.

4. **Save the file to a directory that is accessible to the WAN boot server.**

   For administration purposes, you might want to save the file to the appropriate client directory in the `/etc/netboot` directory on the WAN boot server.

5. **Change the permissions on the system configuration file to 600.**

```
# chmod 600 /path/system-conf-file
```

   path Specifies the path to the directory that contains the system configuration file.

   system-conf-file Specifies the name of the system configuration file.

Example 11–12 System Configuration File for WAN Boot Installation Over HTTPS

In the following example, the WAN boot programs check for the `sysidcfg` and custom JumpStart files on the web server `https://www.example.com` on port 1234. The web server uses secure HTTP to encrypt data and files during the installation.

The `sysidcfg` and custom JumpStart files are located in the `flash` subdirectory of the document root directory `/opt/apache/htdocs`.

```
SsysidCF=https://www.example.com:1234/flash
SjumpsCF=https://www.example.com:1234/flash
```

Example 11–13 System Configuration File for Insecure WAN Boot Installation

In the following example, the WAN boot programs check for the `sysidcfg` and custom JumpStart files on the web server `http://www.example.com`. The web server uses HTTP, so the data and files are not protected during the installation.

The sysidcfg and custom JumpStart files are located in the flash subdirectory of the document root directory /opt/apache/htdocs.

```
SsysidCF=http://www.example.com/flash
SjumpsCF=http://www.example.com/flash
```

More Information

Continuing the WAN Boot Installation

After you create the system configuration file, create the wanboot.conf file. For instructions, see "To Create the wanboot.conf File" on page 210.

▼ To Create the wanboot.conf File

The wanboot.conf file is a plain text configuration file that the WAN boot programs use to perform a WAN installation. The wanboot-cgi program, the boot file system, and the WAN boot miniroot all use the information included in the wanboot.conf file to install the client machine.

Save the wanboot.conf file in the appropriate client subdirectory in the /etc/netboot hierarchy on the WAN boot server. For information about how to define the scope of your WAN boot installation with the /etc/netboot hierarchy, see "Creating the /etc/netboot Hierarchy on the WAN Boot Server" on page 189.

If the WAN boot server is running the Solaris 10 OS, a sample wanboot.conf file is located in /etc/netboot/wanboot.conf.sample. You can use this sample as a template for your WAN boot installation.

You must include the following information in the wanboot.conf file.

Type of Information	Description
WAN boot server information	■ Path to wanboot program on the WAN boot server ■ URL of wanboot-cgi program on WAN boot server
Install server information	■ Path to WAN boot miniroot on the install server ■ Path to system configuration file on the WAN boot server that specifies location of sysidcfg and custom JumpStart files
Security information	■ Signature type for the WAN boot file system or WAN boot miniroot ■ Encryption type for the WAN boot file system ■ Whether the server should be authenticated during the WAN boot installation ■ Whether the client should be authenticated during the WAN boot installation

Type of Information	Description
Optional information	■ Additional hosts that might need to be resolved for the client during a WAN boot installation ■ URL to the bootlog-cgi script on the logging server

You specify this information by listing parameters with associated values in the following format.

parameter=value

For detailed information about wanboot.conf file parameters and syntax, see "wanboot.conf File Parameters and Syntax" on page 258.

To create the wanboot.conf file, follow these steps.

Steps

1. **Assume the same user role as the web server user on the WAN boot server.**

2. **Create the wanboot.conf text file.**

   You can create a new text file that is named wanboot.conf, or use the sample file that is located in /etc/netboot/wanboot.conf.sample. If you use the sample file, rename the file wanboot.conf after you add parameters.

3. **Type the wanboot.conf parameters and values for your installation.**

   For detailed descriptions of wanboot.conf parameters and values, see "wanboot.conf File Parameters and Syntax" on page 258.

4. **Save the wanboot.conf file to the appropriate subdirectory of the /etc/netboot hierarchy.**

   For information about how to create the /etc/netboot hierarchy, see "Creating the /etc/netboot Hierarchy on the WAN Boot Server" on page 189.

5. **Validate the wanboot.conf file.**

   ```
   # bootconfchk /etc/netboot/path-to-wanboot.conf/wanboot.conf
   ```

   path-to-wanboot.conf Specifies the path to the client's wanboot.conf file on the WAN boot server

   - If the wanboot.conf file is structurally valid, the bootconfchk command returns an exit code of 0.
   - If the wanboot.conf file is invalid, the bootconfchk command returns a nonzero exit code.

6. **Change the permissions on the wanboot.conf file to 600.**

   ```
   # chmod 600 /etc/netboot/path-to-wanboot.conf/wanboot.conf
   ```

Example 11–14 wanboot.conf File for WAN Boot Installation Over HTTPS

The following wanboot.conf file example includes configuration information for a WAN installation that uses secure HTTP. The wanboot.conf file also indicates that a 3DES encryption key is used in this installation.

```
boot_file=/wanboot/wanboot.s10_sparc
root_server=https://www.example.com:1234/cgi-bin/wanboot-cgi
root_file=/miniroot/miniroot.s10_sparc
signature_type=sha1
encryption_type=3des
server_authentication=yes
client_authentication=no
resolve_hosts=
boot_logger=https://www.example.com:1234/cgi-bin/bootlog-cgi
system_conf=sys-conf.s10-sparc
```

This wanboot.conf file specifies the following configuration.

boot_file=/wanboot/wanboot.s10_sparc
The second level boot program is named wanboot.s10_sparc. This program is located in the /wanboot directory in the WAN boot server's document root directory.

root_server=https://www.example.com:1234/cgi-bin/wanboot-cgi
The location of the wanboot-cgi program on the WAN boot server is https://www.example.com:1234/cgi-bin/wanboot-cgi. The https portion of the URL indicates that this WAN boot installation uses secure HTTP.

root_file=/miniroot/miniroot.s10_sparc
The WAN boot miniroot is named miniroot.s10_sparc. This miniroot is located in the /miniroot directory in the WAN boot server's document root directory.

signature_type=sha1
The wanboot.s10_sparc program and the WAN boot file system are signed with a HMAC SHA1 hashing key.

encryption_type=3des
The wanboot.s10_sparc program and the boot file system are encrypted with a 3DES key.

server_authentication=yes
The server is authenticated during the installation.

client_authentication=no
The client is not authenticated during the installation.

resolve_hosts=
No additional host names are needed to perform the WAN installation. All required files and information are located in the document root directory on the WAN boot server.

boot_logger=https://www.example.com:1234/cgi-bin/bootlog-cgi
(Optional) Booting and installation log messages are recorded on the WAN boot server by using secure HTTP.

For instructions on how to set up a logging server for your WAN boot installation, see "(Optional) To Configure the WAN Boot Logging Server" on page 193.

system_conf=sys-conf.s10-sparc
The system configuration file that contains the locations of the sysidcfg and JumpStart files is located in a subdirectory of the /etc/netboot hierarchy. The system configuration file is named sys-conf.s10-sparc.

Example 11–15 wanboot.conf File for Insecure WAN Boot Installation

The following wanboot.conf file example includes configuration information for a less secure WAN boot installation that uses HTTP. This wanboot.conf file also indicates that the installation does not use an encryption key or a hashing key.

```
boot_file=/wanboot/wanboot.s10_sparc
root_server=http://www.example.com/cgi-bin/wanboot-cgi
root_file=/miniroot/miniroot.s10_sparc
signature_type=
encryption_type=
server_authentication=no
client_authentication=no
resolve_hosts=
boot_logger=http://www.example.com/cgi-bin/bootlog-cgi
system_conf=sys-conf.s10-sparc
```

This wanboot.conf file specifies the following configuration.

boot_file=/wanboot/wanboot.s10_sparc
The second level boot program is named wanboot.s10_sparc. This program is located in the /wanboot directory in the WAN boot server's document root directory.

root_server=http://www.example.com/cgi-bin/wanboot-cgi
The location of the wanboot-cgi program on the WAN boot server is http://www.example.com/cgi-bin/wanboot-cgi. This installation does not use secure HTTP.

root_file=/miniroot/miniroot.s10_sparc
The WAN boot miniroot is named miniroot.s10_sparc. This miniroot is located in the /miniroot subdirectory in the WAN boot server's document root directory.

signature_type=
The wanboot.s10_sparc program and the WAN boot file system are not signed with a hashing key.

encryption_type=
The wanboot.s10_sparc program and the boot file system are not encrypted.

server_authentication=no
The server is not authenticated with keys or certificates during the installation.

client_authentication=no
The client is not authenticated with keys or certificates during the installation.

`resolve_hosts=`
No additional host names are needed to perform the installation. All required files and information are located in the document root directory on the WAN boot server.

`boot_logger=http://www.example.com/cgi-bin/bootlog-cgi`
(Optional) Booting and installation log messages are recorded on the WAN boot server.

For instructions on how to set up a logging server for your WAN boot installation, see "(Optional) To Configure the WAN Boot Logging Server" on page 193.

`system_conf=sys-conf.s10-sparc`
The system configuration file that contains the locations of the `sysidcfg` and JumpStart files is named `sys-conf.s10-sparc`. This file is located in the appropriate client subdirectory of the `/etc/netboot` hierarchy.

More Information

Continuing the WAN Boot Installation

After you create the `wanboot.conf` file, you can optionally configure a DHCP server to support WAN boot. For instructions, see "(Optional) Providing Configuration Information With a DHCP Server" on page 214.

If you do not want to use a DHCP server in your WAN boot installation, see "To Check the `net` Device Alias in the Client OBP" on page 218 to continue the WAN boot installation.

See Also For detailed descriptions of `wanboot.conf` parameters and values, see "`wanboot.conf` File Parameters and Syntax" on page 258 and the man page `wanboot.conf`(4).

(Optional) Providing Configuration Information With a DHCP Server

If you use a DHCP server on your network, you can configure the DHCP server to supply the following information.

- Proxy server's IP address
- Location of the `wanboot-cgi` program

You can use the following DHCP vendor options in your WAN boot installation.

`SHTTPproxy` Specifies the IP address of the network's proxy server

`SbootURI` Specifies the URL of the `wanboot-cgi` program on the WAN boot server

For information about setting these vendor options on a Solaris DHCP server, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.

For detailed information about setting up a Solaris DHCP server, see Chapter 13, "Configuring the DHCP Service (Tasks)," in *System Administration Guide: IP Services*.

To continue with your WAN boot installation, see Chapter 12.

CHAPTER 12

SPARC: Installing With WAN Boot (Tasks)

This chapter describes how to perform a WAN boot installation on a SPARC based client. For information about how to prepare for a WAN boot installation, see Chapter 11.

This chapter describes the following tasks.

- "Preparing the Client for a WAN Boot Installation" on page 218
- "Installing the Client" on page 225

Task Map: Installing a Client With WAN Boot

The following table lists the tasks you need to perform to install a client over a WAN.

TABLE 12–1 Task Map: Performing a WAN Boot Installation

Task	Description	For Instructions
Prepare the network for a WAN boot installation.	Set up the servers and files that are required to perform a WAN boot installation.	Chapter 11
Verify that the `net` device alias is set correctly in the client OBP.	Use the `devalias` command to verify that the `net` device alias is set to the primary network interface.	"To Check the `net` Device Alias in the Client OBP" on page 218

TABLE 12–1 Task Map: Performing a WAN Boot Installation *(Continued)*

Task	Description	For Instructions
Provide keys to the client	Provide keys to the client by setting OBP variables or entering key values during the installation. This task is required for secure installation configurations. For insecure installations that check data integrity, complete this task to provide the HMAC SHA1 hashing key to the client.	"Installing Keys on the Client" on page 220
Install the client over a wide area network.	Choose the appropriate method to install your client.	"To Perform a Noninteractive WAN Boot Installation" on page 227 "To Perform an Interactive WAN Boot Installation" on page 229 "To Perform a WAN Boot Installation With a DHCP Server" on page 232 "To Perform a WAN Boot Installation With Local CD Media" on page 234

Preparing the Client for a WAN Boot Installation

Before you install the client system, prepare the client by performing the following tasks.

- "To Check the `net` Device Alias in the Client OBP" on page 218
- "Installing Keys on the Client" on page 220

▼ To Check the `net` Device Alias in the Client OBP

To boot the client from the WAN with the `boot net`, the `net` device alias must be set to the client's primary network device. On most systems, this alias is already set correctly. However, if the alias is not set to the network device you want to use, you must change the alias.

For more information about setting device aliases, see "The Device Tree" in *OpenBoot 3.x Command Reference Manual*.

Follow these steps to check the `net` device alias on the client.

Steps

1. **Become superuser on the client.**
2. **Bring the system to run level 0.**

   ```
    # init 0
   ```

   The `ok` prompt is displayed.
3. **At the `ok` prompt, check device aliases that are set in the OBP.**

   ```
   ok devalias
   ```

   The `devalias` command outputs information that is similar to the following example.

   ```
   screen                   /pci@1f,0/pci@1,1/SUNW,m64B@2
   net                      /pci@1f,0/pci@1,1/network@c,1
   net2                     /pci@1f,0/pci@1,1/network@5,1
   disk                     /pci@1f,0/pci@1/scsi@8/disk@0,0
   cdrom                    /pci@1f,0/pci@1,1/ide@d/cdrom@0,0:f
   keyboard                 /pci@1f,0/pci@1,1/ebus@1/su@14,3083f8
   mouse                    /pci@1f,0/pci@1,1/ebus@1/su@14,3062f8
   ```

   - If the `net` alias is set to the network device you wan to use during the installation, you do not need to reset the alias. Go to "Installing Keys on the Client" on page 220 to continue your installation.
   - If the `net` alias is not set to the network device you want to use, you must reset the alias. Continue.
4. **Set the `net` device alias.**

   Choose one of the following commands to set the `net` device alias.

   - To set the `net` device alias for this installation only, use the `devalias` command.

     ```
     ok devalias net device-path
     ```

     `net` *device-path*   Assigns the device *device-path* to the `net` alias
   - To permanently set the `net` device alias, use the `nvalias` command.

     ```
     ok nvalias net device-path
     ```

     `net` *device-path*   Assigns the device *device-path* to the `net` alias

Example 12–1 Checking and Resetting the `net` Device Alias

The following commands show how to check and reset the `net` device alias.

Check the device aliases.

```
ok devalias
screen                   /pci@1f,0/pci@1,1/SUNW,m64B@2
net                      /pci@1f,0/pci@1,1/network@c,1
net2                     /pci@1f,0/pci@1,1/network@5,1
disk                     /pci@1f,0/pci@1/scsi@8/disk@0,0
cdrom                    /pci@1f,0/pci@1,1/ide@d/cdrom@0,0:f
keyboard                 /pci@1f,0/pci@1,1/ebus@1/su@14,3083f8
mouse                    /pci@1f,0/pci@1,1/ebus@1/su@14,3062f8
```

If you want to use the /pci@1f,0/pci@1,1/network@5,1 network device, type the following command.

```
ok devalias net /pci@1f,0/pci@1,1/network@5,1
```

More Information

Continuing the WAN Boot Installation

After you check the net device alias, see the appropriate section to continue the installation.

- If you are using a hashing key and an encryption key in your installation, see "Installing Keys on the Client" on page 220.
- If you are performing a less secure installation without keys, see "Installing the Client" on page 225.

Installing Keys on the Client

For a more secure WAN boot installation or an insecure installation with data integrity checking, you must install keys on the client. By using a hashing key and an encryption key, you can protect the data that is transmitted to the client. You can install these keys in the following ways.

- Set OBP variables – You can assign key values to OBP network boot argument variables before you boot the client. These keys can then be used for future WAN boot installations of the client.
- Enter the key values during the boot process – You can set key values at the wanboot program boot> prompt. If you use this method to install keys, the keys are only used for the current WAN boot installation.

You can also install keys in the OBP of a running client. If you want to install keys on a running client, the system must be running the Solaris 9 12/03 OS, or compatible version.

When you install keys on your client, ensure that the key values are not transmitted over an insecure connection. Follow your site's security policies to ensure the privacy of the key values.

- For instructions about how to assign key values to OBP network boot argument variables, see "To Install Keys in the Client OBP" on page 221.

- For instructions about how to install keys during the boot process, see "To Perform an Interactive WAN Boot Installation" on page 229.
- For instructions about how to install keys in the OBP of a running client, see "To Install a Hashing Key and an Encryption Key on a Running Client" on page 223.

▼ To Install Keys in the Client OBP

You can assign key values to OBP network boot argument variables before you boot the client. These keys can then be used for future WAN boot installations of the client.

To install keys in the client OBP, follow these steps.

If you want to assign key values to OBP network boot argument variables, follow these steps.

Steps

1. **Assume the same user role as the web server user on the WAN boot server.**
2. **Display the key value for each client key.**

   ```
   # wanbootutil keygen -d -c -o net=net-ip,cid=client-ID,type=key-type
   ```

   net-ip The IP address of the client's subnet.

   client-ID The ID of the client you want to install. The client ID can be a user-defined ID or the DHCP client ID.

   key-type The key type you want to install on the client. Valid key types are `3des`, `aes`, or `sha1`.

   The hexadecimal value for the key is displayed.
3. **Repeat the previous step for each type of client key you want to install.**
4. **Bring the client system to run level 0.**

   ```
   # init 0
   ```

   The `ok` prompt is displayed.
5. **At the client `ok` prompt, set the value for the hashing key.**

   ```
   ok set-security-key wanboot-hmac-sha1 key-value
   ```

   `set-security-key` Installs the key on the client

   `wanboot-hmac-sha1` Instructs OBP to install a HMAC SHA1 hashing key

   key-value Specifies the hexadecimal string that is displayed in Step 2.

   The HMAC SHA1 hashing key is installed in the client OBP.

6. **At the client `ok` prompt, install the encryption key.**

   ```
   ok set-security-key wanboot-3des key-value
   ```

   `set-security-key` Installs the key on the client

   `wanboot-3des` Instructs OBP to install a 3DES encryption key. If you want to use an AES encryption key, set this value to `wanboot-aes`.

   key-value Specifies the hexadecimal string that represents the encryption key.

   The 3DES encryption key is installed in the client OBP.

   After you install the keys, you are ready to install the client. See "Installing the Client" on page 225 for instructions about how to install the client system.

7. **(Optional) Verify that the keys are set in the client OBP.**

   ```
   ok list-security-keys
   Security Keys:
            wanboot-hmac-sha1
            wanboot-3des
   ```

8. **(Optional) If you need to delete a key, type the following command.**

   ```
   ok set-security-key key-type
   ```

   key-type Specifies the type of key you need to delete. Use the value `wanboot-hmac-sha1`, `wanboot-3des`, or `wanboot-aes`.

Example 12–2 Installing Keys in the Client OBP

The following example shows how to install a hashing key and an encryption key in the client OBP.

Display the key values on the WAN boot server.

```
# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=sha1
b482aaab82cb8d5631e16d51478c90079cc1d463
# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=3des
9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04
```

The previous example uses the following information.

`net=192.168.198.0`
Specifies the IP address of the client's subnet

`cid=010003BA152A42`
Specifies the client's ID

`b482aaab82cb8d5631e16d51478c90079cc1d463`
Specifies the value of the client's HMAC SHA1 hashing key

`9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04`
Specifies the value of the client's 3DES encryption key

If you use an AES encryption key in your installation, change `wanboot-3des` to `wanboot-aes` to display the encryption key value.

Install the keys on the client system.

```
ok set-security-key wanboot-hmac-sha1 b482aaab82cb8d5631e16d51478c90079cc1d463
ok set-security-key wanboot-3des 9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04
```

The previous commands perform the following tasks.

- Installs the HMAC SHA1 hashing key with a value of `b482aaab82cb8d5631e16d51478c90079cc1d463` on the client
- Installs the 3DES encryption key with a value of `9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04` on the client

  If you use an AES encryption key in your installation, change `wanboot-3des` to `wanboot-aes`.

More Information

Continuing the WAN Boot Installation

After you install keys on your client, you are ready to install the client over the WAN. For instructions, see "Installing the Client" on page 225.

See Also

For more information about how to display key values, see the man page `wanbootutil`(1M).

▼ To Install a Hashing Key and an Encryption Key on a Running Client

You can set key values at the `wanboot` program `boot>` prompt on a running system. If you use this method to install keys, the keys are only used for the current WAN boot installation.

If you want to install a hashing key and an encryption key in the OBP of a running client, follow these steps.

Before You Begin

This procedure makes the following assumptions.

- The client system is powered on.
- The client is accessible over a secure connection, such as a secure shell (`ssh`).

Steps

1. **Assume the same user role as the web server user on the WAN boot server.**
2. **Display the key value for the client keys.**

   ```
   # wanbootutil keygen -d -c -o net=net-ip,cid=client-ID,type=key-type
   ```

net-ip The IP address of the client's subnet.

client-ID The ID of the client you want to install. The client ID can be a user-defined ID or the DHCP client ID.

key-type The key type you want to install on the client. Valid key types are `3des`, `aes`, or `sha1`.

The hexadecimal value for the key is displayed.

3. **Repeat the previous step for each type of client key you want to install.**

4. **Become superuser on the client machine.**

5. **Install the necessary keys on the running client machine.**

```
# /usr/lib/inet/wanboot/ickey -o type=key-type
> key-value
```

key-type Specifies the key type you want to install on the client. Valid key types are `3des`, `aes`, or `sha1`.

key-value Specifies the hexadecimal string that is displayed in Step 2.

6. **Repeat the previous step for each type of client key you want to install.**

After you install the keys, you are ready to install the client. See "Installing the Client" on page 225 for instructions about how to install the client system.

Example 12–3 Installing Keys in the OBP of a Running Client System

The following example shows how to install keys in the OBP of a running client.

Display the key values on the WAN boot server.

```
# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=sha1
b482aaab82cb8d5631e16d51478c90079cc1d463
# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=3des
9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04
```

The previous example uses the following information.

`net=192.168.198.0`
Specifies the IP address of the client's subnet

`cid=010003BA152A42`
Specifies the client's ID

`b482aaab82cb8d5631e16d51478c90079cc1d463`
Specifies the value of the client's HMAC SHA1 hashing key

`9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04`
Specifies the value of the client's 3DES encryption key

If you use an AES encryption key in your installation, change type=3des to type=aes to display the encryption key value.

Install the keys in the OBP of the running client.

```
# /usr/lib/inet/wanboot/ickey -o type=sha1 b482aaab82cb8d5631e16d51478c90079cc1d463
# /usr/lib/inet/wanboot/ickey -o type=3des 9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04
```

The previous commands perform the following tasks.

- Installs a HMAC SHA1 hashing key with a value of b482aaab82cb8d5631e16d51478c90079cc1d463 on the client
- Installs a 3DES encryption key with a value of 9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04 on the client

More Information

Continuing the WAN Boot Installation

After you install keys on your client, you are ready to install the client over the WAN. For instructions, see “Installing the Client” on page 225.

See Also

For more information about how to display key values, see the man page wanbootutil(1M).

For additional information about how to install keys on a running system, see ickey(1M).

Installing the Client

When you finish preparing your network for a WAN boot installation, you can choose from the following ways to install the system.

TABLE 12–2 Methods to Install the Client

Method	Description	Instructions
Noninteractive installation	Use this installation method if you want to install keys on the client and set the client configuration information before you boot the client.	■ To install keys on the client before the installation, see "Installing Keys on the Client" on page 220. ■ To perform a noninteractive installation, see "To Perform a Noninteractive WAN Boot Installation" on page 227.
Interactive installation	Use this installation method if you want to set the client configuration information during the boot process.	"To Perform an Interactive WAN Boot Installation" on page 229
Installing with a DHCP server	Use this installation method if you configured the network DHCP server to provide client configuration information during the installation.	■ To configure a DHCP server to support a WAN boot installation, see "(Optional) Providing Configuration Information With a DHCP Server" on page 214. ■ To use a DHCP server during your installation, see "To Perform a WAN Boot Installation With a DHCP Server" on page 232.
Installing with local CD media	If your client OBP does not support WAN boot, boot the client from a local copy of the Solaris 10 Software CD.	■ To determine if the client OBP supports WAN boot, see "To Check the Client OBP for WAN Boot Support" on page 186. ■ To install the client with a local copy of the Solaris 10 Software CD, see "To Perform a WAN Boot Installation With Local CD Media" on page 234.

▼ To Perform a Noninteractive WAN Boot Installation

Use this installation method if you prefer to install keys and set client configuration information before you install the client. You can then boot the client from the WAN and perform an unattended installation.

This procedure assumes that you have either installed keys in the client's OBP, or that you are performing an insecure installation. For information about installing keys on the client before your installation, see "Installing Keys on the Client" on page 220.

Steps 1. **If the client system is currently running, bring the system to run level 0.**

```
# init 0
```

The ok prompt is displayed.

2. **At the ok prompt on the client system, set the network boot argument variables in OBP.**

```
ok setenv network-boot-arguments  host-ip=client-IP,
router-ip=router-ip,subnet-mask=mask-value,
hostname=client-name,http-proxy=proxy-ip:port,
file=wanbootCGI-URL
```

Note – The line breaks in this command sample are included for formatting purposes only. Do not enter a carriage return until you finish typing the command.

`setenv network-boot-arguments`	Instructs the OBP to set the following boot arguments
`host-ip=`*client-IP*	Specifies the IP address of the client
`router-ip=`*router-ip*	Specifies the IP address of the network router
`subnet-mask=`*mask-value*	Specifies the subnet mask value
`hostname=`*client-name*	Specifies the host name of the client
(Optional) `http-proxy=`*proxy-ip:port*	Specifies the IP address and port of the network's proxy server
`file=`*wanbootCGI-URL*	Specifies the URL of the `wanboot-cgi` program on the web server

3. **Boot the client.**

```
ok boot net - install
```

`net - install` Instructs the client to use the network boot argument variables to boot from the WAN

The client installs over the WAN. If the WAN boot programs do not find all the necessary installation information, the `wanboot` program prompts to provide the missing information. Type the additional information at the prompt.

Example 12–4 Noninteractive WAN Boot Installation

In the following example, the network boot argument variables for the client system `myclient` are set before the machine is booted. This example assumes that a hashing key and encryption key are already installed on the client. For information about installing keys before you boot from the WAN, see “Installing Keys on the Client” on page 220.

```
ok setenv network-boot-arguments host-ip=192.168.198.136,
router-ip=192.168.198.129,subnet-mask=255.255.255.192
hostname=myclient,file=http://192.168.198.135/cgi-bin/wanboot-cgi
ok boot net - install
Resetting ...




Sun Blade 100 (UltraSPARC-IIe), No Keyboard
Copyright 1998-2003 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.x.build_28, 256 MB memory installed, Serial #50335475.
Ethernet address 0:3:ba:e:f3:75, Host ID: 83000ef3.




Rebooting with command: boot net - install
Boot device: /pci@1f,0/network@c,1  File and args: - install
```

The following variables are set.

- The client IP address is set to 192.168.198.136.
- The client’s router IP address is set to 192.168.198.129.
- The client’s subnet mask is set to 255.255.255.192.
- The client’s host name is set to `seahag`.
- The `wanboot-cgi` program is located at `http://192.168.198.135/cgi-bin/wanboot-cgi`.

See Also For more information about how to set network boot arguments, see `set`(1).

For more information about how to boot a system, see `boot`(1M).

▼ To Perform an Interactive WAN Boot Installation

Use this installation method if you want to install keys and set client configuration information at the command line during the installation.

This procedure assumes that you are using HTTPS in your WAN installation. If you are performing an insecure installation that does not use keys, do not display or install the client keys.

Steps

1. **Assume the same user role as the web server user on the WAN boot server.**

2. **Display the key value for each client key.**

   ```
   # wanbootutil keygen -d -c -o net=net-ip,cid=client-ID,type=key-type
   ```

   net-ip The IP address of the subnet for the client you want to install.

   client-ID The ID of the client you want to install. The client ID can be a user-defined ID or the DHCP client ID.

   key-type The key type you want to install on the client. Valid key types are `3des`, `aes`, or `sha1`.

   The hexadecimal value for the key is displayed.

3. **Repeat the previous step for each type of client key you are installing.**

4. **If the client system is currently running, bring the client to run level 0.**

5. **At the `ok` prompt on the client system, set the network boot argument variables in OBP.**

   ```
   ok setenv network-boot-arguments  host-ip=client-IP,router-ip=router-ip,
   subnet-mask=mask-value,hostname=client-name,
   http-proxy=proxy-ip:port,bootserver=wanbootCGI-URL
   ```

   Note – The line breaks in this command sample are included for formatting purposes only. Do not enter a carriage return until you finish typing the command.

   `setenv network-boot-arguments` Instructs the OBP to set the following boot arguments

   `host-ip=`*client-IP* Specifies the IP address of the client

   `router-ip=`*router-ip* Specifies the IP address of the network router

   `subnet-mask=`*mask-value* Specifies the subnet mask value

   `hostname=`*client-name* Specifies the host name of the client

(Optional) `http-proxy=`*proxy-ip:port* — Specifies the IP address and port of the network's proxy server

`bootserver=`*wanbootCGI-URL* — Specifies the URL of the `wanboot-cgi` program on the web server

---

Note – The URL value for the`bootserver` variable must not be an HTTPS URL. The URL must start with `http://`.

---

6. **At the client `ok` prompt, boot the system.**

```
ok boot net -o prompt - install
```

`net -o prompt - install` — Instructs the client to boot and install from the network. The `wanboot` program prompts the user to enter client configuration information at the `boot>` prompt.

The `boot>` prompt is displayed.

7. **Install the encryption key.**

```
boot> 3des=key-value
```

`3des=`*key-value* — Specifies the hexadecimal string of the 3DES key that is displayed in Step 2.

If you use an AES encryption key, use the following format for this command.

```
boot> aes=key-value
```

8. **Install the hashing key.**

```
boot> sha1=key-value
```

`sha1=`*key-value* — Specifies the hashing key value that is displayed in Step 2.

9. **Type the following command to continue the boot process.**

```
boot> go
```

The client installs over the WAN.

10. **If prompted, type client configuration information on the command line.**

If the WAN boot programs do not find all the necessary installation information, the `wanboot` program prompts to provide the missing information. Type the additional information at the prompt.

Example 12–5 Interactive WAN Boot Installation

In the following example, the `wanboot` program prompts you to set the key values for the client system during the installation.

Display the key values on the WAN boot server.

```
# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=sha1
b482aaab82cb8d5631e16d51478c90079cc1d463
# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=3des
9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04
```

The previous example uses the following information.

`net=192.168.198.0`
Specifies the IP address of the client's subnet

`cid=010003BA152A42`
Specifies the client's ID

`b482aaab82cb8d5631e16d51478c90079cc1d463`
Specifies the value of the client's HMAC SHA1 hashing key

`9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04`
Specifies the value of the client's 3DES encryption key

If you use an AES encryption key in your installation, change `type=3des` to `type=aes` to display the encryption key value.

Set the network boot argument variables in the OBP on the client.

```
ok setenv network-boot-arguments host-ip=192.168.198.136,
router-ip=192.168.198.129,subnet-mask=255.255.255.192,hostname=myclient,
bootserver=http://192.168.198.135/cgi-bin/wanboot-cgi
```

The following variables are set.

- The client IP address is set to 192.168.198.136.
- The client's router IP address is set to 192.168.198.129.
- The client's subnet mask is set to 255.255.255.192.
- The client's host name is set to `myclient`.
- The `wanboot-cgi` program is located at `http://192.168.198.135/cgi-bin/wanboot-cgi`.

Boot and install the client.

```
ok boot net -o prompt - install
Resetting ...


Sun Blade 100 (UltraSPARC-IIe), No Keyboard
Copyright 1998-2003 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.x.build_28, 256 MB memory installed, Serial #50335475.
```

```
Ethernet address 0:3:ba:e:f3:75, Host ID: 83000ef3.



Rebooting with command: boot net -o prompt
Boot device: /pci@1f,0/network@c,1  File and args: -o prompt

boot> 3des=9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04

boot> sha1=b482aaab82cb8d5631e16d51478c90079cc1d463

boot> go
```

The previous commands perform the following tasks.

- Installs the 3DES encryption key with the value 9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04 on the client
- Installs the HMAC SHA1 hashing key with the value b482aaab82cb8d5631e16d51478c90079cc1d463 on the client
- Starts the installation

See Also For more information about how to display key values, see wanbootutil(1M).

For more information about how to set network boot arguments, see set(1).

For more information about how to boot a system, see boot(1M).

▼ To Perform a WAN Boot Installation With a DHCP Server

If you configured a DHCP server to support WAN boot options, you can use the DHCP server to provide client configuration information during the installation. For more information about configuring a DHCP server to support a WAN boot installation, see "(Optional) Providing Configuration Information With a DHCP Server" on page 214.

This procedure makes the following assumptions.

- The client system is running.
- You have either installed keys on the client, or you are performing an insecure installation.

  For information about installing keys on the client before your installation, see "Installing Keys on the Client" on page 220.
- You have configured your DHCP server to support the SbootURI and SHTTPproxy WAN boot options.

  These options enable the DHCP server to provide the configuration information that is required by WAN boot.

For information about how to set installation options on your DHCP server, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.

Steps

1. **If the client system is currently running, bring the system to run level 0.**

   ```
   # init 0
   ```

   The `ok` prompt is displayed.

2. **At the `ok` prompt on the client system, set the network boot argument variables in OBP.**

   ```
   ok setenv network-boot-arguments dhcp,hostname=client-name
   ```

   `setenv network-boot-arguments` Instructs the OBP to set the following boot arguments

   `dhcp` Instructs the OBP to use the DHCP server to configure the client

   `hostname=`*client-name* Specifies the host name you want to assign to the client

3. **Boot the client from the network.**

   ```
   ok boot net - install
   ```

   `net - install` Instructs the client to use the network boot argument variables to boot from the WAN

   The client installs over the WAN. If the WAN boot programs do not find all the necessary installation information, the `wanboot` program prompts to provide the missing information. Type the additional information at the prompt.

Example 12–6 WAN Boot Installation With a DHCP Server

In the following example, the DHCP server on the network provides client configuration information. This sample requests the host name `myclient` for the client.

```
ok setenv network-boot-arguments dhcp, hostname=myclient

ok boot net - install
Resetting ...



Sun Blade 100 (UltraSPARC-IIe), No Keyboard
Copyright 1998-2003 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.x.build_28, 256 MB memory installed, Serial #50335475.
Ethernet address 0:3:ba:e:f3:75, Host ID: 83000ef3.
```

```
Rebooting with command: boot net - install
Boot device: /pci@1f,0/network@c,1  File and args: - install
```

See Also For more information about how to set network boot arguments, see `set`(1).

For more information about how to boot a system, see `boot`(1M).

For more information about how to configure a DHCP server, see "(Optional) Providing Configuration Information With a DHCP Server" on page 214.

▼ To Perform a WAN Boot Installation With Local CD Media

If your client's OBP does not support WAN boot, you can install with a Solaris 10 Software - 1 CD inserted in the client's CD-ROM drive. When you use a local CD, the client retrieves the `wanboot` program from the local media, rather than from the WAN boot server.

This procedure assumes that you are using HTTPS in your WAN installation. If you are performing an insecure installation, do not display or install the client keys.

Follow these steps to perform a WAN boot installation from a local CD.

Steps

1. **Assume the same user role as the web server user on the WAN boot server.**
2. **Display the key value for each client key.**

   ```
   # wanbootutil keygen -d -c -o net=net-ip,cid=client-ID,type=key-type
   ```

   net-ip The network IP address for the client you are installing.

   client-ID The ID of the client you are installing. The client ID can be a user-defined ID or the DHCP client ID.

   key-type The key type you are installing on the client. Valid key types are `3des`, `aes`, or `sha1`.

   The hexadecimal value for the key is displayed.
3. **Repeat the previous step for each type of client key you are installing.**
4. **On the client system, insert the Solaris 10 Software - 1 CD in the CD-ROM drive.**
5. **Power on the client system.**
6. **Boot the client from the CD.**

   ```
   ok boot cdrom -o prompt -F wanboot - install
   ```

`cdrom` Instructs the OBP to boot from the local CD-ROM

`-o prompt` Instructs the `wanboot` program to prompt the user to enter client configuration information

`-F wanboot` Instructs the OBP to load the `wanboot` program from the CD-ROM

`- install` Instructs the client to perform a WAN boot installation

The client's OBP loads the `wanboot` program from the Solaris 10 Software - 1 CD. The `wanboot` program boots the system, and the `boot>` prompt is displayed.

7. **Type the encryption key value.**

```
boot> 3des=key-value
```

`3des=`*key-value* Specifies the hexadecimal string of the 3DES key that is displayed in step Step 2.

If you use an AES encryption key, use the following format for this command.

```
boot> aes=key-value
```

8. **Type the hashing key value.**

```
boot> sha1=key-value
```

`sha1=`*key-value* Specifies the hexadecimal string that represents the hashing key value that is displayed in step Step 2.

9. **Set the network interface variables.**

```
boot> variable=value[,variable=value*]
```

Type the following variable and value pairs at the `boot>` prompt.

`host-ip=`*client-IP* Specifies the IP address of the client.

`router-ip=`*router-ip* Specifies the IP address of the network router.

`subnet-mask=`*mask-value* Specifies the subnet mask value.

`hostname=`*client-name* Specifies the host name of the client.

(Optional) `http-proxy=`*proxy-ip:port* Specifies the IP address and port number of the network's proxy server.

`bootserver=`*wanbootCGI-URL* Specifies the URL of the `wanboot-cgi` program on the web server.

Note – The URL value for thebootserver variable must not be an HTTPS URL. The URL must start with http://.

You can enter these variables in the following ways.

- Type one variable and value pair at the boot> prompt, then press the Return key.

```
boot> host-ip=client-IP
boot> subnet-mask=mask-value
```

- Type all the variable and value pairs on one boot> prompt line, then press the Return key. Type commas to separate each variable and value pair.

```
boot> host-ip=client-IP,subnet-mask=mask-value,
router-ip=router-ip,hostname=client-name,
http-proxy=proxy-ip:port,bootserver=wanbootCGI-URL
```

10. **Type the following command to continue the boot process.**

```
boot> go
```

The client installs over the WAN. If the WAN boot programs do not find all the necessary installation information, the wanboot program prompts to provide the missing information. Type the additional information at the prompt.

Example 12–7 Installing With Local CD Media

In the following example, the wanboot program on a local CD prompts you to set the network interface variables for the client during the installation.

Display the key values on the WAN boot server.

```
# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=sha1
b482aaab82cb8d5631e16d51478c90079cc1d463
# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=3des
9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04
```

The previous example uses the following information.

net=192.168.198.0
Specifies the IP address of the client's subnet

cid=010003BA152A42
Specifies the client's ID

b482aaab82cb8d5631e16d51478c90079cc1d463
Specifies the value of the client's HMAC SHA1 hashing key

`9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04`
Specifies the value of the client's 3DES encryption key

If you use an AES encryption key in your installation, change `type=3des` to `type=aes` to display the encryption key value.

Boot and install the client.

```
ok boot cdrom -o prompt -F wanboot - install
Resetting ...


Sun Blade 100 (UltraSPARC-IIe), No Keyboard
Copyright 1998-2003 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.x.build_28, 256 MB memory installed, Serial #50335475.
Ethernet address 0:3:ba:e:f3:75, Host ID: 83000ef3.



Rebooting with command: boot cdrom -F wanboot - install
Boot device: /pci@1f,0/network@c,1  File and args: -o prompt

boot> 3des=9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04

boot> sha1=b482aaab82cb8d5631e16d51478c90079cc1d463

boot> host-ip=192.168.198.124

boot> subnet-mask=255.255.255.128

boot> router-ip=192.168.198.1

boot> hostname=myclient
boot> client-id=010003BA152A42

boot> bootserver=http://192.168.198.135/cgi-bin/wanboot-cgi

boot> go
```

The previous commands perform the following tasks.

- Enters the 3DES encryption key with the value `9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04` on the client
- Enters the HMAC SHA1 hashing key with the value `b482aaab82cb8d5631e16d51478c90079cc1d463` on the client
- Sets the client IP address to 192.168.198.124
- Sets the client's subnet mask to 255.255.255.128
- Sets the client's router IP address to 192.168.198.1
- Sets the client's host name to `myclient`
- Sets the client ID to 010003BA152A42
- Sets the location of the `wanboot-cgi` program to `http://192.168.198.135/cgi-bin/wanboot-cgi/`

See Also For more information about how to display key values, see `wanbootutil`(1M).

For more information about how to set network boot arguments, see `set`(1).

For more information about how to boot a system, see `boot`(1M).

CHAPTER 13

SPARC: Installing With WAN Boot (Examples)

This chapter provides an example of setting up and installing client systems over a wide area network (WAN). The examples in this chapter describe how to perform a secure WAN boot installation over an HTTPS connection.

Sample Site Setup

Figure 13–1 shows the site setup for this example.

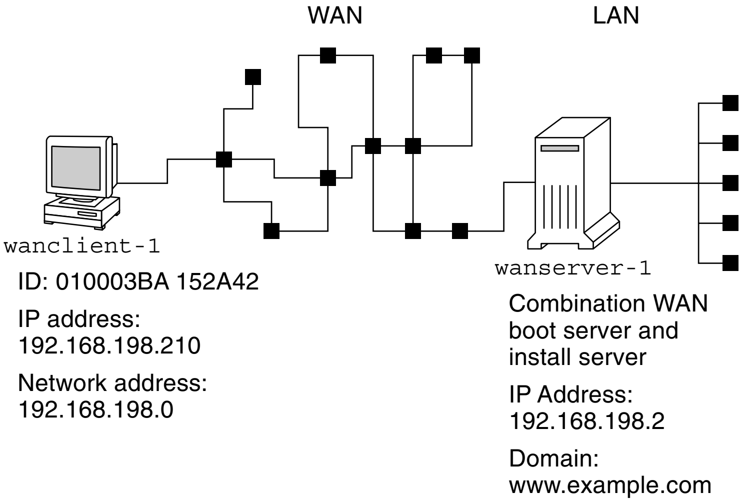

FIGURE 13–1 Sample Site for WAN Boot Installation

This sample site has the following characteristics.

- The server `wanserver-1` is to be configured as a WAN boot server and an install server.
- The IP address of `wanserver-1` is 192.168.198.2.
- The domain name of `wanserver-1` is `www.example.com`.
- `wanserver-1` is running the Solaris 10 OS.
- `wanserver-1` is running the Apache web server. The Apache software on `wanserver-1` is configured to support HTTPS.
- The client to be installed is named `wanclient-1`.
- `wanclient-1` is an UltraSPARCII system.
- The client ID for `wanclient-1` is 010003BA152A42.
- The IP address of `wanclient-1` is 192.168.198.210.
- The IP address of the client's subnet is 192.168.198.0.
- The client system `wanclient-1` has Internet access, but is not directly connected to the network that includes `wanserver-1`.
- `wanclient-1` is a new system that is to be installed with the Solaris 10 software.

Create the Document Root Directory

To store the installation files and data, set up the following directories in the document root directory (/opt/apache/htdocs) on wanserver-1.

- Solaris Flash directory

  ```
  wanserver-1# mkdir -p /opt/apache/htdocs/flash/
  ```

- WAN boot miniroot directory

  ```
  wanserver-1# mkdir -p /opt/apache/htdocs/miniroot/
  ```

- wanboot program directory

  ```
  wanserver-1# mkdir -p /opt/apache/htdocs/wanboot/
  ```

Create the WAN Boot Miniroot

Use the setup_install_server(1M) with the -w option to copy the WAN boot miniroot and the Solaris software image to the /export/install/Solaris_10 directory of wanserver-1.

Insert the Solaris 10 Software media in the media drive that is attached to wanserver-1. Type the following commands.

```
wanserver-1# mkdir -p /export/install/sol_10_sparc
wanserver-1# cd /cdrom/cdrom0/s0/Solaris_10/Tools
wanserver-1# ./setup_install_server -w /export/install/sol_10_sparc/miniroot \
/export/install/sol_10_sparc
```

Move the WAN boot miniroot to the document root directory (/opt/apache/htdocs/) of the WAN boot server.

```
wanserver-1# mv /export/install/sol_10_sparc/miniroot/miniroot \
/opt/apache/htdocs/miniroot/miniroot.s10_sparc
```

Check the Client OBP for WAN Boot Support

Determine that the client OBP supports WAN boot by typing the following command on the client system.

```
# eeprom | grep network-boot-arguments
network-boot-arguments: data not available
```

In the previous example, the network-boot-arguments: data not available output indicates that the client OBP supports WAN boot.

Install the wanboot Program on the WAN Boot Server

To install the wanboot program on the WAN boot server, copy the program from the Solaris 10 Software media to the WAN boot server's document root directory.

Insert the Solaris 10 DVD or the Solaris 10 Software - 1 CD in the media drive that is attached to wanserver-1 and type the following commands.

```
wanserver-1# cd /cdrom/cdrom0/s0/Solaris_10/Tools/Boot/platform/sun4u/
wanserver-1# cp wanboot /opt/apache/htdocs/wanboot/wanboot.s10_sparc
```

Create the /etc/netboot Hierarchy

Create the wanclient-1 subdirectories of the /etc/netboot directory on the WAN boot server. The WAN boot installation programs retrieve configuration and security information from this directory during the installation.

wanclient-1 is located on the subnet 192.168.198.0, and has a client ID of 010003BA152A42. To create the appropriate subdirectory of /etc/netboot for wanclient-1, perform the following tasks.

- Create the /etc/netboot directory.
- Change the permissions of the /etc/netboot directory to 700.
- Change the ownership of the /etc/netboot directory to the owner of the web server process.
- Assume the same user role as the web server user.
- Create a subdirectory of /etc/netboot that is named after the subnet (192.168.198.0).
- Create a subdirectory of the subnet directory that is named after the client ID.
- Change the permissions of the /etc/netboot subdirectories to 700.

```
wanserver-1# cd /
wanserver-1# mkdir /etc/netboot/
wanserver-1# chmod 700 /etc/netboot
wanserver-1# chown nobody:admin /etc/netboot
wanserver-1# exit
wanserver-1# su nobody
Password:
nobody# mkdir -p /etc/netboot/192.168.198.0/010003BA152A42
nobody# chmod 700 /etc/netboot/192.168.198.0
```

```
nobody# chmod 700 /etc/netboot/192.168.198.0/010003BA152A42
```

Copy the wanboot-cgi Program to the WAN Boot Server

On systems that are running the Solaris 10 OS, the wanboot-cgi program is located in the /usr/lib/inet/wanboot/ directory. To enable the WAN boot server to transmit the installation data, copy the wanboot-cgi program to the cgi-bin directory in the web server software directory.

```
wanserver-1# cp /usr/lib/inet/wanboot/wanboot-cgi \
/opt/apache/cgi-bin/wanboot-cgi
wanserver-1# chmod 755 /opt/apache/cgi-bin/wanboot-cgi
```

(Optional) Configure the WAN Boot Server as a Logging Server

By default, all WAN boot logging messages are displayed on the client system. This default behavior enables you to quickly debug any installation issues.

If you want to view the boot and installation messages on the WAN boot server, copy the bootlog-cgi script to the cgi-bin directory on wanserver-1.

```
wanserver-1# cp /usr/lib/inet/wanboot/bootlog-cgi /opt/apache/cgi-bin/
wanserver-1# chmod 755 /opt/apache/cgi-bin/bootlog-cgi
```

Configure the WAN Boot Server to Use HTTPS

To use HTTPS in your WAN boot installation, you must enable SSL support in the web server software. You must also install a digital certificate on the WAN boot server. This example assumes that the Apache web server on wanserver-1 is configured to use SSL. This example also assumes that a digital certificate and a certificate authority that establish the identity of wanserver-1 are already installed on wanserver-1.

For examples about how to configure your web server software to use SSL, see you web server documentation.

Provide the Trusted Certificate to the Client

By requiring the server to authenticate itself to the client, you protect the data that is transmitted from the server to the client over HTTPS. To enable server authentication, you provide a trusted certificate to the client. The trusted certificate enables the client to verify the identity of the server during the installation.

To provide the trusted certificate to the client, assume the same user role as the web server user. Then, split the certificate to extract a trusted certificate. Then, insert the trusted certificate in the client's `truststore` file in the `/etc/netboot` hierarchy.

In this example, you assume the web server user role of `nobody`. Then, you split the server PKCS#12 certificate that is named `cert.p12`, and insert the trusted certificate in `/etc/netboot` directory for `wanclient-1`.

```
wanserver-1# su nobody
Password:
wanserver-1# wanbootutil p12split -i cert.p12 -t \
/etc/netboot/192.168.198.0/010003BA152A42/truststore
```

(Optional) Use Private Key and Certificate for Client Authentication

To further protect your data during the installation, you might want to require `wanclient-1` to authenticate itself to `wanserver-1`. To enable client authentication in your WAN boot installation, insert a client certificate and private key in the client subdirectory of the `/etc/netboot` hierarchy.

To provide a private key and certificate to the client, perform the following tasks.

- Assume the same user role as the web server user
- Split the PKCS#12 file into a private key and a client certificate
- Insert the certificate in the client's `certstore` file
- Insert the private key in the client's `keystore` file

In this example, you assume the web server user role of nobody. Then, you split the server PKCS#12 certificate that is named cert.p12. You insert certificate in the /etc/netboot hierarchy for wanclient-1. You then insert the private key that you named wanclient.key in the client's keystore file.

```
wanserver-1# su nobody
Password:
wanserver-1# wanbootutil p12split -i cert.p12 -c \
/etc/netboot/192.168.198.0/010003BA152A42/certstore -k wanclient.key
wanserver-1# wanbootutil keymgmt -i -k wanclient.key \
-s  /etc/netboot/192.168.198.0/010003BA152A42/keystore \
-o type=rsa
```

Create the Keys for the Server and the Client

To protect the data transmitted between the server and client, you create a hashing key and an encryption key. The server uses the hashing key to protect the integrity of the wanboot program. The server uses the encryption key to encrypt the configuration and installation data. The client uses the hashing key to check the integrity of the downloaded wanboot program. The client uses the encryption key to decrypt the data during the installation.

First, you assume the same user role as the web server user. In this example, the web server user role is nobody.

```
wanserver-1# su nobody
Password:
```

Then, you use the wanbootutil keygen command to create a master HMAC SHA1 key for wanserver-1.

```
wanserver-1# wanbootutil keygen -m
```

Then, create a hashing key and an encryption key for wanclient-1.

```
wanserver-1# wanbootutil keygen -c -o net=192.168.198.0,cid=010003BA152A42,type=sha1
wanserver-1# wanbootutil keygen -c -o net=192.168.198.0,cid=010003BA152A42,type=3des
```

The previous command creates a HMAC SHA1 hashing key and a 3DES encryption key for wanclient-1. 192.168.198.0 specifies the subnet of wanclient-1, and 010003BA152A42 specifies the client ID of wanclient-1.

Create the Solaris Flash Archive

In this example, you create your Solaris Flash archive by cloning the wanserver-1 master system. The archive is named sol_10_sparc, and is copied exactly from the master system. The archive is an exact duplicate of the master system. The archive is stored in sol_10_sparc.flar. You save the archive in the flash/archives subdirectory of the document root directory on the WAN boot server.

```
wanserver-1# flar create -n sol_10_sparc \
/opt/apache/htdocs/flash/archives/sol_10_sparc.flar
```

Create the sysidcfg File

To preconfigure the wanclient-1 system, specify keywords and values in the sysidcfg file. Save this file in the appropriate subdirectory of the document root directory of wanserver-1.

EXAMPLE 13–1 sysidcfg File for client-1 System

The following is an example of a sysidcfg file for wanclient-1. The host name, IP address, and netmask of these systems have been preconfigured by editing the name service. This file is located in the /opt/apache/htdocs/flash/ directory.

```
network_interface=primary {hostname=wanclient-1
                           default_route=192.168.198.1
                           ip_address=192.168.198.210
                           netmask=255.255.255.0
                           protocol_ipv6=no}
timezone=US/Central
system_locale=C
terminal=xterm
timeserver=localhost
name_service=NIS {name_server=matter(192.168.254.254)
                  domain_name=leti.example.com
                  }
security_policy=none
```

Create the Client's Profile

For the wanclient-1 system, create a profile that is named wanclient_1_prof. The wanclient_1_prof file contains the following entries, which define the Solaris 10 software to be installed on the wanclient-1 system.

```
# profile keywords         profile values
# ----------------         -------------------
install_type               flash_install
archive_location           https://192.168.198.2/flash/archives/sol_10_sparc.flar
partitioning               explicit
filesys                    c0t1d0s0 4000 /
filesys                    c0t1d0s1 512 swap
filesys                    c0t1d0s7 free /export/home
```

The following list describes some of the keywords and values from this example.

install_type
The profile installs a Solaris Flash archive on the clone system. All files are overwritten as in an initial installation.

archive_location
The compressed Solaris Flash archive is retrieved from wanserver-1.

partitioning
The file system slices are determined by the filesys keywords, value explicit. The size of root (/) is based on the size of the Solaris Flash archive. The size of swap is set to the necessary size and is installed on c0t1d0s1. /export/home is based on the remaining disk space. /export/home is installed on c0t1d0s7.

Create and Validate the rules File

The custom JumpStart programs use the rules file to select the correct installation profile for the wanclient-1 system. Create a text file that is named rules. Then, add keywords and values to this file.

The IP address of the wanclient-1 system is 192.168.198.210, and the netmask is 255.255.255.0. Use the network rule keyword to specify the profile that the custom JumpStart programs should use to install wanclient-1.

```
network 192.168.198.0 - wanclient_1_prof -
```

This rules file instructs the custom JumpStart programs to use the wanclient_1_prof to install the Solaris 10 software on wanclient-1.

Name this rule file wanclient_rule.

After you create the profile and the rules file, you run the check script to verify that the files are valid.

```
wanserver-1# ./check -r wanclient_rule
```

If the check script does not find any errors, the script creates the rules.ok file.

Save the rules.ok file in the /opt/apache/htdocs/flash/ directory.

Create the System Configuration File

Create a system configuration file that lists the locations of the sysidcfg file and the custom JumpStart files on the install server. Save this file in a directory that is accessible to the WAN boot server.

In the following example, the wanboot-cgi program looks for the sysidcfg and custom JumpStart files in the document root directory of the WAN boot server. The domain name of the WAN boot server is https://www.example.com. The WAN boot server is configured to use secure HTTP, so the data and files are protected during the installation.

In this example, the system configuration file is named sys-conf.s10-sparc, and the file is saved in the /etc/netboot hierarchy on the WAN boot server. The sysidcfg and custom JumpStart files are located in the flash subdirectory of the document root directory.

```
SsysidCF=https://www.example.com/flash/
SjumpsCF=https://www.example.com/flash/
```

Create the wanboot.conf File

WAN boot uses the configuration information that is included in the wanboot.conf file to install the client machine. Create the wanboot.conf file in a text editor. Save the file to the appropriate client subdirectory in the /etc/netboot hierarchy on the WAN boot server.

The following wanboot.conf file for wanclient-1 includes configuration information for a WAN installation that uses secure HTTP. This file also instructs WAN boot to use a HMAC SHA1 hashing key and a 3DES encryption key to protect data.

```
boot_file=/wanboot/wanboot.s10_sparc
root_server=https://www.example.com/cgi-bin/wanboot-cgi
root_file=/miniroot/miniroot.s10_sparc
signature_type=sha1
encryption_type=3des
server_authentication=yes
client_authentication=no
resolve_hosts=
boot_logger=
system_conf=sys-conf.s10-sparc
```

This wanboot.conf file specifies the following configuration.

boot_file=/wanboot/wanboot.s10_sparc
The wanboot program is named wanboot.s10_sparc. This program is located in the wanboot directory in the document root directory on wanserver-1.

root_server=https://www.example.com/cgi-bin/wanboot-cgi
The location of the wanboot-cgi program on wanserver-1 is https://www.example.com/cgi-bin/wanboot-cgi. The https portion of the URL indicates that this WAN boot installation uses secure HTTP.

root_file=/miniroot/miniroot.s10_sparc
The WAN boot miniroot is named miniroot.s10_sparc. The miniroot is located in the miniroot directory in the document root directory on wanserver-1.

signature_type=sha1
The wanboot program and the WAN boot file system are signed by using a HMAC SHA1 hashing key.

encryption_type=3des
The wanboot program and the WAN boot file system are encrypted with a 3DES key.

server_authentication=yes
The server is authenticated during the installation.

client_authentication=no
The client is not authenticated during the installation.

Note – If you performed the tasks in “(Optional) Use Private Key and Certificate for Client Authentication” on page 244, set this parameter as client_authentication=yes resolve_hosts=

No additional host names are needed to perform the WAN installation. All the host names that are required by the wanboot-cgi program are specified in the wanboot.conf file and the client certificate.

boot_logger=

Booting and installation log messages are displayed on the system console. If you configured the logging server in "(Optional) Configure the WAN Boot Server as a Logging Server" on page 243, and you want WAN boot messages to appear on the WAN boot server as well, set this parameter to boot_logger=https://www.example.com/cgi-bin/bootlog-cgi.

system_conf=sys-conf.s10-sparc

The system configuration file that specifies the locations of the sysidcfg and JumpStart files is located in the sys-conf.s10-sparc file in the /etc/netboot hierarchy on wanserver-1.

In this example, you save the wanboot.conf file in the /etc/netboot/192.168.198.0/010003BA152A42 directory on wanserver-1.

Check the net Device Alias in OBP

To boot the client from the WAN with the boot net, the net device alias must be set to the client's primary network device. At the client ok prompt, type the devalias command to verify that the net alias is set to the primary network device /pci@1f,0/pci@1,1/network@c,1.

```
ok devalias
screen                   /pci@1f,0/pci@1,1/SUNW,m64B@2
net                      /pci@1f,0/pci@1,1/network@c,1
net2                     /pci@1f,0/pci@1,1/network@5,1
disk                     /pci@1f,0/pci@1/scsi@8/disk@0,0
cdrom                    /pci@1f,0/pci@1,1/ide@d/cdrom@0,0:f
keyboard                 /pci@1f,0/pci@1,1/ebus@1/su@14,3083f8
mouse                    /pci@1f,0/pci@1,1/ebus@1/su@14,3062f8
```

In the previous output example, the primary network device /pci@1f,0/pci@1,1/network@c,1 is assigned to the net alias. You do not need to reset the alias.

Install Keys on the Client

In “Create the Keys for the Server and the Client” on page 245, you created the hashing key and encryption key to protect your data during the installation. To enable the client to decrypt the data transmitted from `wanserver-1` during the installation, install these keys on `wanclient-1`.

On `wanserver-1`, display the key values.

```
wanserver-1# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=sha1
b482aaab82cb8d5631e16d51478c90079cc1d463
wanserver-1# wanbootutil keygen -d -c -o net=192.168.198.0,cid=010003BA152A42,type=3des
9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04
```

The previous example uses the following information.

`net=192.168.198.0`
Specifies the IP address of the client’s subnet

`cid=010003BA152A42`
Specifies the client’s ID

`b482aaab82cb8d5631e16d51478c90079cc1d463`
Specifies the value of the client’s HMAC SHA1 hashing key

`9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04`
Specifies the value of the client’s 3DES encryption key

If you use an AES encryption key in your installation, change `type=3des` to `type=aes` to display the encryption key value.

At the `ok` prompt on `wanclient-1`, install the keys.

```
ok set-security-key wanboot-hmac-sha1  b482aaab82cb8d5631e16d51478c90079cc1d463
ok set-security-key wanboot-3des  9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04
```

The previous commands perform the following tasks.

- Installs the HMAC SHA1 hashing key with a value of `b482aaab82cb8d5631e16d51478c90079cc1d463` on `wanclient-1`
- Installs the 3DES encryption key with a value of `9ebc7a57f240e97c9b9401e9d3ae9b292943d3c143d07f04` on `wanclient-1`

Install the Client

You can perform an unattended installation by setting network boot argument variables for `wanclient-1` at the `ok` prompt, and then booting the client.

```
ok setenv network-boot-arguments host-ip=192.168.198.210,
router-ip=192.168.198.1,subnet-mask=255.255.255.0,hostname=wanclient-1,
file=http://192.168.198.2/cgi-bin/wanboot-cgi
ok boot net - install
Resetting ...




Sun Blade 100 (UltraSPARC-IIe), No Keyboard
Copyright 1998-2003 Sun Microsystems, Inc.  All rights reserved.
OpenBoot 4.x.build_28, 256 MB memory installed, Serial #50335475.
Ethernet address 0:3:ba:e:f3:75, Host ID: 83000ef3.



Rebooting with command: boot net - install
Boot device: /pci@1f,0/network@c,1  File and args: - install



<time unavailable> wanboot progress: wanbootfs: Read 68 of 68 kB (100%)
<time unavailable> wanboot info: wanbootfs: Download complete
Fri Jun 20 09:16:06 wanboot progress: miniroot: Read 166067 of 166067 kB (100%)
Fri Jun 20Tue Apr 15 09:16:06 wanboot info: miniroot: Download complete
SunOS Release 5.10 Version WANboot10:04/11/03 64-bit
Copyright 1983-2003 Sun Microsystems, Inc.  All rights reserved.
Use is subject to license terms.
Configuring devices.
```

The following variables are set.

- The client IP address is set to 192.168.198.210.
- The client's router IP address is set to 192.168.198.1
- The client's subnet mask is set to 255.255.255.0
- The client's host name is set to `wanclient-1`
- The `wanboot-cgi` program is located at `http://192.168.198.2/cgi-bin/wanboot-cgi`

The client installs over the WAN. If the `wanboot` program does not find all the necessary installation information, you might be prompted to provide the missing information at the command line.

CHAPTER 14

WAN Boot (Reference)

This chapter briefly describes the commands and files you use to perform a WAN installation.

- "WAN Boot Installation Commands" on page 253
- "OBP Commands" on page 256
- "System Configuration File Settings and Syntax" on page 257
- "`wanboot.conf` File Parameters and Syntax" on page 258

WAN Boot Installation Commands

The following tables describe the commands you use to perform a WAN boot installation.

- Table 14–1
- Table 14–2

TABLE 14–1 Preparing the WAN Boot Installation and Configuration Files

Task and Description	Command
Copy the Solaris installation image to *install-dir-path*, and copy the WAN boot miniroot to *wan-dir-path* on the install server's local disk.	`setup_install_server -w` *wan-dir-path install-dir-path*

TABLE 14–1 Preparing the WAN Boot Installation and Configuration Files *(Continued)*

Task and Description	Command
Create a Solaris Flash archive that is named *name*`.flar`. ■ *name* is the name of the archive ■ *optional-parameters* are optional parameters you can use to customize the archive ■ *document-root* is the path to the document root directory on the install server ■ *filename* is the name of the archive	`flar create – n` *name* `[`*optional-parameters*`]` *document-root*`/flash/`*filename*
Check the validity of the custom JumpStart `rules` file that is named *rules*.	`./check -r` *rules*
Check the validity of the `wanboot.conf` file. ■ *net-ip* is the IP address of the client's subnet. ■ *client-ID* can be a user-defined ID or the DHCP client ID.	`bootconfchk /etc/netboot/`*net-ip*`/`*client-ID*`/wanboot.conf`
Check for WAN boot installation support in the client OBP.	`eeprom | grep network-boot-arguments`

TABLE 14–2 Preparing the WAN Boot Security Files

Task and Description	Command
Create a master HMAC SHA1 key for the WAN boot server.	`wanbootutil keygen -m`
Create a HMAC SHA1 hashing key for the client. ■ *net-ip* is the IP address of the client's subnet. ■ *client-ID* can be a user-defined ID or the DHCP client ID.	`wanbootutil keygen -c -o net=`*net-ip*`,cid=`*client-ID*`,type=sha1`

TABLE 14–2 Preparing the WAN Boot Security Files *(Continued)*

Task and Description	Command
Create an encryption key for the client. ■ *net-ip* is the IP address of the client's subnet. ■ *client-ID* can be a user-defined ID or the DHCP client ID. ■ *key-type* is either `3des` or `aes`.	`wanbootutil keygen -c -o` `net=`*net-ip*`,cid=`*client-ID*`,type=`*key-type*

Split a PKCS#12 certificate file and insert the certificate in the client's `truststore`.
■ *p12cert* is the name of the PKCS#12 certificate file.
■ *net-ip* is the IP address of the client's subnet.
■ *client-ID* can be a user-defined ID or the DHCP client ID.
	`wanbootutil p12split -i` *p12cert* `-t`
	`/etc/netboot/`*net-ip*`/`*client-ID*`/truststore`

Split a PKCS#12 certificate file and insert the client certificate in the client's `certstore`
■ *p12cert* is the name of the PKCS#12 certificate file.
■ *net-ip* is the IP address of the client's subnet.
■ *client-ID* can be a user-defined ID or the DHCP client ID.
■ *keyfile* is the name of the client's private key.
	`wanbootutil p12split -i` *p12cert* `-c`
	`/etc/netboot/`*net-ip*`/`*client-ID*`/certstore`
	`-k` *keyfile*

Insert the client private key from a split PKCS#12 file in the client's `keystore`.
■ *keyfile* is the name of the client's private key.
■ *net-ip* is the IP address of the client's subnet.
■ *client-ID* can be a user-defined ID or a DHCP client ID.
	`wanbootutil keymgmt -i -k` *keyfile* `-s`
	`/etc/netboot/`*net-ip*`/`*client-ID*`/keystore`
	`-o type=rsa`

Display the value of a HMAC SHA1 hashing key.
■ *net-ip* is the IP address of the client's subnet.
■ *client-ID* can be a user-defined ID or the DHCP client ID.
	`wanbootutil keygen -d -c -o`
	`net=`*net-ip*`,cid=`*client-ID*`,type=sha1`

TABLE 14–2 Preparing the WAN Boot Security Files *(Continued)*

Task and Description	Command
Display the value of an encryption key. ■ *net-ip* is the IP address of the client's subnet. ■ *client-ID* can be a user-defined ID or the DHCP client ID. ■ *key-type* is either `3des` or `aes`.	`wanbootutil keygen -d -c -o net=`*net-ip*`,cid=`*client-ID*`,type=`*key-type*
Insert a hashing key or an encryption key on a running system. *key-type* can have a value of `sha1`, `3des`, or `aes`.	`/usr/lib/inet/wanboot/ickey -o type=`*key-type*

OBP Commands

The following table lists the OBP commands that you type at the client `ok` prompt to perform a WAN boot installation.

TABLE 14–3 OBP Commands for a WAN Boot Installation

Task and Description	OBP Command
Begin an unattended WAN boot installation.	`boot net – install`
Begin an interactive WAN boot installation.	`boot net –o prompt - install`
Begin a WAN boot installation from a local CD.	`boot cdrom –F wanboot - install`
Install a hashing key before you begin a WAN boot installation.*key-value* is the hexadecimal value of the hashing key.	`set-security-key wanboot-hmac-sha1` *key-value*
Install an encryption key before you begin a WAN boot installation. ■ *key-type* is either `wanboot-3des` or `wanboot-aes`. ■ *key-value* is the hexadecimal value of the encryption key.	`set-security-key` *key-type key-value*
Verify that key values are set in OBP.	`list-security-keys`

TABLE 14–3 OBP Commands for a WAN Boot Installation *(Continued)*

Task and Description	OBP Command
Set client configuration variables before you begin your WAN boot installation. ■ *client-IP* is the IP address of the client. ■ *router-ip* is the IP address of the network router. ■ *mask-value* is the subnet mask value. ■ *client-name* is the host name of the client. ■ *proxy-ip* is the IP address of the network's proxy server. ■ *wanbootCGI-path* is the path to the `wanbootCGI` programs on the web server.	`setenv network-boot-arguments host-ip=`*client-IP*`,router-ip=`*router-ip*`, subnet-mask=`*mask-value*`, hostname=`*client-name*`, http-proxy=`*proxy-ip*`, file=`*wanbootCGI-path*
Check the network device alias.	`devalias`
Set the network device alias, where *device-path* is the path to the primary network device.	■ To set the alias for the current installation only, type `devalias net` *device-path*. ■ To permanently set the alias, type `nvvalias net` *device-path*.

System Configuration File Settings and Syntax

The system configuration file enables you to direct the WAN boot installation programs to the following files.

- `sysidcfg`
- `rules.ok`
- Custom JumpStart profile

The system configuration file is a plain text file, and must be formatted in the following pattern.

setting=value

The `system.conf` file must contain the following settings.

`SsysidCF=`*sysidcfg-file-URL*
This setting points to the directory on the install server that contains the `sysidcfg` file. For WAN installations that use HTTPS, set the value to a valid HTTPS URL.

`SjumpsCF=`*jumpstart-files-URL*
This setting points to the custom JumpStart directory that contains the `rules.ok` and profile files. For WAN installations that use HTTPS, set the value to a valid HTTPS URL.

You can store the system.conf in any directory that is accessible to the WAN boot server.

wanboot.conf File Parameters and Syntax

The wanboot.conf file is a plain-text configuration file that the WAN boot installation programs use to perform a WAN installation. The following programs and files use the information included in the wanboot.conf file to install the client machine.

- wanboot-cgi program
- WAN boot file system
- WAN boot miniroot

Save the wanboot.conf file in the appropriate client subdirectory in the /etc/netboot hierarchy on the WAN boot server. For information on how to define the scope of your WAN boot installation with the /etc/netboot hierarchy, see "Creating the /etc/netboot Hierarchy on the WAN Boot Server" on page 189.

You specify information in the wanboot.conf file by listing parameters with associated values in the following format.

parameter=value

Parameter entries cannot span lines. You can include comments in the file by preceding the comments with the # character.

For detailed information about the wanboot.conf file, see the man page wanboot.conf(4).

You must set the following parameters in the wanboot.conf file.

boot_file=*wanboot-path*

This parameter specifies the path to the wanboot program. The value is a path relative to the document root directory on the WAN boot server.

```
boot_file=/wanboot/wanboot.s10_sparc
```

root_server=*wanbootCGI-URL*/wanboot-cgi

This parameter specifies the URL of the wanboot-cgi program on the WAN boot server.

- Use an HTTP URL if you are performing a WAN boot installation without client or server authentication.

  ```
  root_server=http://www.example.com/cgi-bin/wanboot-cgi
  ```

- Use an HTTPS URL if you are performing a WAN boot installation with server authentication, or server and client authentication.

  ```
  root_server=https://www.example.com/cgi-bin/wanboot-cgi
  ```

`root_file=`*miniroot-path*
This parameter specifies the path to the WAN boot miniroot on the WAN boot server. The value is a path relative to the document root directory on the WAN boot server.

```
root_file=/miniroot/miniroot.s10_sparc
```

`signature_type=sha1 |` *empty*
This parameter specifies the type of hashing key to use to check the integrity of the data and files that are transmitted.

- For WAN boot installations that use a hashing key to protect the `wanboot` program, set this value to `sha1`.

  ```
  signature_type=sha1
  ```

- For insecure WAN installations that do not use a hashing key, leave this value blank.

  ```
  signature_type=
  ```

`encryption_type=3des | aes |` *empty*
This parameter specifies the type of encryption to use to encrypt the `wanboot` program and WAN boot file system.

- For WAN boot installations that use HTTPS, set this value to `3des` or `aes` to match the key formats you use. You must also set the `signature_type` keyword value to `sha1`.

  ```
  encryption_type=3des
  ```

  or

  ```
  encryption_type=aes
  ```

- For an insecure WAN boot installations that do not use encryption key, leave this value blank.

  ```
  encryption_type=
  ```

`server_authentication=yes | no`
This parameter specifies if the server should be authenticated during the WAN boot installation.

- For WAN boot installations with server authentication or server and client authentication, set this value to `yes`. You must also set the value of `signature_type` to `sha1`, `encryption_type` to `3des` or `aes`, and the URL of `root_server` to an HTTPS value.

  ```
  server_authentication=yes
  ```

- For insecure WAN boot installations that do not use server authentication or server and client authentication, set this value to no. You can also leave the value blank.

  ```
  server_authentication=no
  ```

client_authentication=yes | no

This parameter specifies if the client should be authenticated during a WAN boot installation.

- For WAN boot installations with server and client authentication, set this value to yes. You must also set the value of signature_type to sha1, encryption_type to 3des or aes, and the URL of root_server to an HTTPS value.

  ```
  client_authentication=yes
  ```

- For WAN boot installations that do not use client authentication, set this value to no. You can also leave the value blank.

  ```
  client_authentication=no
  ```

resolve_hosts=*hostname* | *empty*

This parameter specifies additional hosts that need to be resolved for the wanboot-cgi program during the installation.

Set the value to the host names of systems that are not specified previously in the wanboot.conf file or in a client certificate.

- If all the required hosts are listed in the wanboot.conf file or the client certificate, leave this value blank.

  ```
  resolve_hosts=
  ```

- If specific hosts are not listed in the wanboot.conf file or the client certificate, set the value to these host names.

  ```
  resolve_hosts=seahag,matters
  ```

boot_logger=*bootlog-cgi-path* | *empty*

This parameter specifies the URL to the bootlog-cgi script on the logging server.

- To record boot or installation log messages on a dedicated logging server, set the value to the URL of the bootlog-cgi script on the logging server.

  ```
  boot_logger=http://www.example.com/cgi-bin/bootlog-cgi
  ```

- To display boot and installation messages on the client console, leave this value blank.

  ```
  boot_logger=
  ```

system_conf=system.conf | *custom-system-conf*

This parameter specifies the path to the system configuration file that includes the location of sysidcfg and custom JumpStart files.

Set the value to the path to the sysidcfg and custom JumpStart files on the web server.

```
system_conf=sys.conf
```

PART IV Appendixes

This part provides reference information.

APPENDIX **A**

Troubleshooting (Tasks)

This chapter contains a list of specific error messages and general problems you might encounter when installing Solaris 10 software. The chapter also explains how to fix the problems. Start by using this list of sections to determine where in the installation process the problem occurred.

- "Problems With Setting Up Network Installations" on page 265
- "Problems With Booting a System" on page 266
- "Initial Installation of the Solaris Operating Environment" on page 272
- "Upgrading the Solaris Operating Environment" on page 274

Note – When you see the phrase "bootable media," this means the Solaris installation program and JumpStart installation method.

Problems With Setting Up Network Installations

`Unknown client "`*host_name*`"`

Cause: The *host_name* argument in the `add_install_client` command is not a host in the name service.

Description: Add the host *host_name* to the name service and execute the `add_install_client` command again.

Problems With Booting a System

Booting From Media, Error Messages

`le0: No carrier - transceiver cable problem`

Cause: The system is not connected to the network.

Solution: If this is a nonnetworked system, ignore this message. If this is a networked system, ensure that the Ethernet cabling is attached securely.

`The file just loaded does not appear to be executable`

Cause: The system cannot find the proper media for booting.

Solution: Verify that the system has been set up properly to install the Solaris 10 software from the network from an install server. The following are examples of checks you can make.

- If you copied the images of the Solaris 10 DVD or the Solaris 10 Software CDs to the install server, ensure that you specified the correct platform group for the system when you set it up.
- If you are using DVD or CD media, ensure that the Solaris 10 DVD or Solaris 10 Software - 1 CD is mounted and accessible on the install server.

`boot: cannot open <`*filename*`>` (*SPARC based systems only*)

Cause: This error occurs when you override the location of the `boot -file` by explicitly setting it.

Note – *filename* is a variable for the name of the file affected.

Solution: Follow these instructions:

- Reset the `boot -file` in the PROM to " " (blank).
- Ensure that the diag-switch is set to off and to true.

`Can't boot from file/device`

Cause: The installation media cannot find the bootable media.

Solution: Ensure that the following conditions are met:

- The DVD-ROM or CD-ROM drive is installed properly and turned on.
- Solaris 10 DVD or the Solaris 10 Software - 1 CD is inserted into the drive.
- The disc is free of damage or dirt.

`WARNING: clock gained` *`xxx`* `days -- CHECK AND RESET DATE!` (*SPARC based systems only*)

Description: This is an informational message.

Solution: Ignore the message and continue with the installation.

`Not a UFS file system` (*x86 based systems only*)

Cause: When Solaris 10 software was installed (either through the Solaris installation program or custom JumpStart), no boot disk was selected. You now must use the Solaris 10 Device Configuration Assistant diskette or edit the BIOS to boot the system.

Solution: Follow these instructions:

- Insert the Solaris 10 Device Configuration Assistant diskette into the system's boot diskette drive (usually drive A). For information about accessing the Solaris 10 Device Configuration Assistant diskette, see Appendix C.
- If you cannot use the bootable media, go into the BIOS and select the BIOS to boot. See your BIOS documentation for instructions.

Booting From Media, General Problems

`The system does not boot.`

Description: When initially setting up a custom JumpStart server, you might encounter boot problems that do not return an error message. To verify information about the system and how the system is booting, run the boot command with the `-v` option. When you use the `-v` option, the boot command displays verbose debugging information about the screen.

Note – If this flag is not given, the messages are still printed, but the output is directed to the system logfile. For more information, see `syslogd`(1M).

Solution: For SPARC based systems, at the `ok` prompt, type the following command.

```
ok boot net -v - install
```

For x86 based systems, when the installation program prompts you to "Select type of installation," type the following command.

```
b - -v install
```

`Boot from DVD media fails on systems with Toshiba SD—M 1401 DVD-ROM`

Description: If your system has a Toshiba SD-M1401 DVD-ROM with firmware revision 1007, the system cannot boot from the Solaris 10 DVD.

Solution: Apply patch 111649–03, or later version, to update the Toshiba SD-M1401 DVD-ROM drive's firmware. The patch 111649–03 is available on http://sunsolve.sun.com.

`The system hangs or panics when nonmemory PC cards are inserted.` (*x86 based systems only*)

Cause: Nonmemory PC cards cannot use the same memory resources that are used by other devices.

Solution: To correct this problem, see the instructions for your PC card and check for the address range.

`The IDE BIOS primary drive on your system was not detected by the Solaris 10 Device Configuration Assistant diskette during the pre-booting phase.` (*x86 based systems only*)

Solution: Follow these instructions:

- If you are using old drives, they might be unsupported. Check your hardware manufacturer's documentation.
- Make sure the ribbon and power cables are connected correctly. Check the manufacturer's documentation.
- If only one drive is attached to the controller, designate the drive as the master drive by setting jumpers. Some drives have different jumper settings for a single master, as opposed to a master operating with a slave. Connect the drive to the connector at the end of the cable to reduce signal ringing that occurs when an unused connector is dangling at the end of the cable.
- If two drives are attached to the controller, designate one drive as the master by setting jumpers (or as a master operating with a slave), and set the second drive as a slave by setting jumpers.
- If one drive is a hard disk and the second a CD-ROM drive, designate one drive as the slave drive by setting jumpers. You can designate either physical drive as the slave drive.
- If problems persist with two drives on a single controller, attach one drive at a time to verify that each drive works. Designate the drive as master or single master by setting jumpers, and use the drive connector at the end of the IDE ribbon cable to attach the drive. Verify that each drive works, then set the jumpers for the drives back to a master and slave configuration.
- If the drive is a disk drive, use the BIOS setup utility to ensure that the drive type (which indicates the number of cylinders, heads, and sectors) is configured correctly. Some BIOS software might have a feature that automatically detects the drive type.
- If the drive is a CD-ROM drive, use the BIOS setup screen to configure the drive type as a CD-ROM drive, provided the BIOS software offers this capability.
- For many systems, IDE CD-ROM drives are only recognized by MS-DOS if an MS-DOS CD-ROM driver has been installed. Try another drive.

`The IDE disk or CD-ROM drive on your system was not found by the Solaris 10 Device Configuration Assistant diskette during the pre-booting phase.` (*x86 based systems only*)

Solution: Follow these instructions:

- If disks are disabled in the BIOS, use the Solaris 10 Device Configuration Assistant diskette to boot from the hard disk. For information about accessing the Solaris 10 Device Configuration Assistant, see Appendix C.
- If the system has no disks, it might be a diskless client.

`The system hangs before displaying the system prompt.` (*x86 based systems only*)

Solution: You have hardware that is not supported. Check your hardware manufacturer's documentation.

Booting From the Network, Error Messages

`WARNING: getfile: RPC failed: error 5 (RPC Timed out).`

Description: This error occurs when you have two or more servers on a network responding to an install client's boot request. The install client connects to the wrong boot server, and the installation hangs. The following specific reasons might cause this error to occur:

Cause: *Reason 1:*`/etc/bootparams` files might exist on different servers with an entry for this install client.

Solution: *Reason 1:* Ensure that servers on the network do not have multiple `/etc/bootparams` entries for the install client. If they do have multiple entries, remove duplicate client entries in the `/etc/bootparams` file on all install servers and boot servers except the one you want the install client to use.

Cause: *Reason 2:* Multiple `/tftpboot` or `/rplboot` directory entries might exist for this install client.

Solution: *Reason 2:* Ensure that servers on the network do not have multiple `/tftpboot` or `/rplboot` directory entries for the install client. If they do have multiple entries, remove duplicate client entries from the `/tftpboot` or `/rplboot` directories on all install servers and boot servers except the one you want the install client to use.

Cause: *Reason 3:* An install client entry might exist in the `/etc/bootparams` file on a server and an entry in another `/etc/bootparams` file that enables all systems to access the profile server. Such an entry resembles the following:

```
* install_config=profile_server:path
```

A line that resembles the previous entry in the NIS or NIS+ `bootparams` table can also cause this error.

Solution: *Reason 3:* If a wildcard entry is in the name service `bootparams` map or table (for example, `* install_config=`), delete it and add it to the `/etc/bootparams` file on the boot server.

`No network boot server. Unable to install the system. See installation instructions.` (*SPARC based systems only*)

Cause: This error occurs on a system that you are attempting to install from the network. The system is not set up correctly.

Solution: Ensure that you correctly set up the system to install from the network. See "Adding Systems to Be Installed From the Network With a CD Image" on page 143.

`prom_panic: Could not mount file system` (*SPARC based systems only*)

Cause: This error occurs when you are installing Solaris from a network, but the boot software cannot locate the following:

- Solaris 10 DVD, either the DVD or a copy of the DVD image on the install server
- Solaris 10 Software - 1 CD image, either the Solaris 10 Software - 1 CD or a copy of the CD image on the install server

Solution: Ensure that the installation software is mounted and shared.

- If you are installing Solaris from the install server's DVD-ROM or CD-ROM drive, ensure that the Solaris 10 DVD or Solaris 10 Software - 1 CD is inserted in the CD-ROM drive, is mounted, and is shared in the `/etc/dfs/dfstab` file.
- If installing from a copy of the Solaris 10 DVD image or Solaris 10 Software - 1 CD image on the install server's disk, ensure that the directory path to the copy is shared in the `/etc/dfs/dfstab` file.

`Timeout waiting for ARP/RARP packet...` (*SPARC based systems only*)

Cause: *Reason 1:* The client is trying to boot from the network, but it cannot find a system that knows about the client.

Solution: *Reason 1:* Verify the system's host name is in the NIS or NIS+ name service. Also, verify the `bootparams` search order in the boot server's `/etc/nsswitch.conf` file.

For example, the following line in the `/etc/nsswitch.conf` file indicates that JumpStart or the Solaris installation program first looks in the NIS maps for `bootparams` information. If the program does not find any information, the installer looks in the boot server's `/etc/bootparams` file.

```
bootparams: nis files
```

Cause: *Reason 2:* The client's Ethernet address is not correct.

Solution: *Reason 2:* Verify that the client's Ethernet address in the install server's `/etc/ethers` file is correct.

Cause: *Reason 3:* In a custom JumpStart installation, the `add_install_client` command specifies the platform group that uses a specified server as an install server. If the wrong architecture value is used when using the `add_install_client`, this problem occurs. For example, the machine you want to install is a sun4u, but you used i86pc instead.

Solution: *Reason 3:* Rerun `add_install_client` with the correct architecture value.

`ip: joining multicasts failed on tr0 - will use link layer broadcasts for multicast` (*x86 based systems only*)

Cause: This error message is displayed when you boot a system with a token ring card. Ethernet multicast and token ring multicast do not work the same way. The driver returns this error message because an invalid multicast address was provided to it.

Solution: Ignore this error message. If multicast does not work, IP uses layer broadcasts instead and does not cause the installation to fail.

`Requesting Internet address for` *ethernet_Address* (*x86 based systems only*)

Cause: The client is trying to boot from the network, but it cannot find a system that knows about the client.

Solution: Verify the system's host name is listed in the name service. If the system's host name is listed in the NIS or NIS+ name service, and the system continues to print this error message, try rebooting.

`RPC: Timed out No bootparams (whoami) server responding; still trying...` (*x86 based systems only*)

Cause: The client is trying to boot from the network, but it cannot find a system with an entry in the `/etc/bootparams` file on the install server.

Solution: Use `add_install_client` on the install server. Using this command adds the proper entry in the `/etc/bootparams` file, enabling the client to boot from the network.

`Still trying to find a RPL server...` (*x86 based systems only*)

Cause: The system is trying to boot from the network, but the server is not set up to boot this system.

Solution: On the install server, execute `add_install_client` for the system to be installed. The `add_install_client` command sets up an `/rplboot` directory, which contains the necessary network boot program.

`CLIENT MAC ADDR: FF FF FF FF FF FF` (*network installations with DHCP only*)

Cause: The DHCP server is not configured correctly. This error might occur if the options or macros are not correctly defined in the DHCP Manager software.

Solution: In the DHCP Manager software, verify that the options and macros are correctly defined. Confirm that the Router option is defined, and that the value of the Router option is correct for the subnet you are using for the network installation.

Booting From the Network, General Problems

`The system boots from the network, but from a system other than the specified install server.`

Cause: An `/etc/bootparams` and perhaps `/etc/ethers` entry exist on another system for the client.

Solution: On the name server, update the `/etc/bootparams` entry for the system that is being installed. The entry should conform to the following syntax:

install_system `root=`*boot_server*`:`*path* `install=`*install_server*`:`*path*

Also, ensure that only one `bootparams` entry is on the subnet for the install client.

Initial Installation of the Solaris Operating Environment

`Initial installation fails`

Solution: If the Solaris installation fails, you must restart the installation. To restart the installation, boot the system from the Solaris 10 DVD, the Solaris 10 Software - 1 CD, or from the network.

You cannot uninstall the Solaris software after the software has been partially installed. You must restore your system from a backup or begin the Solaris installation process again.

`/cdrom/Solaris_10/SUNW`*xxxx*`/reloc.cpio: Broken pipe`

Description: This error message is informational and does not affect the installation. The condition occurs when a write on a pipe does not have a reading process.

Solution: Ignore the message and continue with the installation.

`WARNING: CHANGE DEFAULT BOOT DEVICE` *(x86 based systems only)*

Cause: This is an informational message. The default boot device set in the system's BIOS might be set to a device that requires you to use the Solaris 10 Device Configuration Assistant diskette to boot the system.

Solution: Continue with the installation and, if necessary, change the system's default boot device specified in the BIOS after you install the Solaris software to a device that does not require the Solaris 10 Device Configuration Assistant diskette.

▼ x86: To Check IDE Disk for Bad Blocks

IDE disk drives do not automatically map out bad blocks like other drives supported by Solaris software. Before installing Solaris on an IDE disk, you might want to perform a surface analysis on the disk. To perform surface analysis on an IDE disk, follow this procedure.

Steps

1. **Boot to the installation media in single-user mode.**

   ```
    # b -s
   ```

2. **Start the `format`(1M) program.**

   ```
   # format
   ```

3. **Specify the IDE disk drive on which you want to perform a surface analysis.**

   ```
   # cxdy
   ```

   c*x* Is the controller number

   d*y* Is the device number

4. **You need an `fdisk` partition.**

   - If a Solaris `fdisk` partition already exists, proceed to Step 5.
   - If a Solaris `fdisk` partition does not exist, use the `fdisk` command to create a Solaris partition on the disk.

   ```
   format> fdisk
   ```

5. **Type:**

   ```
   format> analyze
   ```

6. **Type:**

   ```
   analyze> config
   ```

   The current settings for a surface analysis are displayed.

   a. **If you want to change settings, type:**

      ```
      analyze> setup
      ```

7. **Type:**

   ```
   analyze> type_of_surface_analysis
   ```

   type_of_surface_analysis Is read, write, or compare

If `format` finds bad blocks, it remaps them.

8. **Type:**

   ```
   analyze> quit
   ```

9. **Do you want to specify blocks to remap?**
   - If no, go to Step 10.
   - If yes, type:

     ```
     format> repair
     ```

10. **Type:**

    ```
    quit
    ```

    The `format` program quits.

11. **To restart the media in multiuser mode, type:**

    ```
    ok b
    ```

Upgrading the Solaris Operating Environment

Upgrading, Error Messages

`No upgradable disks`

Cause: A swap entry in the `/etc/vfstab` file is causing the upgrade to fail.

Solution: Comment out the following lines in the `/etc/vfstab` file:

- All swap files and slices on disks not being upgraded
- Swap files that are no longer present
- Any unused swap slices

`usr/bin/bzczt not found`

Cause: Solaris Live Upgrade fails because of needing a patch cluster.

Solution: A patch is needed to install Solaris Live Upgrade. Go to http://sunsolve.sun.com for the patch.

```
Upgradeable Solaris root devices were found, however, no suitable
partitions to hold the Solaris install software were found.
Upgrading using the Solaris Installer is not possible. It might be
possible to upgrade using the Solaris Software 1 CDROM. (x86 based
systems only)
```

Cause: You cannot upgrade with Solaris 10 Software - 1 because you do not have enough space.

Solution: To upgrade, you can either create a swap slice that is larger than or equal to 512 Mbytes or use another method of upgrading such as the Solaris installation program from Solaris 10 DVD or a net installation image, or JumpStart.

Upgrading, General Problems

```
The upgrade option is not presented even though there is a version
of Solaris software that's upgradable on the system.
```

Cause: *Reason 1:* The `/var/sadm` directory is a symlink or it is mounted from another file system.

Solution: *Reason 1:* Move the `/var/sadm` directory into the root (`/`) or `/var` file system.

Cause: *Reason 2:* The `/var/sadm/softinfo/INST_RELEASE` file is missing.

Solution: *Reason 2:* Create a new `INST_RELEASE` file by using the following template:

```
OS=Solaris
VERSION=x
REV=0
```

x

Is the version of Solaris software on the system

Cause: *Reason 3:* SUNWusr is missing from `/var/sadm/softinfo`.

Solution: *Solution 3:* You need to do an initial installation. The Solaris software is not upgradable.

```
Couldn't shut down or initialize the md driver
```

Solution: Follow these instructions:

- If the file system is not a RAID-1 volume, comment out in the `vsftab` file.
- If the file system is a RAID-1 volume, break the mirror and reinstall. For information about unmirroring, see "Removing RAID-1 Volumes (Unmirroring)" in *Solaris Volume Manager Administration Guide*.

`The upgrade fails because the Solaris installation program cannot mount a file system.`

Cause: During an upgrade, the script attempts to mount all the file systems that are listed in the system's `/etc/vfstab` file on the root (/) file system that is being upgraded. If the installation script cannot mount a file system, it fails and exits.

Solution: Ensure that all file systems in the system's `/etc/vfstab` file can be mounted. Comment out any file systems in the `/etc/vfstab` file that cannot be mounted or that might cause the problem so that the Solaris installation program does not try to mount them during the upgrade. Any system-based file systems that contain software to be upgraded (for example, `/usr`) cannot be commented out.

`The upgrade fails`

Description: The system does not have enough space for the upgrade.

Cause: Check "Upgrading With Disk Space Reallocation" on page 33 for the space problem and see if you can fix it without using auto-layout to reallocate space.

`Problems upgrading RAID—1 volume root (/) file systems`

Solution: If you have problems upgrading when using Solaris Volume Manager RAID-1 volumes that are the root (/) file system, see Chapter 25, "Troubleshooting Solaris Volume Manager (Tasks)," in *Solaris Volume Manager Administration Guide*.

▼ To Continue Upgrading After a Failed Upgrade

The upgrade fails and the system cannot be soft-booted. The failure is for reasons beyond your control, such as a power failure or a network connection failure.

Steps

1. **Reboot the system from the Solaris 10 DVD, the Solaris 10 Software - 1 CD, or from the network.**
2. **Choose the upgrade option for installation.**

   The Solaris installation program determines if the system has been partially upgraded and continues the upgrade.

▼ System Panics When Upgrading With Solaris Live Upgrade Running Veritas VxVm

When you use Solaris Live Upgrade while upgrading and running Veritas VxVM, the system panics on reboot unless you upgrade by using the following procedure. The problem occurs if packages do not conform to Solaris advanced packaging guidelines.

Steps

1. **Create an inactive boot environment. See "Creating a New Boot Environment" in** *Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning***.**

2. **Before upgrading the inactive boot environment, you must disable the existing Veritas software on the inactive boot environment.**

   a. **Mount the inactive boot environment.**

   ```
   # lumount inactive_boot_environment_name mount_point
   ```

   For example:

   ```
   # lumount solaris8 /.alt.12345
   ```

   b. **Change to the directory that contains the `vfstab`, for example:**

   ```
   # cd /.alt.12345/etc
   ```

   c. **Make a copy of the inactive boot environment's `vfstab` file, for example:**

   ```
   # cp vfstab vfstab.501
   ```

   d. **In the copied `vfstab`, comment out all Veritas file system entries, for example:**

   ```
   #  sed  ’/vx\/dsk/s/^/#/g’  < vfstab >  vfstab.novxfs
   ```

   The first character of each line is changed to #, which makes the line a comment line. Note that this comment line is different than the system file-comment lines.

   e. **Copy the changed `vfstab` file, for example:**

   ```
   # cp vfstab.novxfs vfstab
   ```

   f. **Change directories to the inactive boot environment's system file, for example:**

   ```
   # cd /.alt.12345/etc
   ```

   g. **Make a copy of the inactive boot environment's system file, for example:**

   ```
   # cp system system.501
   ```

   h. **Comment out all "forceload:" entries that include `drv/vx`.**

   ```
   # sed ’/forceload:   drv\/vx/s/^/*/’ <system> system.novxfs
   ```

   The first character of each line is changed to *, which makes the line a command line. Note that this comment line is different than the `vfstab` file comment lines.

   i. **Change directories to the `install-db` file on the inactive boot environment, for example:**

   ```
   # cd /.alt.12345/etc
   ```

   j. **Create the Veritas `install-db` file, for example:**

   ```
   # touch vx/reconfig.d/state.d/install-db
   ```

k. **Unmount the inactive boot environment.**

```
# luumount inactive_boot_environment_name mount_point
```

3. **Upgrade the inactive boot environment. See Chapter 7, “Upgrading With Solaris Live Upgrade (Tasks),” in** *Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning*.

4. **Activate the inactive boot environment. See “Activating a Boot Environment” in** *Solaris 10 Installation Guide: Solaris Live Upgrade and Upgrade Planning*.

5. **Shut down the system.**

```
# init 0
```

6. **Boot the inactive boot environment in single-user mode:**

```
OK boot -s
```

Several messages and error messages that contain “vxvm” or “VXVM” are displayed that can be ignored. The inactive boot environment becomes active.

7. **Upgrade Veritas.**

a. **Remove the Veritas VRTSvmsa package from the system, for example:**

```
# pkgrm VRTSvmsa
```

b. **Change directories to the Veritas packages.**

```
# cd /location_of_Veritas_software
```

c. **Add the latest Veritas packages to the system:**

```
#pkgadd -d ‘pwd‘ VRTSvxvm VRTSvmsa VRTSvmdoc VRTSvmman VRTSvmdev
```

8. **Restore the original `vfstab` and system files:**

```
# cp /etc/vfstab.original /etc/vfstab
    # cp /etc/system.original /etc/system
```

9. **Reboot the system.**

```
# init 6
```

x86: Service Partition Not Created by Default on Systems With No Existing Service Partition

If you install the Solaris 10 OS on a system that does not currently include a service or diagnostic partition, the installation program might not create a service partition by default. If you want to include a service partition on the same disk as the Solaris partition, you must recreate the service partition before you install the Solaris 10 OS.

If you installed the Solaris 8 2/02 operating environment on a system with a service partition, the installation program might not have preserved the service partition. If you did not manually edit the `fdisk` boot partition layout to preserve the service partition, the installation program deleted the service partition during the installation.

Note – If you did not specifically preserve the service partition when you installed the Solaris 8 2/02 operating environment, you might not be able to recreate the service partition and upgrade to the Solaris 10 OS.

If you want to include a service partition on the disk that contains the Solaris partition, choose one of the following workarounds.

- To install the software from a net installation image or from the Solaris 10 DVD over the network, follow these steps.
  1. Delete the contents of the disk.
  2. Before you install, create the service partition by using the diagnostics CD for your system.

     For information about how to create the service partition, see your hardware documentation.
  3. Boot the system from the network.

     The Customize `fdisk` Partitions screen is displayed.
  4. To load the default boot disk partition layout, click Default.

     The installation program preserves the service partition and creates the x86 boot partition and the Solaris partition.
- To use the Solaris installation program to install from the Solaris 10 Software - 1 CD or from a network installation image on a boot server, follow these steps.
  1. Delete the contents of the disk.
  2. Before you install, create the service partition by using the diagnostics CD for your system.

     For information about how to create the service partition, see your hardware documentation.
  3. Boot the system.

     The installation program prompts you to choose a method for creating the Solaris partition.
  4. Select the `Use rest of disk for Solaris partition` option.

     The installation program preserves the service partition and creates the Solaris partition.
  5. Complete the installation.

APPENDIX B

Installing or Upgrading Remotely (Tasks)

This appendix describes how to use the Solaris installation program program to install or upgrade to the Solaris OS on a machine or domain that does not have a directly attached DVD-ROM or CD-ROM drive.

Note – If you are installing or upgrading the Solaris OS on a multi–domain server, refer to the system controller or system service processor documentation before beginning the installation process.

SPARC: Using the Solaris Installation Program to Install or Upgrade From a Remote DVD-ROM or CD-ROM

If you want to install the Solaris OS on a machine or domain that does not have a directly attached DVD-ROM or CD-ROM drive, you can use a drive that is attached to another machine. Both machines must be connected to the same subnet. Use the following instructions to complete the installation.

▼ SPARC: To Install or Upgrade From a Remote DVD-ROM and CD-ROM

Note – This procedure assumes that the system is running the Volume Manager. If you are not using the Volume Manager to manage media, refer to *System Administration Guide: Devices and File Systems* for detailed information about managing removable media without the Volume Manager.

In the following procedure, the remote system with the DVD-ROM or CD-ROM is identified as *remote system*. The system that is the client to be installed is identified as *client system*.

Steps

1. **Identify a system that is running the Solaris OS and has a DVD-ROM or CD-ROM drive.**

2. **On the** *remote system* **with the DVD-ROM or CD-ROM drive, insert the Solaris 10 DVD or the Solaris 10 Software for SPARC Platforms - 1 CD in the drive.**

   The Volume Manager mounts the disc.

3. **On the remote system, change directories to the DVD or CD where the `add_install_client` command is located.**

   - For DVD media, type:

     ```
     remote system# cd /cdrom/cdrom0/s0/Solaris_10/Tools
     ```

   - For CD media, type:

     ```
     remote system# cd /cdrom/cdrom0/s0
     ```

4. **On the remote system, add the system that you want to install as a client.**

   - For DVD media, type:

     ```
     remote system# ./add_install_client \
     client_system_name arch
     ```

   - For CD media, type:

     ```
     remote system# ./add_install_client -s remote_system_name: \
     /cdrom/cdrom0/s0 client_system_name arch
     ```

   remote_system_name The name of the system with the DVD-ROM or CD-ROM drive

   client_system_name The name of the machine you want to install

   arch The platform group of the machine you want to install, for example sun4u. On the system that you want to install, find the platform group by using the `uname -m` command.

5. **Boot the *client system* that you want to install.**

   ```
   client system: ok boot net
   ```

   The installation begins.

6. **Follow the instructions to type system configuration information if needed.**
   - If you are using DVD media, follow the instructions on the screen to complete the installation. You are finished.
   - If you are using CD media, the machine reboots and the Solaris installation program begins. After the Welcome panel, the Specify Media panel appears with Network File System selected. Proceed to Step 7.

7. **On the Specify Media panel, click Next.**

   The Specify Network File System Path panel appears and the text field contains the installation path.

   client_system_ip_address:/cdrom/cdrom0/s0

8. **On the remote system where the DVD or CD is mounted, change directories to `root`.**

   ```
   remote system# cd /
   ```

9. **On the remote system, check for the path to the slice that has been shared.**

   ```
   remote system# share
   ```

10. **On the remote system, unshare the Solaris 10 DVD or Solaris 10 Software for SPARC Platforms - 1 CD by using the path that is found in Step 9. If paths lead to two slices, `unshare` both slices.**

    ```
    remote system# unshare absolute_path
    ```

    absolute_path Is the absolute path shown in the `share` command

    In this example, slice 0 and slice 1 are unshared.

    ```
    remote system# unshare /cdrom/cdrom0/s0
    remote system# unshare /cdrom/cdrom0/s1
    ```

11. **On the client system that you are installing, continue the Solaris installation by clicking Next.**

12. **If the Solaris installation program prompts you to insert the Solaris 10 Software - 2 CD, repeat Step 9 through Step 11 to unshare the Solaris 10 Software - 1 CD and to export and install the Solaris 10 Software - 2 CD.**

13. **If the Solaris installation program prompts you to insert additional Solaris 10 Software CDs, repeat Step 9 through Step 11 to unshare the Solaris 10 Software CDs and to export and install the additional CDs.**

14. **If the Solaris installation program prompts you to insert the Solaris 10 Languages CD, repeat Step 9 through Step 11 to unshare the Solaris 10 Software CDs and to export and install the Solaris 10 Languages CD.**

    When you export the Solaris 10 Languages CD, an installer window appears on the machine where the CD-ROM is mounted. Ignore the installer window while you install the Solaris 10 Languages CD. After you complete the installation of the Solaris 10 Languages CD, close the installer window.

APPENDIX C

x86: Preparing to Boot From the Solaris 10 Device Configuration Assistant or the Network (Tasks)

This appendix describes the following topics.

- "x86: Copying the Boot Software to a Diskette" on page 285
- "x86: Booting and Installing Over the Network With PXE" on page 287

x86: Copying the Boot Software to a Diskette

The Solaris Device Configuration Assistant is a program that enables you to perform various hardware configuration and booting tasks. The Solaris 10 Device Configuration Assistant image is found in the Tools directory of either the Solaris 10 Operating System for x86 Platforms DVD or the Solaris 10 Software for x86 Platforms - 2 CD. Use the following procedure to copy the boot image to a 3.5 diskette.

Note – You can boot directly from DVD or CD media or by using a net image with PXE. For information on these methods of booting, see "x86: Booting and Installing Over the Network With PXE" on page 287.

▼ x86: To Copy the Boot Software to a Diskette

Note – This procedure assumes that the system is running Volume Manager. If you are not using Volume Manager to manage diskettes and discs, refer to *System Administration Guide: Devices and File Systems* for detailed information about managing removable media without Volume Manager.

Steps

1. **Log in as superuser on an x86 based system to which a diskette drive is attached.**
2. **On the system with the DVD-ROM or CD-ROM drive, insert the Solaris 10 Operating System for x86 Platforms DVD or the Solaris 10 Software for x86 Platforms - 2 CD in the drive.**

   The Volume Manager mounts the disc.
3. **Change to the directory that contains the boot image.**
   - For DVD media, type:

     ```
     # cd /cdrom/sol_10_x86/s2/Solaris_10/Tools
     ```
   - For CD media, type:

     ```
     # cd /cdrom/sol_10_x86/Solaris_10/Tools
     ```
4. **Insert a blank diskette or a diskette that can be overwritten in the diskette drive.**
5. **Notify Volume Manager to check for new media.**

   ```
   # volcheck
   ```
6. **Format the diskette:**

   Caution – Formatting erases all data on the diskette.

   ```
   # fdformat -d -U
   ```
7. **Copy the file to the diskette.**

   ```
   # dd if=d1_image of=/vol/dev/aliases/floppy0 bs=36k
   ```
8. **Eject the diskette by typing `eject floppy` at the command line, and then manually ejecting the diskette from the drive.**

More Information

Continuing the Installation

If you are installing the Solaris OS from CD or DVD media, see "Performing an Installation or Upgrade With the Solaris Installation Program" in *Solaris 10 Installation Guide: Basic Installations*.

If you are installing the Solaris OS over the network, see "x86: To Create an x86 Install Server" on page 289.

x86: Booting and Installing Over the Network With PXE

This section describes how to set up an x86 based system to install over the network without local boot media. This section describes the following topics.

- "x86: What is PXE?" on page 287
- "x86: Guidelines for Booting With PXE" on page 287
- "x86: Booting With PXE (Task Map)" on page 288
- "x86: To Create an x86 Install Server" on page 289
- "x86: To Add Systems to Install Over the Network By Using PXE" on page 293
- "x86: To Boot the Client Over the Network By Using PXE" on page 297

x86: What is PXE?

PXE network boot is a "direct" network boot. No boot media is required on the client system. With PXE, you can install an x86 based client over the network by using DHCP.

PXE network boot is available only for devices that implement the Intel Preboot Execution Environment specification. To determine if your system supports PXE network boot, see your hardware manufacturer's documentation.

The Solaris boot diskette is still available for systems that do not support PXE. The boot diskette image is available on the Solaris 10 Software for x86 Platforms - 2 CD.

x86: Guidelines for Booting With PXE

To boot over the network by using PXE, you need the following systems.

- An install server
- A DHCP server

- An x86 client that supports PXE

When you are preparing to use PXE to install a client over the network, consider the following issues.

- Set up only one DHCP server on the subnet that includes the client system that you want to install. The PXE network boot does not work properly over subnets that include multiple DHCP servers.
- Some early versions of PXE firmware cannot boot the Solaris system. A system with these older versions can read the PXE network bootstrap program from a boot server, but the bootstrap does not transmit packets. To avoid this problem, upgrade the PXE firmware on the adapter. Obtain firmware upgrade information from the adapter manufacturer's web site. Refer to the `elxl`(7D) and `iprb`(7D) man pages for more information.

x86: Booting With PXE (Task Map)

Perform the following tasks to boot and install your system over the network by using PXE.

TABLE C–1 x86: Task Map: Booting From the Network By Using PXE

Task	Description	Instructions
Verify that your system supports PXE.	Confirm that your machine can use PXE to boot without local boot media.	Check your hardware manufacturer's documentation.
Choose an installation method.	The Solaris OS provides several methods for installation or upgrade. Choose the installation method that is most appropriate for your environment.	"Choosing a Solaris Installation Method" on page 21
Gather information about your system.	Use the checklist and complete the worksheet to collect all of the information that you need to install or upgrade.	Chapter 3
(Optional) Preconfigure system information.	You can preconfigure system information to avoid being prompted for the information during the installation or upgrade.	Chapter 4

TABLE C–1 x86: Task Map: Booting From the Network By Using PXE *(Continued)*

Task	Description	Instructions
Create an install server.	Set up a Solaris server to install the Solaris OS from the network.	"x86: To Create an x86 Install Server" on page 289
Add systems to be installed over the network.	Use `add_install_client -d` to add DHCP support to install a class of client (of a certain machine type, for example) or a particular client ID.	"x86: To Add Systems to Install Over the Network By Using PXE" on page 293
Set up a DHCP server.	Plan for and configure your DHCP service.	Chapter 12, "Planning for DHCP Service (Tasks)," in *System Administration Guide: IP Services*.
Create DHCP options for installation parameters and macros that include the options.	Use DHCP Manager or `dhtadm` to create the vendor options and macros that are output from the `add_install_client -d` command.	"Creating DHCP Options and Macros for Solaris Installation Parameters" on page 80
Boot the client.	Instruct the client BIOS to boot from the network.	"x86: To Boot the Client Over the Network By Using PXE" on page 297

▼ x86: To Create an x86 Install Server

The install server contains the installation image that is needed to install systems from the network. You must create an install server to install the Solaris software on a system from the network.

Before You Begin

This procedure makes the following assumptions.

- You are creating an install server on an x86 based system. For instructions on how to use a SPARC system to serve Solaris installation images for x86 based systems, see "SPARC: To Create a SPARC Install Server With SPARC or x86 DVD Media" on page 99.
- You are using the Solaris Software CDs to create the network installation image. For instructions about how to use the Solaris 10 DVD to create a network installation image, see Chapter 6.
- The system has a CD-ROM drive.
- The system is running the Volume Manager. If you are not using the Volume Manager to manage media, refer to *System Administration Guide: Devices and File Systems* for detailed information about managing removable media without the

Volume Manager.

Note – If you want use the Solaris DVD media to set up an install server on a system that is running the Solaris 7 operating environment, you must first apply one of the following patches.

- Solaris 7 *Intel Platform Edition* operating environment - Patch ID 107260-03

You need the following media.

- Solaris 10 Software for x86 Platforms CDs
- Solaris 10 Languages for x86 Platforms CD

Steps

1. **On the system that is to become the install server, become superuser.**

   The system must include a CD-ROM drive and be part of the site's network and name service. If you use a name service, the system must already be in a name service, such as NIS, NIS+, DNS, or LDAP. If you do not use a name service, you must distribute information about this system by following your site's policies.

2. **Insert the Solaris 10 Software - 1 CD in the system's drive.**

3. **Create a directory for the CD image.**

   ```
   # mkdir -p install_dir_path
   ```

   install_dir_path Specifies the directory where the CD image is to be copied

4. **Change to the `Tools` directory on the mounted disc.**

   ```
   # cd /cdrom/cdrom0/s2/Solaris_10/Tools
   ```

   In the previous example, **cdrom0** is the path to the drive that contains the Solaris OS CD media.

5. **Copy the image in the drive to the install server's hard disk.**

   ```
   # ./setup_install_server install_dir_path
   ```

   install_dir_path Specifies the directory where the CD image is to be copied

   ---

   Note – The `setup_install_server` command indicates whether you have enough disk space available for the Solaris 10 Software disc images. To determine available disk space, use the `df -kl` command.

   ---

6. **Change directories to root (`/`).**

   ```
   # cd /
   ```

7. **Eject the Solaris 10 Software - 1 CD.**

8. **Insert the Solaris 10 Software - 2 CD in the system's CD-ROM drive.**

9. **Change to the `Tools` directory on the mounted CD:**

   ```
   # cd /cdrom/cdrom0/Solaris_10/Tools
   ```

10. **Copy the CD in the CD-ROM drive to the install server's hard disk.**

    ```
    # ./add_to_install_server install_dir_path
    ```

    install_dir_path Specifies the directory where the CD image is to be copied

11. **Change directories to root (/).**

    ```
    # cd /
    ```

12. **Eject the Solaris 10 Software - 2 CD.**

13. **Repeat Step 8 through Step 12 for each Solaris 10 Software CD you want to install.**

14. **Insert the Solaris 10 Languages CD in the system's CD-ROM drive.**

15. **Change to the `Tools` directory on the mounted CD:**

    ```
    # cd /cdrom/cdrom0/Tools
    ```

16. **Copy the CD in the CD-ROM drive to the install server's hard disk.**

    ```
    # ./add_to_install_server install_dir_path
    ```

    install_dir_path Specifies the directory where the CD image is to be copied

17. **Change directories to root (/).**

    ```
    # cd /
    ```

18. **If you want to patch the files that are located in the miniroot (/*install_dir_path*/`Solaris_10/Tools/Boot`) on the net install image, use the `patchadd -C` command to patch these files. You might need to patch a file if a boot image has problems.**

    Caution – Don't use the `patchadd -C` command unless you have read the `Patch README` instructions or have contacted your local Sun support office.

Example C–1 x86: Creating an x86 Install Server With x86 CD Media

The following example illustrates how to create an install server by copying the following CDs to the install server's `/export/home/cdx86` directory.

- Solaris 10 Software for x86 Platforms CDs
- Solaris 10 Languages for x86 Platforms CD

Insert the Solaris 10 Software for x86 Platforms - 1 CD in the system's CD-ROM drive.

```
# mkdir -p /export/home/cdx86
# cd /cdrom/cdrom0/s2/Solaris_10/Tools
# ./setup_install_server /export/home/cdx86
# cd /
```

Eject the Solaris 10 Software for x86 Platforms - 1 CD. Insert the Solaris 10 Software for x86 Platforms - 2 CD in the system's CD-ROM drive.

```
# cd /cdrom/cdrom0/Solaris_10/Tools
# ./add_to_install_server /export/home/cdx86
# cd /
```

Repeat the previous commands for each Solaris 10 Software CD that you want to install.

Insert the Solaris 10 Languages for x86 Platforms CD in the system's CD-ROM drive.

```
# cd /cdrom/cdrom0/Tools
# ./add_to_install_server /export/home/cdx86
# cd /
#
```

More Information Continuing the Installation

After you set up the install server, you must add the client as an installation client. For information about how to add client systems to install over the network with PXE, see "x86: To Add Systems to Install Over the Network By Using PXE" on page 293.

If you are not using PXE, and your client system is on a different subnet than your install server, you must create a boot server. For more information, see "Creating a Boot Server on a Subnet With a CD Image" on page 141.

See Also For additional information about the `setup_install_server` and the `add_to_install_server` commands, see `install_scripts`(1M).

▼ x86: To Add Systems to Install Over the Network By Using PXE

After you create an install server, you must set up each system that you want to install from the network.

Use the following `add_install_client` procedure for set up an x86 client to install from the network by using PXE.

Before You Begin

Each system that you want to install needs to find the following:

- An install server. For instructions about how to create an install server from CD media, see "x86: To Create an x86 Install Server" on page 289.
- A DHCP server. For instructions about how to set up a DHCP server to support network installations, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.
- The `sysidcfg` file if you use a `sysidcfg` file to preconfigure system information. For information about how to create a `sysidcfg` file, see "Preconfiguring With the `sysidcfg` File" on page 57.
- A name server if you use a name service to preconfigure system information. For information about how to preconfigure information with a name service, see "Preconfiguring With the Name Service" on page 75.
- The profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method. For information about how to set up a custom JumpStart installation, see Chapter 4, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

Steps

1. **On the install server, become superuser.**

2. **If you use the NIS, NIS+, DNS, or LDAP name service, verify that the following information about the system to be installed has been added to the name service:**

   - Host name
   - IP address
   - Ethernet address

   For more information on name services, see *System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)*.

3. **Change to the `Tools` directory on the Solaris 10 CD image on the install server:**

```
# cd /install_dir_path/Solaris_10/Tools
```

install_dir_path Specifies the path to the `Tools` directory

4. **Set up the client system to be installed from the network.**

```
# ./add_install_client -d -s install_server:install_dir_path \
-c jumpstart_server:jumpstart_dir_path \
-p sysid_server:path \
-t boot_image_path -b "boot-property=value" \
-e ethernet_address client_name platform_group
```

`-d`

Specifies that the client is to use DHCP to obtain the network install parameters. If you use the `-d` only, the `add_install_client` command sets up the installation information for client systems of the same class, for example, all x86 client machines. To set up the installation information for a specific client, use the `-d` with the `-e` option.

For more information about class-specific installations by using DHCP, see "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 80.

`-s` *install_server:install_dir_path*

Specifies the name and path to the install server.

- *install_server* is the host name of the install server
- *install_dir_path* is the absolute path to the Solaris 10 CD image

`-c` *jumpstart_server*`:`*jumpstart_dir_path*

Specifies a JumpStart directory for custom JumpStart installations. *jumpstart_server* is the host name of the server on which the JumpStart directory is located. *jumpstart_dir_path* is the absolute path to the JumpStart directory.

`-p` *sysid_server*`:`*path*

Specifies the path to the `sysidcfg` file for preconfiguring system information. *sysid_server* is either a valid host name or an IP address for the server that contains the file. *path* is the absolute path to the directory containing the `sysidcfg` file.

`-t` *boot_image_path*

Specifies the path to an alternate boot image if you want to use a boot image other than the image in the Tools directory on the Solaris 10 net installation image, CD, or DVD.

`-b` "*boot-property=value*"

x86 based systems only: Enables you to set a boot property variable that you want to use to boot the client from the network. The `-b` must be used with the `-e` option.

See the `eeprom`(1M) man page for descriptions of boot properties.

`-e` *ethernet_address*

Specifies the Ethernet address of the client that you want to install. This option enables you to set up the installation information to use for a specific client.

For more information about client-specific installations by using DHCP, see "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 80.

client_name
Is the name of the system to be installed from the network. This name is *not* the host name of the install server.

platform_group
Is the platform group of the system to be installed. A detailed list of platform groups appears in "Platform Names and Groups" on page 35.

The previous command outputs the vendor options and macros that you need to add to you DHCP server. See "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 80 for instructions about how to define these vendor options and macros in your DHCP server.

Example C–2 x86: Adding an x86 Install Client on an x86 Install Server When Using DHCP and PXE (CDs)

The following example illustrates how to add an x86 install client to an install server when you are using DHCP to set installation parameters on the network. The -d option is used to specify that clients are to use the DHCP protocol for configuration. The DHCP class name SUNW.i86pc indicates that this command applies to all Solaris x86 network boot clients, not just a single client. The -s option is used to specify that the clients are to be installed from the install server that is named rosemary. This server contains a Solaris 10 Software for x86 Platforms - 1 CD image in /export/home/cdx86.

For more information on how to use DHCP to set installation parameters for network installations, see Preconfiguring System Configuration Information With the DHCP Service (Tasks).

```
x86_install_server# cd /export/boot/Solaris_10/Tools
x86_install_server# ./add_install_client -d -s rosemary:/export/home/cdx86 \
SUNW.i86pc i86pc
```

Example C–3 x86: Specifying a Serial Console to Use During a Network Installation (CDs)

The following example illustrates how to add an x86 install client to an install server and specify a serial console to use during the installation. This example sets up the install client in the following manner.

- The -d option indicates that the client is set up to use DHCP to set installation parameters.
- The -e option indicates that this installation occurs only on the client with the Ethernet address 00:07:e9:04:4a:bf.
- The -b option instructs the installation program to use the serial port ttya as an input and an output device.

```
install server# cd /export/boot/Solaris_10/Tools
install server# ./add_install_client -d -e "00:07:e9:04:4a:bf" \
-b "input-device=ttya" -b "output-device=ttya" i86pc
```

For a complete description of the boot property variables that you can use with the `-b` option, see the `eeprom`(1M) man page.

Example C–4 x86: Specifying a Boot Device to Use During a Network Installation (CDs)

The following example illustrates how to add an x86 install client to an install server and specify a boot device to use during the installation. If you specify the boot device when you set up the install client, you are not prompted for this information by the Device Configuration Assistant during the installation.

This example sets up the install client in the following manner.

- The `-d` option indicates that the client is set up to use DHCP to set installation parameters.
- The `-e` option indicates that this installation occurs only on the client with the Ethernet address 00:07:e9:04:4a:bf.
- The first two uses of the `-b` option instruct the installation program to use the serial port `ttya` as an input and an output device.
- The third use of the `-b` option instructs the installation program to use a specific boot device during the installation.

  Note – The value of the boot device path varies based on your hardware.

- The i86pc platform name indicates that the client is an x86 based system.

```
install server# cd /export/boot/Solaris_10/Tools
install server# ./add_install_client -d -e "00:07:e9:04:4a:bf" \
-b "input-device=ttya" -b "output-device=ttya" \
-b "bootpath=/pci@0,0/pci108e,16a8@8" i86pc
```

For a complete description of the boot property variables that you can use with the `-b` option, see the `eeprom`(1M) man page.

More Information

Continuing the Installation

After you add your system as an installation client, you are ready to install your system from the network. For information about how to boot and install the system from the network, see "x86: To Boot the Client Over the Network By Using PXE" on page 297.

See Also

For additional information about the `add_install_client` command, see `install_scripts`(1M).

▼ x86: To Boot the Client Over the Network By Using PXE

To install the system over the network, you must instruct the client system to boot over the network. Enable PXE network boot on the client system by using the BIOS setup program in the system BIOS, the network adapter BIOS, or both. On some systems, you must also adjust the boot device priority list so that network boot is attempted before booting from other devices. See the manufacturer's documentation for each setup program, or watch for setup program instructions during boot.

Before You Begin

This procedure assumes that you have completed the following tasks.

- Set up an install server. For instructions about how to create an install server from CD media, see "x86: To Create an x86 Install Server" on page 289.
- Set up a DHCP server. For instructions about how to set up a DHCP server to support network installations, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 78.
- Gathered or preconfigured the information you need to install. You can perform this task in one or more of the following ways.
  - Gather the information in "Checklist for Installation" on page 41.
  - Create a `sysidcfg` file if you use a `sysidcfg` file to preconfigure system information. For information about how to create a `sysidcfg` file, see "Preconfiguring With the `sysidcfg` File" on page 57.
  - Set up a name server if you use a name service to preconfigure system information. For information about how to preconfigure information with a name service, see "Preconfiguring With the Name Service" on page 75.
  - Create a profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method. For information about how to set up a custom JumpStart installation, see Chapter 4, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 Installation Guide: Custom JumpStart and Advanced Installations*.

This procedure also assumes that your system can boot from the network. If your system cannot boot from the network, you must create a boot diskette to install over the network. See "x86: Copying the Boot Software to a Diskette" on page 285 for information about how to create a boot diskette.

Steps

1. **Turn on the system.**
2. **Type the appropriate keystroke combination to enter the system BIOS.**

   Some PXE-capable network adapters have a feature that enables PXE boot if you type a particular keystroke in response to a brief boot-time prompt.
3. **In the system BIOS, instruct the system to boot from the network.**

See your hardware documentation for information about how to set the boot priority in the BIOS.

4. **Exit the BIOS.**

   The system boots from the network.

5. **When prompted, select an installation type.**

   - **To install with the Solaris interactive installation GUI, type 1 and Enter.**
   - **To perform a custom JumpStart installation, type 2 and Enter.**
   - **To install with the Solaris interactive text installer in a desktop session, type 3 and Enter.**
   - **To install with the Solaris interactive text installer in a console session, type 4 and Enter.**

   The installation program begins.

6. **If you are prompted, answer the system configuration questions.**

   - If you preconfigured all of the system information, the installation program does not prompt you to enter any configuration information. See Chapter 4 for more information.
   - If you did not preconfigure all the system information, use the "Checklist for Installation" on page 41 to help you answer the configuration questions.

   If you are using the installation GUI, after you confirm the system configuration information, the Welcome to Solaris dialog box appears.

7. **After the system boots and installs over the network, instruct the system to boot from the disk drive on subsequent boots.**

See Also For information about how to complete an interactive installation with the Solaris installation GUI, see "To Install or Upgrade With the Solaris Installation Program" in *Solaris 10 Installation Guide: Basic Installations*.

Glossary

3DES ([Triple DES] Triple-Data Encryption Standard). A symmetric-key encryption method that provides a key length of 168 bits.

AES (Advanced Encryption Standard) A symmetric 128-bit block data encryption technique. The U.S. government adopted the Rijndael variant of the algorithm as its encryption standard in October 2000. AES replaces DES encryption as the government standard.

archive A file that contains a collection of files that were copied from a master system. The file also contains identification information about the archive, such as a name and the date that you created the archive. After you install an archive on a system, the system contains the exact configuration of the master system.

An archive could be a differential archive which is a Solaris Flash archive that contains only the differences between two system images, an unchanged master image and an updated master image. The differential archive contains files to be retained, modified, or deleted from the clone system. A differential update changes only the files specified and is restricted to systems that contain software consistent with the unchanged master image.

arrow keys One of the four directional keys on the numeric keypad.

begin script A user-defined Bourne shell script, specified within the `rules` file, that performs tasks before the Solaris software is installed on the system. You can use begin scripts only with custom JumpStart installations.

boot To load the system software into memory and start it.

boot environment A collection of mandatory file systems (disk slices and mount points) that are critical to the operation of the Solaris OS. These disk slices might be on the same disk or distributed across multiple disks.

The active boot environment is the one that is currently booted. Exactly one active boot environment can be booted. An inactive boot environment is not currently booted, but can be in a state of waiting for activation on the next reboot.

`bootlog-cgi`

The CGI program that enables a web server to collect and store remote client-booting and installation console messages during a WAN boot installation.

boot server

A server system that provides client systems on the same network subnet with the programs and information that they need to start. A boot server is required to install over the network if the install server is on a different subnet than the systems on which Solaris software is to be installed.

certificate authority

(CA) A trusted third-party organization or company that issues digital certificates that are used to create digital signatures and public-private key pairs. The CA guarantees that the individual who is granted the unique certificate is who she or he claims to be.

`certstore`

A file that contains a digital certificate for a specific client system. During an SSL negotiation, the client might be asked to provide the certificate file to the server. The server uses this file to verify the identity of the client.

CGI

(Common Gateway Interface) An interface by which external programs communicate with the HTTP server. Programs that are written to use CGI are called CGI programs or CGI scripts. CGI programs handle forms or parse output the server does not normally handle or parse.

checksum

The result of adding a group of data items that are used for checking the group. The data items can be either numerals or other character strings that are treated as numerals during the checksum calculation. The checksum value verifies that communication between two devices is successful.

client

In the client-server model for communications, the client is a process that remotely accesses resources of a compute server, such as compute power and large memory capacity.

clone system

A system that you installed by using a Solaris Flash archive. The clone system has the same installation configuration as the master system.

cluster

A logical collection of packages (software modules). The Solaris software is divided into *software groups*, which are each composed of clusters and *packages*.

command line

A string of characters that begins with a command, often followed by arguments, including options, file names, and other expressions, and terminated by the end-of-line character.

concatenation A RAID-0 volume. If slices are concatenated, the data is written to the first available slice until that slice is full. When that slice is full, the data is written to the next slice, serially. A concatenation provides no data redundancy unless it is contained in a mirror. See also RAID-0 volume.

Core Software Group A software group that contains the minimum software that is required to boot and run the Solaris OS on a system. Core includes some networking software and the drivers that are required to run the Common Desktop Environment (CDE) desktop. Core does not include the CDE software.

critical file systems File systems that are required by the Solaris OS. When you use Solaris Live Upgrade, these file systems are separate mount points in the `vfstab` of the active and inactive boot environments. Examples are `root` (/), `/usr`, `/var`, and `/opt`. These file systems are always copied from the source to the inactive boot environment.

custom JumpStart A type of installation in which the Solaris software is automatically installed on a system that is based on a user-defined profile. You can create customized profiles for different types of users and systems. A custom JumpStart installation is a JumpStart installation you create.

custom probes file A file, which must be located in the same JumpStart directory as the `rules` file, that is a Bourne shell script that contains two types of functions: probe and comparison. Probe functions gather the information you want or do the actual work and set a corresponding `SI_` environment variable you define. Probe functions become probe keywords. Comparison functions call a corresponding probe function, compare the output of the probe function, and return 0 if the keyword matches or 1 if the keyword doesn't match. Comparison functions become rule keywords. See also *rules file*.

decryption The process of converting coded data to plain text. See also encryption.

derived profile A profile that is dynamically created by a begin script during a custom JumpStart installation.

DES (Data Encryption Standard) A symmetric-key encryption method that was developed in 1975 and standardized by ANSI in 1981 as ANSI X.3.92. DES uses a 56-bit key.

Developer Solaris Software Group A software group that contains the End User Solaris Software Group plus the libraries, include files, man pages, and programming tools for developing software.

DHCP (Dynamic Host Configuration Protocol) An application-layer protocol. Enables individual computers, or clients, on a TCP/IP network to extract an IP address and other network configuration information from a designated and centrally maintained DHCP server or servers. This facility reduces the overhead of maintaining and administering a large IP network.

differential archive A Solaris Flash archive that contains only the differences between two system images, an unchanged master image and an updated master image. The differential archive contains files to be retained, modified, or deleted from the clone system. A differential update changes only the files that are specified and is restricted to systems that contain software consistent with the unchanged master image.

digital certificate A nontransferable, nonforgeable, digital file issued from a third party that both communicating parties already trust.

disc An optical disc, as opposed to a magnetic disk, which recognizes the common spelling that is used in the compact disc (CD) market. For example, a CD-ROM or DVD-ROM is an optical disc.

disk A round platter, or set of platters, of a magnetized medium that is organized into concentric tracks and sectors for storing data such as files. See also disc.

disk configuration file A file that represents a structure of a disk (for example, bytes/sector, flags, slices). Disk configuration files enable you to use `pfinstall` from a single system to test profiles on different–size disks.

diskless client A client on a network that relies on a server for all of its disk storage.

document root directory The root of a hierarchy on a web server machine that contains the files, images, and data you want to present to users who are accessing the web server.

domain A part of the Internet naming hierarchy. A domain represents a group of systems on a local network that share administrative files.

domain name The name that is assigned to a group of systems on a local network that share administrative files. The domain name is required for the Network Information Service (NIS) database to work properly. A domain name consists of a sequence of component names that are separated by periods (for example: `tundra.mpk.ca.us`). As you read a domain name from left to right, the component names identify more general (and usually remote) areas of administrative authority.

encryption The process of protecting information from unauthorized use by making the information unintelligible. Encryption is based on a code, called a key, which is used to decrypt the information. See also decryption.

End User Solaris Software Group
A software group that contains the Core Software Group plus the recommended software for an end user, including the Common Desktop Environment (CDE) and DeskSet software.

Entire Solaris Software Group
A software group that contains the entire Solaris 10 release.

Entire Solaris Software Group Plus OEM Support
A software group that contains the entire Solaris 10 release, plus additional hardware support for OEMs. This software group is recommended when installing Solaris software on SPARC based servers.

`/etc`
A directory that contains critical system configuration files and maintenance commands.

`/etc/netboot` directory
The directory on a WAN boot server that contains the client configuration information and security data that are required for a WAN boot installation.

`/export`
A file system on an OS server that is shared with other systems on a network. For example, the `/export` file system can contain the root file system and swap space for diskless clients and the home directories for users on the network. Diskless clients rely on the `/export` file system on an OS server to boot and run.

fallback
A reversion to the environment that ran previously. Use fallback when you are activating an environment and the boot environment that is designated for booting fails or shows some undesirable behavior.

`fdisk` partition
A logical partition of a disk drive that is dedicated to a particular operating system on x86 based systems. To install the Solaris software, you must set up at least one Solaris `fdisk` partition on an x86 based system. x86 based systems allow up to four different `fdisk` partitions on a disk. These partitions can be used to hold individual operating systems. Each operating system must be located on a unique `fdisk` partition. A system can only have one Solaris `fdisk` partition per disk.

file server
A server that provides the software and file storage for systems on a network.

file system
In the SunOS™ operating system, a tree-structured network of files and directories that you can access.

finish script
A user-defined Bourne shell script, specified within the `rules` file, that performs tasks after the Solaris software is installed on the system, but before the system reboots. You use finish scripts with custom JumpStart installations.

format
To put data into a structure or divide a disk into sectors for receiving data.

function key
One of the 10 or more keyboard keys that are labeled F1, F2, F3, and so on that are mapped to particular tasks.

global zone In Solaris Zones, the global zone is both the default zone for the system and the zone used for system-wide administrative control. The global zone is the only zone from which a non-global zone can be configured, installed, managed, or uninstalled. Administration of the system infrastructure, such as physical devices, routing, or dynamic reconfiguration (DR), is only possible in the global zone. Appropriately privileged processes running in the global zone can access objects associated with other zones. See also Solaris Zones and non-global zone.

hard link A directory entry that references a file on disk. More than one such directory entry can reference the same physical file.

hash A number that is produced by taking some input and generating a number that is significantly shorter than the input. The same output value is always generated for identical inputs. Hash functions can be used in table search algorithms, in error detection, and in tamper detection. When used for tamper detection, hash functions are chosen such that it is difficult to find two inputs that yield the same hash result. MD5 and SHA-1 are examples of one-way hash functions. For example, a message digest takes a variable-length input such as a disk file and reduces it to a small value.

hashing The process of changing a string of characters into a value or key that represents the original string.

HMAC Keyed hashing method for message authentication. HMAC is used with an iterative cryptographic hash function, such as MD5 or SHA-1, in combination with a secret shared key. The cryptographic strength of HMAC depends on the properties of the underlying hash function.

host name The name by which a system is known to other systems on a network. This name must be unique among all the systems within a particular domain (usually, this means within any single organization). A host name can be any combination of letters, numbers, and minus signs (-), but it cannot begin or end with a minus sign.

HTTP (Hypertext Transfer Protocol) (n.) The Internet protocol that fetches hypertext objects from remote hosts. This protocol is based on TCP/IP.

HTTPS A secure version of HTTP, implemented by using the Secure Sockets Layer (SSL).

initial installation An installation that overwrites the currently running software or initializes a blank disk.

An initial installation of the Solaris OS overwrites the system's disk or disks with the new version of the Solaris OS. If your system is not running the Solaris OS, you must perform an initial installation. If your system is running an upgradable version of the Solaris OS, an initial installation overwrites the disk and does not preserve the OS or local modifications.

install server A server that provides the Solaris DVD or CD images from which other systems on a network can install Solaris (also known as a *media server*). You can create an install server by copying the Solaris DVD or CD images to the server's hard disk.

IP address (Internet protocol address) In TCP/IP, a unique 32-bit number that identifies each host in a network. An IP address consists of four numbers that are separated by periods (192.168.0.0, for example). Most often, each part of the IP address is a number between 0 and 225. However, the first number must be less than 224 and the last number cannot be 0.

IP addresses are logically divided into two parts: the network (similar to a telephone area code), and the local system on the network (similar to a phone number). The numbers in a Class A IP address, for example, represent "`network.local.local.local`" and the numbers in a Class C IP address represent "`network.network.network.local`."

Class	Range (*xxx* is a number 0 to 255)	Number of Available IP Addresses
Class A	1.*xxx.xxx.xxx* - 126.*xxx.xxx.xxx*	Over 16 million
Class B	128.0.*xxx.xxx* - 191.255.*xxx.xxx*	Over 65,000
Class C	192.0.0.*xxx* - 223.255.255.*xxx*	256

IPv6 IPv6 is a version (version 6) of Internet Protocol (IP) that is designed to be an evolutionary step from the current version, IPv4 (version 4). Deploying IPv6, by using defined transition mechanisms, does not disrupt current operations. In addition, IPv6 provides a platform for new Internet functionality.

IPv6 is described in more detail in Part I, "Introducing System Administration: IP Services," in *System Administration Guide: IP Services*.

job A user-defined task to be completed by a computer system.

JumpStart directory When you use a profile diskette for custom JumpStart installations, the JumpStart directory is the root directory on the diskette that contains all the essential custom JumpStart files. When you use a profile server for custom JumpStart installations, the JumpStart directory is a directory on the server that contains all the essential custom JumpStart files.

JumpStart installation A type of installation in which the Solaris software is automatically installed on a system by using the factory-installed JumpStart software.

Kerberos A network authentication protocol that uses strong, secret-key cryptography to enable a client and server to identify themselves to each other over an insecure network connection.

key The code for encrypting or decrypting data. See also encryption.

`keystore` A file that contains keys shared by a client and server. During a WAN boot installation, the client system uses the keys to verify the integrity of, or decrypt the data and files transmitted from, the server.

LAN (local area network) A group of computer systems in close proximity that can communicate by way of some connecting hardware and software.

LDAP (Lightweight Directory Access Protocol) A standard, extensible directory access protocol that is used by LDAP naming service clients and servers to communicate with each other.

locale A geographic or political region or community that shares the same language, customs, or cultural conventions (English for the U.S. is en_US, and English for the U.K. is en_UK).

logical device A group of physical slices on one or more disks that appear to the system as a single device. A logical device is called a volume in Solaris Volume Manager. A volume is functionally identical to a physical disk in the view of an application or file system.

manifest section A section of a Solaris Flash archive that is used to validate a clone system. The manifest section lists the files on a system to be retained, added to, or deleted from the clone system. This section is informational only. The section lists the files in an internal format and cannot be used for scripting.

master system A system that you use to create a Solaris Flash archive. The system configuration is saved in the archive.

MD5 (Message Digest 5) An iterative cryptographic hash function that is used for message authentication, including digital signatures. The function was developed in 1991 by Rivest.

media server See *install server*.

metadevice See *volume*.

miniroot The smallest possible bootable Solaris `root` file system. A miniroot contains a kernel and just enough software to install the Solaris environment on a hard disk. The miniroot is the file system that is copied to a machine in the initial installation.

mirror See RAID-1 volume.

mount The process of accessing a directory from a disk that is attached to a machine that is making the mount request or a remote disk on a network. To mount a file system, you need a mount point on the local system and the name of the file system to be mounted (for example, `/usr`).

mount point A workstation directory to which you mount a file system that exists on a remote machine.

name server A server that provides a name service to systems on a network.

name service A distributed network database that contains key system information about all the systems on a network so that the systems can communicate with each other. With a name service, the system information can be maintained, managed, and accessed on a network-wide basis. Without a name service, each system has to maintain its own copy of the system information in the local `/etc` files. Sun supports the following name services: LDAP, NIS, and NIS+.

networked systems A group of systems (called hosts) that are connected through hardware and software so that they can communicate and share information. Referred to as a local area network (LAN). One or more servers are usually needed when systems are networked.

network installation A way to install software over the network—from a system with a CD-ROM or DVD-ROM drive to a system without a CD-ROM or DVD-ROM drive. Network installations require a *name server* and an *install server*.

NIS The SunOS 4.0 (minimum) Network Information Service. A distributed network database that contains key information about the systems and the users on the network. The NIS database is stored on the master server and all the slave servers.

NIS+ The SunOS 5.0 (minimum) Network Information Service. NIS+ replaces NIS, the SunOS 4.0 (minimum) Network Information Service.

non-global zone A virtualized operating system environment created within a single instance of the Solaris Operating System. One or more applications can run in a non-global zone without interacting with the rest of the system. Non-global zones are also called zones. See also Solaris Zones and global zone.

nonnetworked systems Systems that are not connected to a network or do not rely on other systems.

`/opt` A file system that contains the mount points for third-party and unbundled software.

OS server A system that provides services to systems on a network. To serve diskless clients, an OS server must have disk space set aside for each diskless client's root file system and swap space (`/export/root`, `/export/swap`).

package A collection of software that is grouped into a single entity for modular installation. The Solaris software is divided into *software groups*, which are each composed of *clusters* and packages.

panel A container for organizing the contents of a window, a dialog box, or applet. The panel might collect and confirm user input. Panels might be used by wizards and follow an ordered sequence to fulfill a designated task.

patch analyzer A script that you can run manually or as part of the Solaris installation program. The patch analyzer performs an analysis on your system to determine which (if any) patches will be removed by upgrading to a Solaris update.

platform group A vendor-defined grouping of hardware platforms for the purpose of distributing specific software. Examples of valid platform groups are i86pc and sun4u.

platform name The output of the `uname -i` command. For example, the platform name for the Ultra 60 is SUNW,Ultra-60.

Power Management Software that automatically saves the state of a system and turns it off after it is idle for 30 minutes. When you install the Solaris software on a system that complies with Version 2 of the U.S. Environmental Protection Agency's Energy Star guidelines—a sun4u SPARC system, for example—the Power Management software is installed by default. After a subsequent reboot, you are prompted to enable or disable the Power Management software.

Energy Star guidelines require that systems or monitors automatically enter a "sleep state" (consume 30 watts or less) after the system or monitor becomes inactive.

probe keyword A syntactical element that extracts attribute information about a system when using the custom JumpStart method to install. A probe keyword does not require you to set up a matching condition and run a profile as required for a rule. See also *rule*.

profile A text file that defines how to install the Solaris software when using the custom JumpStart method. For example, a profile defines which software group to install. Every rule specifies a profile that defines how a system is to be installed when the rule is matched. You usually create a different profile for every rule. However, the same profile can be used in more than one rule. See also *rules file*.

profile diskette A diskette that contains all the essential custom JumpStart files in its root directory (JumpStart directory).

profile server A server that contains all the essential custom JumpStart files in a JumpStart directory.

private key The decryption key used in public-key encryption.

public key The encryption key used in public-key encryption.

public-key cryptography A cryptographic system that uses two keys: a public key known to everyone, and a private key known only to the recipient of the message.

RAID-1 volume A class of volume that replicates data by maintaining multiple copies. A RAID-1 volume is composed of one or more RAID-0 volumes called submirrors. A RAID-1 volume is sometimes called a mirror.

RAID-0 volume A class of volume that can be a stripe or a concatenation. These components are also called submirrors. A stripe or concatenation is the basic building block for mirrors.

Reduced Network Support Software Group A software group that contains the minimum code that is required to boot and run a Solaris system with limited network service support. The Reduced Networking Software Group provides a multiuser text-based console and system administration utilities. This software group also enables the system to recognize network interfaces, but does not activate network services.

/ (root) In a hierarchy of items, the one item from which all other items are descended. The root item has nothing above it in the hierarchy. / is the base directory from which all other directories stem, directly or indirectly. The root directory contains the directories and files critical for system operation, such as the kernel, device drivers, and the programs that are used to start (boot) a system.

rule A series of values that assigns one or more system attributes to a profile. A rule is used in a custom JumpStart installation.

`rules` file A text file that contains a rule for each group of systems or single systems that you want to install automatically. Each rule distinguishes a group of systems, based on one or more system attributes. The `rules` file links each group to a profile, which is a text file that defines how the Solaris software is to be installed on each system in the group. A rules file is used in a custom JumpStart installation. See also *profile*.

`rules.ok` file A generated version of the `rules` file. The `rules.ok` file is required by the custom JumpStart installation software to match a system to a profile. You *must* use the `check` script to create the `rules.ok` file.

Secure Sockets Layer (SSL) A software library establishing a secure connection between two parties (client and server) used to implement HTTPS, the secure version of HTTP.

server A network device that manages resources and supplies services to a client.

SHA1 (Secure Hashing Algorithm) The algorithm that operates on any input length less than 2^{64} to produce a message digest.

shareable file systems File systems that are user-defined files such as `/export/home` and `/swap`. These file systems are shared between the active and inactive boot environment when you use Solaris Live Upgrade. Shareable file systems contain the same mount point in the `vfstab` in both the active and inactive boot environments. Updating shared files in the active boot environment also updates data in the inactive boot environment. Shareable file systems are shared by default, but you can specify a destination slice, and then the file systems are copied.

slice The unit into which the disk space is divided by the software.

software group A logical grouping of the Solaris software (clusters and packages). During a Solaris installation, you can install one of the following software groups: Core, End User Solaris Software, Developer Solaris Software, or Entire Solaris Software, and for SPARC systems only, Entire Solaris Software Group Plus OEM Support.

Solaris DVD or CD images The Solaris software that is installed on a system, which you can access on the Solaris DVDs or CDs or an install server's hard disk to which you have copied the Solaris DVD or CD images.

Solaris Flash A Solaris installation feature that enables you to create an archive of the files on a system, known as the master system. You can then use the archive to install other systems, making the other systems identical in their configuration to the master system. See also *archive.*

Solaris installation program A graphical user interface (GUI) or command–line interface (CLI) installation program that uses wizard panels to guide you step-by-step through installing the Solaris software and third-party software.

Solaris Live Upgrade An upgrade method that enables a duplicate boot environment to be upgraded while the active boot environment is still running, thus eliminating downtime of the production environment.

Solaris Zones A software partitioning technology used to virtualize operating system services and provide an isolated and secure environment for running applications. When you create a non-global zone, you produce an application execution environment in which processes are isolated from the all other zones. This isolation prevents processes that are running in a zone from monitoring or affecting processes that are running in any other zones. See also global zone and non-global zone.

standalone A computer that does not require support from any other machine.

state database A database that stores information about disk about the state of your Solaris Volume Manager configuration. The state database is a collection of multiple, replicated database copies. Each copy is referred to as a state database replica. The state database tracks the location and status of all known state database replicas.

state database replica A copy of a state database. The replica ensures that the data in the database is valid.

submirror See RAID-0 volume.

subnet A working scheme that divides a single logical network into smaller physical networks to simplify routing.

subnet mask A bit mask that is used to select bits from an Internet address for subnet addressing. The mask is 32 bits long and selects the network portion of the Internet address and 1 or more bits of the local portion.

swap space A slice or file that temporarily holds the contents of a memory area till it can be loaded back into memory. Also called the `/swap` or `swap` file system.

`sysidcfg` file A file in which you specify a set of special system configuration keywords that preconfigure a system.

system configuration file (`system.conf`) A text file in which you specify the locations of the `sysidcfg` file and the custom JumpStart files you want to use in a WAN boot installation.

time zone Any of the 24 longitudinal divisions of the earth's surface for which a standard time is kept.

`truststore` A file that contains one or more digital certificates. During a WAN boot installation, the client system verifies the identity of the server that is trying to perform the installation by consulting the data in the `truststore` file.

unmount The process of removing access to a directory on a disk that is attached to a machine or to a remote disk on a network.

update An installation, or to perform an installation, on a system that changes software that is of the same type. Unlike an upgrade, an update might downgrade the system. Unlike an initial installation, software of the same type that is being installed must be present before an update can occur.

upgrade An installation that merges files with existing files and saves modifications where possible.

An upgrade of the Solaris OS merges the new version of the Solaris OS with the existing files on the system's disk or disks. An upgrade saves as many modifications as possible that you have made to the previous version of the Solaris OS.

upgrade option An option that is presented by the Solaris installation program program. The upgrade procedure merges the new version of Solaris with existing files on your disk or disks. An upgrade also saves as many local modifications as possible since the last time Solaris was installed.

URL (Uniform Resource Locator) The addressing system used by the server and the client to request documents. A URL is often called a location. The format of a URL is *protocol://machine:port/document*.

A sample URL is `http://www.example.com/index.html`.

`/usr` A file system on a standalone system or server that contains many of the standard UNIX programs. Sharing the large `/usr` file system with a server rather than maintaining a local copy minimizes the overall disk space that is required to install and run the Solaris software on a system.

utility A standard program, usually furnished at no charge with the purchase of a computer, that does the computer's housekeeping.

`/var` A file system or directory (on standalone systems) that contains system files that are likely to change or grow over the life of the system. These files include system logs, `vi` files, mail files, and uucp files.

volume A group of physical slices or other volumes that appear to the system as a single logical device. A volume is functionally identical to a physical disk in the view of an application or file system.

In some command-line utilities, a volume is called a metadevice. Volume is also called pseudo device or virtual device in standard UNIX terms.

Volume Manager A program that provides a mechanism to administer and obtain access to the data on DVD-ROMs, CD-ROMs, and diskettes.

WAN (wide area network) A network that connects multiple local area networks (LANs) or systems at different geographical sites by using telephone, fiber-optic, or satellite links.

WAN boot installation A type of installation that enables you to boot and install software over a wide area network (WAN) by using HTTP or HTTPS. The WAN boot installation method enables you to transmit an encrypted Solaris Flash archive over a public network and perform a custom JumpStart installation on a remote client.

WAN boot miniroot A miniroot that has been modified to perform a WAN boot installation. The WAN boot miniroot contains a subset of the software in the Solaris miniroot. See also miniroot.

WAN boot server A web server that provides the configuration and security files that are used during a WAN boot installation.

`wanboot` program The second-level boot program that loads the WAN boot miniroot, client configuration files, and installation files that are required to perform a WAN boot installation. For WAN boot installations, the `wanboot` binary performs tasks similar to the `ufsboot` or `inetboot` second-level boot programs.

`wanboot-cgi` program The CGI program that retrieves and transmits the data and files that are used in a WAN boot installation.

`wanboot.conf` file A text file in which you specify the configuration information and security settings that are required to perform a WAN boot installation.

zone See non-global zone

Index