Solaris 10 10/08 Installation Guide: Network-Based Installations

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

Part No: 820–5237–10
September 2008

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more U.S. patents or pending patent applications in the U.S. and in other countries.

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

This distribution may include materials developed by third parties.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, the Solaris logo, the Java Coffee Cup logo, docs.sun.com, Sun4U, Power Management, SunOS, Ultra, JumpStart, Java, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. or its subsidiaries in the U.S. and other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

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

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

Copyright 2008 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. Tous droits réservés.

Sun Microsystems, Inc. détient les droits de propriété intellectuelle relatifs à la technologie incorporée dans le produit qui est décrit dans ce document. En particulier, et ce sans limitation, ces droits de propriété intellectuelle peuvent inclure un ou plusieurs brevets américains ou des applications de brevet en attente aux Etats-Unis et dans d'autres pays.

Cette distribution peut comprendre des composants développés par des tierces personnes.

Certaines composants de ce produit peuvent être dérivées du logiciel Berkeley BSD, licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays; elle est licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, le logo Solaris, le logo Java Coffee Cup, docs.sun.com, Sun4U, Power Management, SunOS, Ultra, JumpStart, Java et Solaris sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc., ou ses filiales, aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui, en outre, se conforment aux licences écrites de Sun.

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

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

080930@20795

Contents

Preface 9

Part I Planning to Install Over the Network 13

1 Where to Find Solaris Installation Planning Information 15
Where to Find Planning and System Requirement Information 15

2 Preconfiguring System Configuration Information (Tasks) 17
Advantages of Preconfiguring System Configuration Information 17
Preconfiguring With the `sysidcfg` File 18
▼ To Create a `sysidcfg` Configuration File 19
Syntax Rules for the `sysidcfg` File 21
`sysidcfg` File Keywords 22
SPARC: Preconfiguring Power Management Information 36

3 Preconfiguring With a Naming Service or DHCP 37
Choosing a Naming Service 37
Preconfiguring With the Naming Service 39
▼ To Preconfigure the Locale Using NIS 40
▼ To Preconfigure the Locale Using NIS+ 42
Preconfiguring System Configuration Information With the DHCP Service (Tasks) 43
Creating DHCP Options and Macros for Solaris Installation Parameters 44

Part II Installing Over a Local Area Network 55

4 Installing From the Network (Overview) 57
Network Installation Introduction 57
Required Servers for Network Installation 57
x86: Overview of Booting and Installing Over the Network With PXE 59
x86: What is PXE? 60
x86: Guidelines for Booting With PXE 60

5 Installing From the Network With DVD Media (Tasks) 61
Task Map: Installing From the Network With DVD Media 62
Creating an Install Server With DVD Media 64
▼ To Create an Install Server With SPARC or x86 DVD Media 64
Creating a Boot Server on a Subnet With a DVD Image 67
▼ To Create a Boot Server on a Subnet With a DVD Image 68
Adding Systems to Be Installed From the Network With a DVD Image 69
▼ To Add Systems to Be Installed From the Network With `add_install_client` (DVD) ... 70
Installing the System From the Network With a DVD Image 74
▼ SPARC: To Install the Client Over the Network (DVD) 75
▼ x86: To Install the Client Over the Network With GRUB (DVD) 77

6 Installing From the Network With CD Media (Tasks) 83
Task Map: Installing From the Network With CD Media 84
Creating an Install Server With SPARC or x86 CD Media 86
▼ SPARC: To Create an Install Server With SPARC or x86 CD Media 86
Creating a Boot Server on a Subnet With a CD Image 90
▼ To Create a Boot Server on a Subnet With a CD Image 91
Adding Systems to Be Installed From the Network With a CD Image 92
▼ To Add Systems to Be Installed From the Network With `add_install_client` (CDs) 93
Installing the System From the Network With a CD Image 97
▼ SPARC: To Install the Client Over the Network (CDs) 97
▼ x86: To Install the Client Over the Network With GRUB (CDs) 100

7 Patching the Miniroot Image (Tasks) 107
Patching the Miniroot Image (Tasks) 107
About the Miniroot Image (Overview) 107
▼ How To Patch the Miniroot Image 108
Patching the Miniroot Image (Example) 109
Patching the Miniroot Image 109

8 Installing Over the Network (Examples) 113
Network Installation Over the Same Subnet (Examples) 114

9 Installing From the Network (Command Reference) 123
Network Installation Commands 123
x86: GRUB Menu Commands for Installation 124

Part III Installing Over a Wide Area Network 129

10 WAN Boot (Overview) 131
What Is WAN Boot? 131
When to Use WAN Boot 132
How WAN Boot Works (Overview) 133
Sequence of Events in a WAN Boot Installation 133
Protecting Data During a WAN Boot Installation 135
Security Configurations Supported by WAN Boot (Overview) 137
Secure WAN Boot Installation Configuration 137
Insecure WAN Boot Installation Configuration 138

11 Preparing to Install With WAN Boot (Planning) 139
WAN Boot Requirements and Guidelines 139
Web Server Software Requirements and Guidelines 141
Server Configuration Options 141
Storing Installation and Configuration Files in the Document Root Directory 142
Storing Configuration and Security Information in the `/etc/netboot` Hierarchy 144
Storing the `wanboot-cgi` Program 147
Digital Certificate Requirements 147

WAN Boot Security Limitations 148
Gathering Information for WAN Boot Installations 148

12 Installing With WAN Boot (Tasks) 151
Installing Over a Wide Area Network (Task Maps) 151
Configuring the WAN Boot Server 155
Creating the Document Root Directory 155
Creating the WAN Boot Miniroot 156
Verifying WAN Boot Support on the Client 158
Installing the `wanboot` Program on the WAN Boot Server 160
Creating the `/etc/netboot` Hierarchy on the WAN Boot Server 162
Copying the WAN Boot CGI Program to the WAN Boot Server 165
▼ (Optional) To Configure the WAN Boot Logging Server 166
(Optional) Protecting Data by Using HTTPS 167
▼ (Optional) To Use Digital Certificates for Server and Client Authentication 168
▼ (Optional) To Create a Hashing Key and an Encryption Key 170
Creating the Custom JumpStart Installation Files 173
▼ To Create the Solaris Flash Archive 173
▼ To Create the `sysidcfg` File 175
▼ To Create the Profile 176
▼ To Create the `rules` File 178
(Optional) Creating Begin and Finish Scripts 180
Creating the Configuration Files 181
▼ To Create the System Configuration File 181
▼ To Create the `wanboot.conf` File 183
(Optional) Providing Configuration Information With a DHCP Server 187

13 SPARC: Installing With WAN Boot (Tasks) 189
Task Map: Installing a Client With WAN Boot 189
Preparing the Client for a WAN Boot Installation 190
▼ To Check the `net` Device Alias in the Client OBP 190
Installing Keys on the Client 192
Installing the Client 197
▼ To Perform a Noninteractive WAN Boot Installation 198
▼ To Perform an Interactive WAN Boot Installation 200

▼ To Perform a WAN Boot Installation With a DHCP Server 204
▼ To Perform a WAN Boot Installation With Local CD Media 206

14 SPARC: Installing With WAN Boot (Examples) 211
Sample Site Setup 212
Create the Document Root Directory 213
Create the WAN Boot Miniroot 213
Check the Client OBP for WAN Boot Support 213
Install the `wanboot` Program on the WAN Boot Server 214
Create the `/etc/netboot` Hierarchy 214
Copy the `wanboot-cgi` Program to the WAN Boot Server 215
(Optional) Configure the WAN Boot Server as a Logging Server 215
Configure the WAN Boot Server to Use HTTPS 215
Provide the Trusted Certificate to the Client 215
(Optional) Use Private Key and Certificate for Client Authentication 216
Create the Keys for the Server and the Client 217
Create the Solaris Flash Archive 217
Create the `sysidcfg` File 218
Create the Client's Profile 218
Create and Validate the `rules` File 219
Create the System Configuration File 219
Create the `wanboot.conf` File 220
Check the `net` Device Alias in OBP 221
Install Keys on the Client 222
Install the Client 223

15 WAN Boot (Reference) 225
WAN Boot Installation Commands 225
OBP Commands 228
System Configuration File Settings and Syntax 229
`wanboot.conf` File Parameters and Syntax 229

Part IV Appendixes 233

A Troubleshooting (Tasks) 235
Problems With Setting Up Network Installations 235
Problems With Booting a System 236
Booting From Media, Error Messages 236
Booting From Media, General Problems 237
Booting From the Network, Error Messages 238
Booting From the Network, General Problems 241
Initial Installation of the Solaris OS 241
▼ x86: To Check IDE Disk for Bad Blocks 242
Upgrading the Solaris OS 244
Upgrading, Error Messages 244
Upgrading, General Problems 245
▼ To Continue Upgrading After a Failed Upgrade 247
x86: Problems With Solaris Live Upgrade When You Use GRUB 247
▼ System Panics When Upgrading With Solaris Live Upgrade Running Veritas VxVm 249
x86: Service Partition Not Created by Default on Systems With No Existing Service Partition 251
▼ To Install Software From a Network Installation Image or From the Solaris DVD 251
▼ To Install From the Solaris Software - 1 CD or From a Network Installation Image 252

B Installing or Upgrading Remotely (Tasks) 253
SPARC: Using the Solaris Installation Program to Install or Upgrade From a Remote DVD-ROM or CD-ROM 253
▼ SPARC: To Install or Upgrade From a Remote DVD-ROM and CD-ROM 253

Glossary 257

Index 269

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 OS: Hardware Compatibility Lists* at `http://www.sun.com/bigadmin/hcl`. This document cites any implementation differences between the platform types.

In this document these x86 related terms mean the following:

- "x86" refers to the larger family of 64-bit and 32-bit x86 compatible products.
- "x64" points out specific 64-bit information about AMD64 or EM64T systems.
- "32-bit x86" points out specific 32-bit information about x86 based systems.

For supported systems, see the *Solaris OS: Hardware Compatibility Lists*.

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 10/08 Installation Guide: Basic Installations*.

Related Books

Table P–1 lists documentation for system administrators.

TABLE P–1 Are You a System Administrator Who is Installing Solaris?

Description	Information
Do you need system requirements or high-level planning information? Or want a high-level overview of Solaris ZFS™ installations, booting, Solaris Zones partitioning technology, or creating RAID-1 volumes?	*Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*
Do you need to install a single system from DVD or CD media? The Solaris installation program steps you through an installation.	*Solaris 10 10/08 Installation Guide: Basic Installations*
Do you need to upgrade or patch your system with almost no downtime? Save system downtime when upgrading by using Solaris Live Upgrade.	*Solaris 10 10/08 Installation Guide: Solaris Live Upgrade and Upgrade Planning*
Do you need to install a secure installation over the network or Internet? Use WAN boot to install a remote client. Or, do you need to install over the network from a network installation image? The Solaris installation program steps you through an installation.	*Solaris 10 10/08 Installation Guide: Network-Based Installations*
Do you need to install or patch multiple systems quickly? Use Solaris Flash™ software to create a Solaris Flash archive and install a copy of the OS on clone systems.	*Solaris 10 10/08 Installation Guide: Solaris Flash Archives (Creation and Installation)*
Do you need to back up your system?	Chapter 23, "Backing Up and Restoring UFS File Systems (Overview)," in *System Administration Guide: Devices and File Systems*
Do you need troubleshooting information, a list of known problems, or a list of patches for this release?	*Solaris Release Notes*
Do you need to verify that your system works on Solaris?	SPARC: *Solaris Sun Hardware Platform Guide*
Do you need to check on which packages have been added, removed, or changed in this release?	*Solaris Package List*
Do you need to verify that your system and devices work with Solaris SPARC and x86 based systems and other third-party vendors.	Solaris Hardware Compatibility List for x86 Platforms

Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- Documentation (`http://www.sun.com/documentation/`)
- Support (`http://www.sun.com/support/`)
- Training (`http://www.sun.com/training/`)

Typographic Conventions

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

TABLE P–2 Typographic Conventions

Typeface	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`
`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`** `Password:`
aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. A *cache* is a copy that is stored locally. Do *not* save the file. **Note:** Some emphasized items appear bold online.

Shell Prompts in Command Examples

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

TABLE P–3 Shell Prompts

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

PART I

Planning to Install Over the Network

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

◆ ◆ ◆ CHAPTER 1

Where to Find Solaris Installation Planning Information

This book describes how to install the Solaris OS remotely over a local area network or a wide area network.

This chapter describes the preparations for completing a successful installation. Many preparatory tasks are common to all Solaris installations, and so are described in one master planning document.

Where to Find Planning and System Requirement Information

The *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade* provides system requirements and high-level planning information, such as planning guidelines for file systems, and upgrade planning and much more. The following list describes the chapters in the planning book.

Chapter Descriptions From the Planning Guide	Reference
This chapter describes new features in the Solaris installation programs.	Chapter 2, "What's New in Solaris Installation," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*
This chapter provides you with information about decisions you need to make before you install or upgrade the Solaris OS. For example, you'll find information on deciding when to use a network installation image or DVD media and descriptions of all the Solaris installation programs.	Chapter 3, "Solaris Installation and Upgrade (Roadmap)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*
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. Upgrade limitations are also described.	Chapter 4, "System Requirements, Guidelines, and Upgrade (Planning)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*

Chapter Descriptions From the Planning Guide	Reference
This chapter contains checklists to help you gather all of the information that you need to install or upgrade your system. This is useful if you are doing an interactive installation. You'll have all the information in the checklist that you'll need to do an interactive installation.	Chapter 5, "Gathering Information Before Installation or Upgrade (Planning)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*
This book part includes chapters that provide overviews of several technologies that relate to a Solaris OS installation or upgrade. Guidelines and requirements related to these technologies are also included. These chapters include information about ZFS installations, booting, Solaris Zones partitioning technology, and RAID-1 volumes that can be created at installation.	Part II, "Understanding Installations That Relate to ZFS, Booting, Solaris Zones, and RAID-1 Volumes," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*

◆ ◆ ◆ CHAPTER 2

Preconfiguring System Configuration Information (Tasks)

This chapter describes how to preconfigure system information by using the `sysidcfg` file. 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" on page 17
- "Preconfiguring With the `sysidcfg` File" on page 18
- "SPARC: Preconfiguring Power Management Information" on page 36

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 naming service. Before the installation tools prompt you for configuration information, they check for configuration information that is stored elsewhere.

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

TABLE 2–1 Preconfiguration Options

Preconfiguration File or Service	Description	Further Information
`sysidcfg` file	Preset the domain name, netmask, DHCP, IPv6 and other parameters by using keywords in the `sysidcfg` file.	"Preconfiguring With the `sysidcfg` File" on page 18

TABLE 2–1 Preconfiguration Options *(Continued)*

Preconfiguration File or Service	Description	Further Information
Naming service	Preset host name and IP addresses by preconfiguring your system information in your naming service.	"Preconfiguring With the Naming Service" on page 39
DHCP	DHCP enables host system in a TCP/IP network to be configured automatically for the network as the system boots. DHCP can manage IP addresses by leasing them as needed to clients.	"Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 43

For more detailed information about choosing a preconfiguration method, see "Choosing a Naming Service" on page 37.

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 current Solaris release on one of the systems. You can specify the time zone in the `sysidcfg` file or the naming service databases. When you install the current Solaris release, the installation program does not prompt you to type a time zone.

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 22.

Note – The `name_service` keyword in the `sysidcfg` file automatically sets the naming service during installation of the Solaris OS. This setting overrides SMF services previously set up for `site.xml`. Therefore, you might need to reset your name service after installation.

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.

TABLE 2–2 sysidcfg Locations

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 current Solaris release.
UFS or PCFS diskette	Place the sysidcfg file in the root (/) directory on the diskette. 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 10/08 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.
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.

You can use the naming service or DHCP to preconfigure your system. For information, see Chapter 3, "Preconfiguring With a Naming Service or DHCP."

▼ To Create a sysidcfg Configuration File

1. **Create a file called sysidcfg in a text editor with the keywords you want.**
2. **Make the sysidcfg file available to clients by using one of the locations described at Table 2–2.**

Example 2–1 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 have been preconfigured by editing the naming 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. In this example, the

NFSv4 domain name is automatically derived from the naming service. Because the `service_profile` keyword is not included in this example, configuration is not altered for the network services during installation.

```
keyboard=US-English
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)}
nfs4_domain=dynamic
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 2–2 x86: `sysidcfg` File

The following sample `sysidcfg` file is for a group of x86 based systems. In this example, the NFSv4 domain name is specified to be `example.com`. This custom name overrides the default domain name. Also in this example, the network services are disabled or restricted to local connections only.

```
keyboard=US-English
timezone=US/Central
timeserver=timehost1
terminal=ibm-pc
service_profile=limited_net

name_service=NIS {domain_name=marquee.central.example.com
                  name_server=nmsvr2(172.25.112.3)}
nfs4_domain=example.com
root_password=URFUni9
```

Example 2–3 `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. In this example, the NFSv4 domain name is automatically derived from the naming service.

```
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)}
nfs4_domain=dynamic
```

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 4, "Installing From the Network (Overview)."

If you plan to use the `sysidcfg` file in a WAN boot installation, you need to perform additional tasks. For more information, see Chapter 10, "WAN Boot (Overview)."

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 2, "Custom JumpStart (Overview)," in *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

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

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 2–3 lists the keywords you can use to configure system information in the `sysidcfg` file.

TABLE 2–3 Keywords to Use in `sysidcfg`

Configuration Information	Keyword
Keyboard layout and language	“`keyboard` Keyword” on page 23
Naming service, domain name, name server	“`name_service` Keyword” on page 24
Network interface, host name, Internet Protocol (IP) address, netmask, DHCP, IPv6	“`network_interface` Keyword” on page 27
Domain name definition for NFSv4	“`nfs4_domain` Keyword” on page 32
Root password	“`root_password` Keyword” on page 33
Security policy	“`security_policy` Keyword” on page 33
Network security profile	“`service_profile` Keyword” on page 34
Language in which to display the install program and desktop	“`system_locale` Keyword” on page 35
Terminal type	“`terminal` Keyword” on page 35
Time zone	“`timezone` Keyword” on page 35

TABLE 2–3 Keywords to Use in sysidcfg *(Continued)*

Configuration Information	Keyword
Date and time	"timeserver Keyword" on page 35

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

keyboard Keyword

The sysidkdb tool configures your USB language and its corresponding keyboard layout.

The following procedure occurs:

- If the keyboard is self-identifying, the keyboard language and layout automatically configures during installation.
- If the keyboard is not self-identifying, the sysidkdb tool provides you with a list of supported keyboard layouts during installation, so that you can select a layout for keyboard configuration.

Note – PS/2 keyboards are not self-identifying. You will be asked to select the keyboard layout during the installation.

You can configure the keyboard language and its corresponding keyboard layout information by using the keyboard keyword. Each language has its own keyboard layout. Use the following syntax to select a language and its corresponding layout.

```
keyboard=keyboard_layout
```

For example, this entry sets the keyboard language and its corresponding keyboard layout for the German language:

```
keyboard=German
```

The value provided for *keyboard_layout* must be a valid value. Otherwise, an interactive response is required during installation. The valid *keyboard_layout* strings are defined in the /usr/share/lib/keytables/type_6/kbd_layouts file.

SPARC only – Previously, the USB keyboard assumed a self-identifying value of 1 during the installation. Therefore, all of the keyboards that were not self-identifying always configured for a U.S. English keyboard layout during installation.

If the keyboard is not self-identifying and you want to prevent being prompted during your JumpStart installation, select the keyboard language in your sysidcfg file. For JumpStart

installations, the default is for the U.S. English language. To select another language and its corresponding keyboard layout, set the keyboard entry in your sysidcfg file as in the example provided above.

For further information, see the sysidcfg(4) and sysidtool(1M) man pages.

name_service Keyword

You can use the name_service keyword to configure the naming 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 {}.

Note – The name_service option in the sysidcfg file automatically sets the naming service during installation of the Solaris OS. This setting overrides SMF services that were previously set up for site.xml. Therefore, you might need to reset your name service after installation.

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

NIS Syntax for name_service Keyword

Use the following syntax to configure the system to use the NIS naming 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 2–4 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 2–5 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 naming service information. You can specify up to six domain names to search. The total length of each search entry cannot exceed 250 characters.

EXAMPLE 2–6 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 naming service information.

```
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 2–7 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.

EXAMPLE 2–7 Specifying an LDAP Server With the name_service Keyword *(Continued)*

```
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 the default router 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 43.

  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 use either 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 2–8 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 2–9 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.

EXAMPLE 2–9 Configuring a Single Interface By Specifying Configuration Information With the network_interface Keyword *(Continued)*

- 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 43.

  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 2–10 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}
```

nfs4_domain Keyword

To prevent being asked to specify an NFSv4 domain name during installation, use the nfs4_domain keyword in the sysidcfg file. This keyword suppresses selection of a domain name during the installation process. Use the following syntax:

nfs4_domain=dynamic or *custom_domain_name*

dynamic

This reserved keyword dynamically derives the NFSv4 domain name, based on naming services configuration. For example:

```
nfs4_domain=dynamic
```

This example enables the domain name to be derived by the naming service.

The reserved keyword, dynamic, is not case sensitive.

Note – By default, NFSv4 uses a domain name that is automatically derived from the system's naming services. This domain name is sufficient for most configurations. In a few cases, mount points that cross domain boundaries can cause files to appear to be owned by "nobody" because no common domain name exists. To prevent this situation, you can override the default domain name and select a custom domain name.

---

custom_domain_name

This value overrides the default domain name.

This value must be a valid custom domain name. A valid domain name is composed of a combination of alphanumeric characters,

dots, underscores, and dashes only. The first character must be an alphabetical character. For example:

```
nfs4_domain=example.com
```

This example sets the value that is used by the `nfsmapid` daemon to be *example.com*. This selection overrides the default domain name.

Note – In previous releases, scripts enabled users to avoid being prompted for the NFSv4 domain name during installation.

For JumpStart installations in the Solaris 10 OS, you could use the workaround JumpStart sample script, `set_nfs4_domain`, to suppress the NFSv4 prompt during installation. This script is no longer required. Use the `sysidcfg` keyword, `nfs4_domain`, instead.

In prior releases, the `/etc/.NFS4inst_state.domain` file was created by the `sysidnfs4` program. This file would suppress the prompt for an NFSv4 domain name during installation. This file is no longer created. Use the `sysidcfg` keyword, `nfs4_domain`, instead.

`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 2–11 Configuring the System to Use Kerberos With the security_policy Keyword

This 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}
```

service_profile Keyword

You can use the service_profile keyword to install a more secure system by restricting network services. This security option is available only for initial installations. An upgrade maintains all previously set services.

Use one of the following syntaxes to set this keyword.

```
service_profile=limited_net
```

```
service_profile=open
```

limited_net specifies that all network services, except for Secure Shell, are either disabled or constrained to respond to local requests only. After installation, any individual network service can be enabled by using the svcadm and svccfg commands.

open specifies that no network service changes are made during installation.

If the service_profile keyword is not present in the sysidcfg file, no changes are made to the status of the network services during installation.

The network services can be enabled after installation by using the netservices open command or by enabling individual services by using SMF commands. See "Revising Security Settings After Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

For further information about limiting network security during installation, see "Planning Network Security" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*. See also the following man pages.

- netservices(1M)
- svcadm(1M)
- svccfg(1M) commands

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 2–12 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.

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.

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 current Solaris release 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 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

◆ ◆ ◆ CHAPTER 3

Preconfiguring With a Naming Service or DHCP

This chapter describes procedures about preconfiguring system information with a naming service or DHCP. This chapter contains the following sections:

- "Choosing a Naming Service" on page 37
- "Preconfiguring With the Naming Service" on page 39
- "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 43

Choosing a Naming Service

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

- A `sysidcfg` file on a remote system or diskette

Note – The `name_service` option in the `sysidcfg` file automatically sets the naming service during installation of the Solaris OS. This setting overrides SMF services previously setup for `site.xml`. Therefore, you might need to reset your name service after installation.

- The naming 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 43.

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

TABLE 3–1 Methods to Preconfigure System Configuration Information

Preconfigurable System Information	Preconfigurable With the `sysidcfg` File?	Preconfigurable With the Naming Service?
Naming service	Yes	Yes
Domain name	Yes	No
Name server	Yes	No
Network interface	Yes	No
Host name	Yes	Yes
	Because this information is system specific, edit the naming service rather than create a different `sysidcfg` file for each system.	
Internet Protocol (IP) address	Yes	Yes
	Because this information is system specific, edit the naming service rather than create a different `sysidcfg` file for each system.	
Netmask	Yes	No
DHCP	Yes	No
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	No
	You can configure this information with the Solaris installation program, but not through the `sysidcfg` file or the naming service.	
x86: Monitor type	Yes	No

TABLE 3–1 Methods to Preconfigure System Configuration Information *(Continued)*

Preconfigurable System Information	Preconfigurable With the `sysidcfg` File?	Preconfigurable With the Naming Service?
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 naming service. "SPARC: Preconfiguring Power Management Information" on page 36 contains details.	No	No

Preconfiguring With the Naming Service

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

System Information to Preconfigure	Naming 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 below to use your naming service to preconfigure the locale for a system.

Note – In order to successfully preconfigure your system locale by using NIS or NIS+, the following requirements must be met:

- You must boot the system from the network by using the following command:

  ```
  ok boot net
  ```

  You can use options with this command. See details in Step 2 of the"SPARC: To Install the Client Over the Network (DVD)" on page 75 procedure.
- The NIS or NIS+ server must be available to access during the installation.

If these requirements are fulfilled, the installer uses the preconfigured settings and does not prompt for the locale during the installation. If either requirement is not fulfilled, the installer prompts for your locale information during the installation.

- "To Preconfigure the Locale Using NIS" on page 40
- "To Preconfigure the Locale Using NIS+" on page 42

▼ To Preconfigure the Locale Using NIS

1. **Become superuser or assume an equivalent role 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:

- **Enter `locale domain_name`.**

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

  ```
  fr example.com
  ```

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

- **Or, enter `locale system_name`.**

  The following example 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 DVD or Solaris 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 4, "Installing From the Network (Overview)."

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 2, "Custom JumpStart (Overview)," in *Solaris 10 10/08 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)*.

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=name locale=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 DVD or Solaris 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 4, "Installing From the Network (Overview)."

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 2, "Custom JumpStart (Overview)," in *Solaris 10 10/08 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. All SPARC based systems that are supported by the Solaris OS 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 3–2 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 4, "Installing From the Network (Overview)"
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 69 Using Solaris CD: "Adding Systems to Be Installed From the Network With a CD Image" on page 92 `add_install_client(1M)`
Prepare your network to use the DHCP service.	Decide how you want to configure your DHCP server.	Chapter 13, "Planning for DHCP Service (Tasks)," in *System Administration Guide: IP Services*
Configure the DHCP server.	Use DHCP Manager to configure your DHCP server	Chapter 14, "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 44

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.

You can customize the options and 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.
- **Network-specific installations** – You can instruct the DHCP service to perform a network installation for all clients in a specific network. For example, you can define a DHCP macro that performs the same installation on all systems in the 192.168.2 network.
- **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 use a DHCP server for a network installation, 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 69.
- For network installations that use CD media, see "Adding Systems to Be Installed From the Network With a CD Image" on page 92.

DHCP Options and Macro Values

To install DHCP clients from the network, you must create Vendor category options to pass information that is needed to install the Solaris OS. The following tables describe common DHCP options that you can use to install a DHCP client.

- You can use the standard DHCP options that are listed in Table 3–3 to configure and install x86 based systems. These options are not platform specific, and can be used to install the Solaris OS on a variety of x86 based systems. Use these options to install the Solaris 10 release on x86 based systems by using DHCP. For a complete list of standard options, see `dhcp_inittab(4)`.
- Table 3–4 lists options that you can use to install Sun client systems. The vendor client classes that are listed in this 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 3–3 Values for Standard DHCP Options

Option Name	Code	Data Type	Granularity	Maximum	Description
`BootFile`	N/A	ASCII	1	1	Path to the client's boot file
`BootSrvA`	N/A	IP address	1	1	IP address of boot server
`DNSdmain`	15	ASCII	1	0	DNS domain name
`DNSserv`	6	IP address	1	0	List of DNS name servers
`NISdmain`	40	ASCII	1	0	NIS domain name
`NISservs`	41	IP address	1	0	IP address of NIS server
`NIS+dom`	64	ASCII	1	0	NIS+ domain name
`NIS+serv`	65	IP address	1	0	IP address of NIS+ server
`Router`	3	IP address	1	0	IP addresses of network routers

TABLE 3–4 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.						
Note – 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.						
`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

TABLE 3–4 Values for Creating Vendor Category Options for Solaris Clients *(Continued)*

Name	Code	Data Type	Granularity	Maximum	Vendor Client Classes *	Description
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.						
Note – 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.						
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*
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`

TABLE 3–4 Values for Creating Vendor Category Options for Solaris Clients *(Continued)*

Name	Code	Data Type	Granularity	Maximum	Vendor Client Classes *	Description
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.						
Note – 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.						
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
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 3–5 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
sun4v	Solaris and sparc macros
i86pc	Solaris macro, SrootPTH, SinstPTH, SbootFIL

TABLE 3–5 Sample Macros to Support Network Installation Clients *(Continued)*

Macro Name	Contains These Options and Macros
`SUNW.i86pc`	`i86pc` macro **Note –** The `SUNW.i86pc` vendor client class is only valid for the Solaris 10 3/05 release and compatible versions.
`SUNW.Sun-Blade-1000`	`sun4u` macro, `SbootFIL`
`SUNW.Sun-Fire-880`	`sun4u` macro, `SbootFIL`
`PXEClient:Arch:00000:UNDI:002001`	`BootSrvA`, `BootFile`
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.
`01`*client-MAC-address* client-specific macros (for example, `010007E9044ABF`)	`BootSrvA`, `BootFile`

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.

- Create the options and macros in DHCP Manager. See "Using DHCP Manager to Create Install Options and Macros" on page 49 for instructions about how to create options and macros in DHCP Manager.
- 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 52 for information about how to write scripts that create these options and macros.

Note that the total size of the vendor options that are provided to a particular client must not exceed 255 bytes, including the option codes and length information. This is a limitation of the current Solaris DHCP protocol implementation. 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 use the shorter link names.

Using DHCP Manager to Create Install Options and Macros

You can use DHCP Manager to create the options that are listed in Table 3–4 and the macros that are listed in Table 3–5.

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

Before You Begin Perform the following tasks before you create DHCP macros for your installation.

- Add the clients that you want to install with DHCP as install clients of your network installation server. For information about how to add a client to an install server, see Chapter 4, "Installing From the Network (Overview)."
- Configure your DHCP server. If you have not configured your DHCP server, see Chapter 13, "Planning for DHCP Service (Tasks)," in *System Administration Guide: IP Services*.

1. **Become superuser or assume an equivalent role 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 panel opens.

5. **Type the option name for the first option, then type values appropriate for that option.**

   Use the output of the `add_install_client` command , Table 3–3, and Table 3–4 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 panel 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 3–3 and Table 3–4 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)

Before You Begin Perform the following tasks before you create DHCP macros for your installation.

- Add the clients that you want to install with DHCP as install clients of your network installation server. For information about how to add a client to an install server, see Chapter 4, "Installing From the Network (Overview)."
- Configure your DHCP server. If you have not configured your DHCP server, see Chapter 13, "Planning for DHCP Service (Tasks)," in *System Administration Guide: IP Services*.
- Create the DHCP options that you want to use in your macro. For instructions about how to create DHCP options, see "How to Create Options to Support Solaris Installation (DHCP Manager)" on page 50.

1. **Select the Macros tab in DHCP Manager.**
2. **Choose Create from the Edit menu.**

   The Create Macro panel opens.
3. **Type the name of a macro.**

   See Table 3–5 for macro names you might use.
4. **Click the Select button.**

   The Select Option panel 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 3–3 and Table 3–4 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 4, "Installing From the Network (Overview)."

If you plan to use DHCP in a WAN boot installation, you need to perform additional tasks. For more information, see Chapter 10, "WAN Boot (Overview)."

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 2, "Custom JumpStart (Overview)," in *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

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

Writing a Script That Uses `dhtadm` to Create Options and Macros

You can create a Korn shell script by adapting the example in Example 3–1 to create all the options listed in Table 3–3 and Table 3–4 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 3–1 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. Note that the
# SUNW.i86pc option only applies for the Solaris 10 3/05 release.
# 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'
```

EXAMPLE 3–1 Sample Script to Support Network Installation *(Continued)*

```
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 \
'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.
# Note that this macro applies only for the Solaris 10 3/05 release.
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.
# Note that this class only applies for the Solaris 10 3/05 release.
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.
```

EXAMPLE 3–1 Sample Script to Support Network Installation *(Continued)*

```
# 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_
# Create a macro for PXE clients that want to boot from our boot server.
# Note that this macro applies for the Solaris 10 3/05 release.
dhtadm -A -m PXEClient:Arch:00000:UNDI:002001 -d \
:BootFile=nbp.i86pc:BootSrvA=10.21.0.2:
# Create a macro for PXE clients that want to boot from our boot server.
# Note that this macro applies for the Solaris 10 2/06 release.
dhtadm -A -m PXEClient:Arch:00000:UNDI:002001 -d \
:BootFile=i86pc:BootSrvA=10.21.0.2:
# Create a macro for the x86 based client with the Ethernet address 00:07:e9:04:4a:bf
# to install from the network by using PXE.
dhtadm -A -m 010007E9044ABF -d :BootFile=010007E9044ABF:BootSrvA=10.21.0.2:
# 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)`.

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 4

Installing 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. This chapter provides overview information on the following topics.

- "Network Installation Introduction" on page 57
- "x86: Overview of Booting and Installing Over the Network With PXE" on page 59

For information on how to install a client over a wide area network, see Chapter 10, "WAN Boot (Overview)."

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 current Solaris release disc images. You copy the contents of the current Solaris release 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 current Solaris release disc images from which you can install current Solaris release on other systems on the network. You create an install server by copying the images from the following media:
  - Solaris DVD
  - Solaris Software CDs

After you copy the image from the Solaris Software CDs, you can also copy the image from the Solaris Languages CDs 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.

- "To Create an Install Server With SPARC or x86 DVD Media" on page 64
- "SPARC: To Create an Install Server With SPARC or x86 CD Media" on page 86

- **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 current Solaris release 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 current Solaris release boot software for multiple releases, including the current Solaris release boot software for different platforms. For example, a SPARC boot server can provide the Solaris 9 and current Solaris release boot software for SPARC based systems. The same SPARC boot server can also provide the current Solaris release boot software for x86 based systems.

  Note – When using DHCP, you do not need to create a separate boot server. For more information, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 43.

  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 67
  - "Creating a Boot Server on a Subnet With a CD Image" on page 90

- **(Optional) DHCP server –** A server that uses the Dynamic Host Configuration Protocol (DHCP) to provide the network parameters that are necessary for installation. You can configure a DHCP server to configure and install specific clients, all clients on a specific network, or an entire class of clients. 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 43.

- **(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 4–1 illustrates the servers that are typically used for network installation. Note that this sample network does not include a DHCP server.

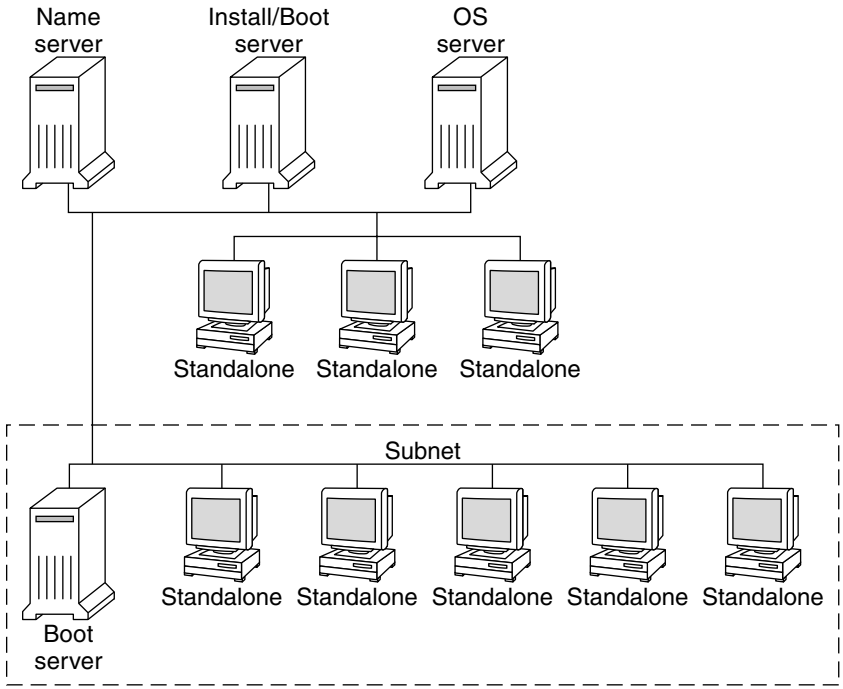

FIGURE 4–1 Network Installation Servers

x86: Overview of Booting and Installing Over the Network With PXE

This section provides an overview of the Preboot Execution Environment (PXE).

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.

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 have a variety of shortcomings. If you experience difficulty with a particular PXE 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.

◆ ◆ ◆ CHAPTER 5

Installing 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 current Solaris release disc images, called an install server, to other systems on the network. You copy the contents of the current Solaris release 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: Installing From the Network With DVD Media" on page 62
- "Creating an Install Server With DVD Media" on page 64
- "Creating a Boot Server on a Subnet With a DVD Image" on page 67
- "Adding Systems to Be Installed From the Network With a DVD Image" on page 69
- "Installing the System From the Network With a DVD Image" on page 74

Note –

- **Starting with the Solaris 10 11/06 release**, you have the option during an initial installation to change the network security settings so that all network services, except Secure Shell, are disabled or restricted to respond to local requests only. This security option is available only during an initial installation, not during an upgrade. An upgrade maintains any previously set services. If necessary, you can restrict network services after an upgrade by using the `netservices` command. See "Planning Network Security" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

  The network services can be enabled after installation by using the `netservices open` command or by enabling individual services by using SMF commands. See "Revising Security Settings After Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.
- **Starting with the Solaris 10 10/08 release**, the structure of the Solaris DVD and Solaris Software - 1 CD has changed for the SPARC platform. Slice 0 is no longer at the top of the directory structure. Therefore, the structure of the x86 and SPARC DVDs and Solaris Software - 1 CD are the same. This change in structure makes setting up an install server easier if you have a mix of platforms, such as a SPARC install server and x86 media.

Task Map: Installing From the Network With DVD Media

TABLE 5–1 Task Map: Setting Up an Install Server With DVD Media

Task	Description	For Instructions
(x86 only): Verify that your system supports PXE.	If you want to install an x86 based system over the network, confirm that your machine can use PXE to boot without local boot media. If your x86 based system does not support PXE, you must boot the system from a local DVD or CD.	Check your hardware manufacturer's documentation or the system BIOS.
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" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*

TABLE 5–1 Task Map: Setting Up an Install Server With DVD Media *(Continued)*

Task	Description	For Instructions
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 5, "Gathering Information Before Installation or Upgrade (Planning)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*
(Optional) Preconfigure system information.	You can preconfigure system information to avoid being prompted for the information during the installation or upgrade.	Chapter 2, "Preconfiguring System Configuration Information (Tasks)"
Create an install server.	Use the `setup_install_server(1M)` command to copy the Solaris DVD to the install server's hard disk.	"Creating an Install Server With DVD Media" on page 64
(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 67
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 69
(Optional) Configure the DHCP server.	If you want to use DHCP to provide system configuration and installation parameters, configure the DHCP server, then create the appropriate options and macros for your installation. **Note –** If you want to install an x86 based system from the network with PXE, you must configure a DHCP server.	Chapter 13, "Planning for DHCP Service (Tasks)," in *System Administration Guide: IP Services* "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 43
Install the system over the network.	Begin the installation by booting the system from the network.	"Installing the System From the Network With a DVD Image" on page 74

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.

▼ To Create an 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*.

1 On the system that is to become the install server, become superuser or assume an equivalent role.

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

2 Insert the Solaris DVD in the system's drive.

3 Create a directory to contain the DVD 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.

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

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 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.

  Using the `share` command, add 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 current Solaris release, 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 DVD.

9 (Optional) Patch the files that are located in the miniroot on the net install image that was created by `setup_install_server`.

Patching a file might be necessary if a boot image has problems. For step-by-step procedures, see Chapter 7, "Patching the Miniroot Image (Tasks)."

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 69.**
- **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 67.**

Example 5–1 SPARC: Creating an Install Server With a DVD

The following example illustrates how to create an install server by copying the Solaris DVD to the install server's `/export/home/dvd` directory. This example assumes that the install server is running the current Solaris release.

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

If you need a separate boot server, make the install server available to the boot server.

Using the `share` command, add this entry 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 70.

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 67.

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 69.
- 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

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

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

2 **Mount the Solaris 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 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 69.

Example 5–2 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 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 69.

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. 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 naming 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 for a SPARC client, see Example 5–3.
- If your install server and client are on the same subnet, see Example 5–4.

- If your install server and your client are not on the same subnet and you are not using DHCP, see Example 5–5.
- If you are using DHCP to set installation parameters for x86 clients, see Example 5–6.
- If you want to use a specific serial port to display output during the installation of an x86 based system, see Example 5–7.

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 naming service to preconfigure system information
- The profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method

1 On the install server or boot server, become superuser or assume an equivalent role.

2 If you use the NIS, NIS+, DNS, or LDAP naming service, verify that the following information about the system to be installed has been added to the naming service.

- Host name
- IP address
- Ethernet address

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

3 Add the client to the install server's `/etc/ethers` file.

a. On the client, find the ethers address. The `/etc/ethers` map is taken from the local file.

```
# ifconfig -a grep ether
ether 8:0:20:b3:39:1d
```

b. On the install server, open the `/etc/ethers` file in an editor. Add the address to the list.

4 Change to the `Tools` directory on the Solaris DVD image:

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

install_dir_path Specifies the path to the `Tools` directory

5 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. The output of this option lists the DHCP options you need to create on the DHCP server.

For more information about class-specific installations by using DHCP, see "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 44.

`-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 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 current Solaris release 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, including a boot file for that client.

The `nbp.` prefix is not used in boot file names. For example, if you specify `-e 00:07:e9:04:4a:bf` for an x86 based client, the command creates the boot file `010007E9044ABF.i86pc` in the `/tftpboot` directory. However, the current Solaris release supports the use of legacy boot files with the `nbp.` prefix.

For more information about client-specific installations by using DHCP, see "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 44.

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" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

Example 5–3 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)" on page 43.

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

Example 5–4 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 5–5 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 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 5–6 x86: Adding a Single 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 `-e` option indicates that this installation will only occur on the client with the Ethernet address 00:07:e9:04:4a:bf .
- 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 Operating System for x86 Platforms DVD image in `/export/home/dvdx86`.

```
x86_install_server# cd /export/boot/dvdx86/Solaris_10/Tools
x86_install_server# ./add_install_client -d -e 00:07:e9:04:4a:bf \
-s rosemary:/export/home/dvdx86 i86pc
```

The previous commands set up the client with the Ethernet address `00:07:e9:04:4a:bf` as an installation client. The boot file `010007E9044ABF.i86pc` is created on the installation server. In previous releases, this boot file was named `nbp.010007E9044ABF.i86pc`.

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)" on page 43.

Example 5–7 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 `-b` option instructs the installation program to use the serial port `ttya` as an input and an output device.

Use this set of commands to add the client.

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

More Information Continuing the Installation

If you are using a DHCP server to install the x86 based client over the network, configure the DHCP server and create the options and macros that are listed in the output of the `add_install_client -d` command. For instructions about how to configure a DHCP server to support network installations, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 43.

x86 based systems: If you are not using a DHCP server, you must boot the system from a local Solaris OS DVD or CD.

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

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.

- See "SPARC: To Install the Client Over the Network (DVD)" on page 75 for instructions about how to boot and install SPARC based systems over the network.
- See "x86: To Install the Client Over the Network With GRUB (DVD)" on page 77 for instructions about how to boot and install x86 based systems over the network.

▼ SPARC: To Install the Client Over the Network (DVD)

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 DVD media, see "To Create an Install Server With SPARC or x86 DVD Media" on page 64.
- 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 boot server, see "Creating a Boot Server on a Subnet With a DVD Image" on page 67. 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 43.
- 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" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

    Note – If you have a system that contains non-global zones, Solaris Live Upgrade is the recommended upgrade program or program to add patches. Other upgrade programs might require extensive upgrade time, because the time required to complete the upgrade increases linearly with the number of installed non-global zones.

    For information about upgrading with Solaris Live Upgrade, see Part I, "Upgrading With Solaris Live Upgrade," in *Solaris 10 10/08 Installation Guide: Solaris Live Upgrade and Upgrade Planning*.

  - 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 18.
  - Set up a name server if you use a naming service to preconfigure system information. For information about how to preconfigure information with a naming service, see "Preconfiguring With the Naming Service" on page 39.
  - 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 3, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

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
  ```

- **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 2, "Preconfiguring System Configuration Information (Tasks)," for more information.
- If you did not preconfigure all the system information, use the "Checklist for Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade* to help you answer the configuration questions.

Note – If the keyboard is self-identifying, the keyboard layout automatically configures during installation. If the keyboard is not self-identifying, you can select from a list of supported keyboard layouts during installation.

PS/2 keyboards are not self-identifying. You will be asked to select the keyboard layout during the installation.

For further information, see "`keyboard` Keyword" on page 23.

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

4 **If you are prompted, answer any additional questions to complete your installation.**

- **If you preconfigured all of the installation options, the installation program does not prompt you to enter any installation information. See Chapter 2, "Preconfiguring System Configuration Information (Tasks)," for more information.**
- **If you did not preconfigure all the installation options, use the "Checklist for Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade* to help you answer the installation questions.**

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 With GRUB" in *Solaris 10 10/08 Installation Guide: Basic Installations*.

▼ x86: To Install the Client Over the Network With GRUB (DVD)

The Solaris installation programs for x86 based systems use the GRUB boot loader. This procedure describes how to install an x86 based system over the network with the GRUB boot loader. For overview information about the GRUB boot loader, see Chapter 7, "SPARC and x86 Based Booting (Overview and Planning)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

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 DVD media, see "To Create an Install Server With SPARC or x86 DVD Media" on page 64.
- 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 boot server, see "Creating a Boot Server on a Subnet With a DVD Image" on page 67. 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 43.
- 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" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

Note – If you have a system that contains non-global zones, Solaris Live Upgrade is the recommended upgrade program or program to add patches. Other upgrade programs might require extensive upgrade time, because the time required to complete the upgrade increases linearly with the number of installed non-global zones.

For information about upgrading with Solaris Live Upgrade, see Part I, "Upgrading With Solaris Live Upgrade," in *Solaris 10 10/08 Installation Guide: Solaris Live Upgrade and Upgrade Planning*.

- 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 18.
- Set up a name server if you use a naming service to preconfigure system information. For information about how to preconfigure information with a naming service, see "Preconfiguring With the Naming Service" on page 39.
- 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 3, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

This procedure also assumes that your system can boot from the network.

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. The GRUB menu is displayed.

Note – The GRUB menu that is displayed on your system might vary from the following sample, depending on the configuration of your network installation server.

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-------------------------------------------------------------------------------------------------------------+
```

```
| Solaris 10 10 10/08 /cdrom0
|                                                                                                                    |
|                                                                                                                    |
+--------------------------------------------------------------------------------------------------------------------+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

5 Select the appropriate installation option.

- **To install the Solaris OS from the network, select the appropriate Solaris entry on the menu, then press Enter.**

  Select this entry if you want to install from the network installation server you set up in "To Create an Install Server With SPARC or x86 DVD Media" on page 64.

- **To install the Solaris OS from the network with specific boot arguments, follow these steps.**

  You might need to set specific boot arguments if you want to modify the device configuration during the installation, and did not set these boot arguments previously with the `add_install_client` command as described in "To Add Systems to Be Installed From the Network With `add_install_client` (DVD)" on page 70.

  a. On the GRUB menu, select the installation option you want to edit, then press e.

  Boot commands that are similar to the following text are displayed in the GRUB menu.

  ```
  kernel /I86pc.Solaris_10/multiboot kernel/unix \
  -B install_media=192.168.2.1:/export/cdrom0/boot \
  module /platform/i86pc/boot_archive
  ```

  b. Use the arrow keys to select the boot entry that you want to edit, then press e.

  The boot command that you want to edit is displayed in the GRUB edit window.

  c. Edit the command by typing the boot arguments or options you want to use.

  The command syntax for the Grub edit menu is as follows.

  ```
  grub edit>kernel /image_directory/multiboot kernel/unix/ \
  install [url|ask] -B options install_media=media_type
  ```

  For information about boot arguments and command syntax, see Table 9–1.

  d. To accept your edits and return to the GRUB menu, press Enter.

  Note – To cancel your edits and return to the GRUB menu, press Escape.

  The GRUB menu is displayed. The edits you made to the boot command are displayed.

  e. To begin the installation, type b in the GRUB menu.

The Solaris installation program checks the default boot disk for the requirements to install or upgrade the system. If the Solaris installation cannot detect the system configuration, the program prompts you for any missing information.

When the check is completed, the installation selection screen is displayed.

6 Select an installation type.

The installation selection screen displays the following options.

```
Select the type of installation you want to perform:

   1 Solaris Interactive
   2 Custom JumpStart
   3 Solaris Interactive Text (Desktop session)
   4 Solaris Interactive Text (Console session)
   5 Apply driver updates
   6 Single user shell

   Enter the number of your choice followed by the <ENTER> key.
   Alternatively, enter custom boot arguments directly.

   If you wait 30 seconds without typing anything,
   an interactive installation will be started.
```

- **To install the Solaris OS, choose from the following options.**
  - **To install with the Solaris interactive installation GUI, type 1, then press Enter.**
  - **To install with the interactive text installer in a desktop session, type 3, then press Enter.**
    Select this installation type to override the default GUI installer and run the text installer.
  - **To install with the interactive text installer in a console session, type 4, then press Enter.**
    Select this installation type to override the default GUI installer and run the text installer.

If you want to perform an unattended custom JumpStart installation (option 2), see *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

For detailed information about the Solaris installation GUI and text installer, see "System Requirements and Recommendations" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

The system configures the devices and interfaces, and searches for configuration files.The installation program begins. Go to Step 7 to continue the installation.

- **To perform system administration tasks before your installation, choose from the following options.**

  - **To update drivers or install an install time update (ITU), insert the update media, type 5, then press Enter.**

    You might need to update drivers or install an ITU to enable the Solaris OS to run on your system. Follow the instructions for your driver update or ITU to install the update.

  - **To perform system administration tasks, type 6, then press Enter.**

    You might want to launch a single user shell if you need to perform any system administration tasks on your system before you install. For information about system administration tasks you can perform prior to installation, see *System Administration Guide: Basic Administration*.

  After you perform these system administration tasks, the previous list of options is displayed. Select the appropriate option to continue the installation.

7 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 2, "Preconfiguring System Configuration Information (Tasks)," for more information.
- If you did not preconfigure all the system information, use the "Checklist for Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade* to help you answer the configuration questions.

Note – If the keyboard is self-identifying, the keyboard layout automatically configures during installation. If the keyboard is not self-identifying, you can select from a list of supported keyboard layouts during installation.

For further information, see "`keyboard` Keyword" on page 23.

Note – During installation, you can choose the default NFSv4 domain name. Or, you can specify a custom NFSv4 domain name. For further information, see "`nfs4_domain` Keyword" on page 32.

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

8 **If you are prompted, answer any additional questions to complete your installation.**

- **If you preconfigured all of the installation options, the installation program does not prompt you to enter any installation information. See Chapter 2, "Preconfiguring System Configuration Information (Tasks)," for more information.**
- **If you did not preconfigure all the installation options, use the "Checklist for Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade* to help you answer the installation questions.**

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

Note – When you boot the system after the installation, a GRUB menu lists the operating systems that are installed, including the newly-installed Solaris OS. Select which operating system you want to boot. The default selection loads if you do not make another selection.

More Information

Next Steps

If you install multiple operating systems on your machine, you need to instruct the GRUB boot loader to recognize these operating systems in order to boot. For more information, see "Modifying Boot Behavior on x86 Based Systems" in *System Administration Guide: Basic Administration*.

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 With GRUB" in *Solaris 10 10/08 Installation Guide: Basic Installations*.

◆ ◆ ◆ CHAPTER 6

Installing 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 current Solaris release 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: Installing From the Network With CD Media" on page 84
- "Creating an Install Server With SPARC or x86 CD Media" on page 86
- "Creating a Boot Server on a Subnet With a CD Image" on page 90
- "Adding Systems to Be Installed From the Network With a CD Image" on page 92
- "Installing the System From the Network With a CD Image" on page 97

Note –

- **Starting with the Solaris 10 11/06 release**, you have the option during an initial installation to change the network security settings so that all network services, except Secure Shell, are disabled or restricted to respond to local requests only. This security option is available only during an initial installation, not during an upgrade. An upgrade maintains any previously set services. If necessary, you can restrict network services after an upgrade by using the `netservices` command. See "Planning Network Security" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

  The network services can be enabled after installation by using the `netservices open` command or by enabling individual services by using SMF commands. See "Revising Security Settings After Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.
- **Starting with the Solaris 10 10/08 release**, the structure of the Solaris DVD and Solaris Software - 1 CD has changed for the SPARC platform. Slice 0 is no longer at the top of the directory structure. Therefore the structure of the x86 and SPARC DVDs and Solaris Software - 1 CD are the same. This change in structure makes setting up an install server easier if you have a mix of platforms, such as a SPARC install server and x86 media.

Task Map: Installing From the Network With CD Media

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

Task	Description	For Instructions
(x86 only): Verify that your system supports PXE.	If you want to install an x86 based system over the network, confirm that your machine can use PXE to boot without local boot media. If your x86 based system does not support PXE, you must boot the system from a local DVD or CD.	Check your hardware manufacturer's documentation or the system BIOS.
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" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*

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

Task	Description	For Instructions
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 5, "Gathering Information Before Installation or Upgrade (Planning)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*
(Optional) Preconfigure system information.	You can preconfigure system information to avoid being prompted for the information during the installation or upgrade.	Chapter 2, "Preconfiguring System Configuration Information (Tasks)"
Create an install server.	Use the `setup_install_server(1M)` command to copy the Solaris Software - 1 CD to the install server's hard disk. Use the `add_to_install_server(1M)` command to copy additional Solaris Software CDs and the Solaris Languages CDs to the install server's hard disk.	"Creating an Install Server With SPARC or x86 CD Media" on page 86
(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 CD Image" on page 90
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 92
(Optional) Configure the DHCP server.	If you want to use DHCP to provide system configuration and installation parameters, configure the DHCP server, then create the appropriate options and macros for your installation. **Note –** If you want to install an x86 based system from the network with PXE, you must configure a DHCP server.	Chapter 13, "Planning for DHCP Service (Tasks)," in *System Administration Guide: IP Services* "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 43
Install the system over the network.	Begin the installation by booting the system from the network.	"Installing the System From the Network With a CD Image" on page 97

Creating an Install Server With SPARC or x86 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 an Install Server With SPARC or x86 CD 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*.

1 On the system that is to become the install server, become superuser or assume an equivalent role.

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

2 Insert the Solaris 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/Solaris_10/Tools
```

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 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 to the boot server.

  Using the `share` command, add 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 current Solaris release, 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 Software - 1 CD.**

9 **Insert the Solaris 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 Software - 2 CD.**

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

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

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

```
# cd /cdrom/cdrom0/Solaris_10/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 **Eject theCD.**

19 **Repeat Step 15 through Step 18 for the second Solaris Languages CD.**

20 Change directories to root (/).

```
# cd /
```

21 (Optional) Patch the files that are located in the miniroot on the net install image that was created by `setup_install_server`.

Patching a file might be necessary if a boot image has problems. For step-by-step procedures, see Chapter 7, "Patching the Miniroot Image (Tasks)."

22 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 92.**
- **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 90.**

Example 6–1 x86: Creating an Install Server With CD Media

The following example illustrates how to create an install server by copying the following CDs to the install server's `/export/home/cdimage` directory. This example assumes that the install server is running the current Solaris release.

- Solaris Software CDs
- Solaris Languages CDs

Insert the Solaris Software - 1 CD in the system's CD-ROM drive.

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

- If you have a separate boot server, add these steps.
  1. Make the install server available to the boot server.

     Using the `share` command, add this entry to the `/etc/dfs/dfstab` file.

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

  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 Software - 1 CD. Insert the Solaris Software - 2 CD in the CD-ROM drive.

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

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

Insert the first Solaris Languages CD in the CD-ROM drive.

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

Eject the CD.

Repeat the previous commands for each Solaris Languages 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 "Adding Systems to Be Installed From the Network With a CD Image" on page 92.

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 90.

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 92.
- 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

1 On the system you intend to make the boot server for the subnet, log in and become superuser or assume an equivalent role.

The system must include a local CD-ROM drive or have access to the remote current Solaris release disc images, which are normally on the install server. If you use a naming service, the system should be in the naming service. If you do not use a naming service, you must distribute information about this system by following your site's policies.

2 Mount the Solaris 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 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 6–2 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 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 92.

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 naming 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.

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)

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. See the procedure "To Create an Install Server," Step 6.

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

- 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 naming service to preconfigure system information
- The profile in the JumpStart directory on the profile server if you are using the custom JumpStart installation method

1. **On the install server or boot server, become superuser or assume an equivalent role.**

2. **If you use the NIS, NIS+, DNS, or LDAP naming service, verify that the following information about the system to be installed has been added to the naming service:**

   - Host name
   - IP address
   - Ethernet address

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

3. **Change to the `Tools` directory on the current Solaris release CD image on the install server:**

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

install_dir_path Specifies the path to the `Tools` directory

4 Add the client to the install server's `/etc/ethers` file.

a. On the client, find the ethers address. The `/etc/ethers` map is taken from the local file.

```
# ifconfig -a grep ether
ether 8:0:20:b3:39:1d
```

b. On the install server, open the `/etc/ethers` file in an editor. Add the address to the list.

5 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. The output of this option lists the DHCP options you need to create on the DHCP server.

For more information about class-specific installations by using DHCP, see "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 44.

`-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 current Solaris release 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 current Solaris release 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, including a boot file for that client.

The `nbp.` prefix is not used in boot file names. For example, if you specify `-e 00:07:e9:04:4a:bf` for an x86 based client, the command creates the boot file `010007E9044ABF.i86pc` in the `/tftpboot` directory. However, the current Solaris release supports the use of legacy boot files with the `nbp.` prefix.

For more information about client-specific installations by using DHCP, see "Creating DHCP Options and Macros for Solaris Installation Parameters" on page 44.

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" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

Example 6–3 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)" on page 43.

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

Example 6–4 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 6–5 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 current Solaris release 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 6–6 x86: Adding a Single x86 Install Client on an x86 Install Server When Using DHCP (CD)

The GRUB bootloader does not use the `SUNW.i86pc` DHCP class name. 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 `-e` option indicates that this installation will only occur on the client with the Ethernet address 00:07:e9:04:4a:bf .
- 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 Operating System for x86 Platforms DVD image in `/export/home/cdx86`.

```
x86_install_server# cd /export/boot/cdx86/Solaris_10/Tools
x86_install_server# ./add_install_client -d -e 00:07:e9:04:4a:bf \
-s rosemary:/export/home/cdx86 i86pc
```

The previous commands set up the client with the Ethernet address 00:07:e9:04:4a:bf as an installation client. The boot file `010007E9044ABF.i86pc` is created on the installation server. In previous releases, this boot file was named `nbp.010007E9044ABF.i86pc`.

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)" on page 43.

Example 6–7 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 `-b` option instructs the installation program to use the serial port `ttya` as an input and an output device.

Add the client.

```
install server# cd /export/boot/cdx86/Solaris_10/Tools
install server# ./add_install_client -d -e "00:07:e9:04:4a:bf" \
-b "console=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.

More Information

Continuing the Installation

If you are using a DHCP server to install the x86 based client over the network, configure the DHCP server and create the options and macros that are listed in the output of the `add_install_client -d` command. For instructions about how to configure a DHCP server to support network installations, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 43.

x86 based systems: If you are not using a DHCP server, you must boot the system from a local Solaris OS DVD or CD.

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

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.

- See "SPARC: To Install the Client Over the Network (CDs)" on page 97 for instructions about how to boot and install SPARC based systems over the network.
- See "x86: To Install the Client Over the Network With GRUB (CDs)" on page 100 for instructions about how to boot and install x86 based systems over the network.

▼ SPARC: To Install the Client Over the Network (CDs)

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 "SPARC: To Create an Install Server With SPARC or x86 CD Media" on page 86.
- 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 boot server, see "Creating a Boot Server on a Subnet With a CD Image" on page 90. 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 43.
- 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" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.
  - 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 18.
  - Set up a name server if you use a naming service to preconfigure system information. For information about how to preconfigure information with a naming service, see "Preconfiguring With the Naming Service" on page 39.
  - 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 3, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

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
  ```

- **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 2, "Preconfiguring System Configuration Information (Tasks)," for more information.
- If you did not preconfigure all the system information, use the "Checklist for Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*to help you answer the configuration questions.

Note – If the keyboard is self-identifying, the keyboard layout automatically configures during installation. If the keyboard is not self-identifying, you can select from a list of supported keyboard layouts during installation.

PS/2 keyboards are not self-identifying. You will be asked to select the keyboard layout during the installation.

For further information, see "`keyboard` Keyword" on page 23.

Note – During installation, you can choose the default NFSv4 domain name. Or, you can specify a custom NFSv4 domain name. For further information, see "NFSv4 Domain Name Configurable During Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

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

4 **If you are prompted, answer any additional questions to complete your installation.**

- **If you preconfigured all of the installation options, the installation program does not prompt you to enter any installation information. See Chapter 2, "Preconfiguring System Configuration Information (Tasks)," for more information.**
- **If you did not preconfigure all the installation options, use the "Checklist for Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade* to help you answer the installation questions.**

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 With GRUB" in *Solaris 10 10/08 Installation Guide: Basic Installations*.

▼ x86: To Install the Client Over the Network With GRUB (CDs)

The Solaris installation programs for x86 based systems use the GRUB boot loader. This procedure describes how to install an x86 based system over the network with the GRUB boot loader. For overview information about the GRUB boot loader, see Chapter 7, "SPARC and x86 Based Booting (Overview and Planning)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

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 "To Create an Install Server With SPARC or x86 DVD Media" on page 64.
- 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 boot server, see "Creating a Boot Server on a Subnet With a DVD Image" on page 67. 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 43.
- 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" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.
  - 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 18.
  - Set up a name server if you use a naming service to preconfigure system information. For information about how to preconfigure information with a naming service, see "Preconfiguring With the Naming Service" on page 39.
  - 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 3, "Preparing Custom JumpStart Installations (Tasks)," in *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

This procedure also assumes that your system can boot from the network.

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. The GRUB menu is displayed.

   Note – The GRUB menu that is displayed on your system might vary from the following sample, depending on the configuration of your network installation server.

```
GNU GRUB version 0.95 (631K lower / 2095488K upper memory)
+-------------------------------------------------------------------------------------------------------------------+
| Solaris 10 10 10/08 /cdrom0
|                                                                                                                   |
|                                                                                                                   |
+-------------------------------------------------------------------------------------------------------------------+
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, 'e' to edit the
commands before booting, or 'c' for a command-line.
```

5. **Select the appropriate installation option.**

   - **To install the Solaris OS from the network, select the appropriate Solaris entry on the menu, then press Enter.**

     Select this entry if you want to install from the network installation server you set up in "To Create an Install Server With SPARC or x86 DVD Media" on page 64.

   - **To install the Solaris OS from the network with specific boot arguments, follow these steps.**

     You might need to set specific boot arguments if you want to modify the device configuration during the installation, and did not set these boot arguments previously with

the `add_install_client` command as described in "To Add Systems to Be Installed From the Network With `add_install_client` (DVD)" on page 70.

a. On the GRUB menu, select the installation option you want to edit, then press e.

Boot commands that are similar to the following text are displayed in the GRUB menu.

```
kernel /I86pc.Solaris_10/multiboot kernel/unix \
-B install_media=192.168.2.1:/export/cdrom0/boot \
module /platform/i86pc/boot_archive
```

b. Use the arrow keys to select the boot entry that you want to edit, then press e.

The boot command that you want to edit is displayed in the GRUB edit window.

c. Edit the command by typing the boot arguments or options you want to use.

The command syntax for the Grub edit menu is as follows.

```
grub edit>kernel /image_directory/multiboot kernel/unix/ \
install [url|ask] -B options install_media=media_type
```

For information about boot arguments and command syntax, see Table 9–1.

d. To accept your edits and return to the GRUB menu, press Enter.

The GRUB menu is displayed. The edits you made to the boot command are displayed.

e. To begin the installation, type b in the GRUB menu.

The Solaris installation program checks the default boot disk for the requirements to install or upgrade the system. If the Solaris installation cannot detect the system configuration, the program prompts you for any missing information.

When the check is completed, the installation selection screen is displayed.

6 Select an installation type.

The installation selection screen displays the following options.

```
Select the type of installation you want to perform:

   1 Solaris Interactive
   2 Custom JumpStart
   3 Solaris Interactive Text (Desktop session)
   4 Solaris Interactive Text (Console session)
   5 Apply driver updates
   6 Single user shell

   Enter the number of your choice followed by the <ENTER> key.
   Alternatively, enter custom boot arguments directly.

   If you wait 30 seconds without typing anything,
```

```
an interactive installation will be started.
```

- **To install the Solaris OS, choose from the following options.**
  - **To install with the Solaris interactive installation GUI, type 1, then press Enter.**
  - **To install with the interactive text installer in a desktop session, type 3, then press Enter.**

    Select this installation type to override the default GUI installer and run the text installer.
  - **To install with the interactive text installer in a console session, type 4, then press Enter.**

    Select this installation type to override the default GUI installer and run the text installer.

  If you want to perform an unattended custom JumpStart installation (option 2), see *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

  For detailed information about the Solaris installation GUI and text installer, see "System Requirements and Recommendations" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

  The system configures the devices and interfaces, and searches for configuration files. The installation program begins. Go to Step 7 to continue the installation.

- **To perform system administration tasks before your installation, choose from the following options.**
  - **To update drivers or install an install time update (ITU), insert the update media, type 5, then press Enter.**

    You might need to update drivers or install an ITU to enable the Solaris OS to run on your system. Follow the instructions for your driver update or ITU to install the update.
  - **To perform system administration tasks, type 6, then press Enter.**

    You might want to launch a single user shell if you need to perform any system administration tasks on your system before you install. For information about system administration tasks you can perform prior to installation, see *System Administration Guide: Basic Administration*.

  After you perform these system administration tasks, the previous list of options is displayed. Select the appropriate option to continue the installation.

7 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 2, "Preconfiguring System Configuration Information (Tasks)," for more information.

- If you did not preconfigure all the system information, use the "Checklist for Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade* to help you answer the configuration questions.

Note – If the keyboard is self-identifying, the keyboard layout automatically configures during installation. If the keyboard is not self-identifying, you can select from a list of supported keyboard layouts during installation.

For further information, see "`keyboard` Keyword" on page 23.

Note – During installation, you can choose the default NFSv4 domain name. Or, you can specify a custom NFSv4 domain name. For further information, see "NFSv4 Domain Name Configurable During Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

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

8 If you are prompted, answer any additional questions to complete your installation.

- **If you preconfigured all of the installation options, the installation program does not prompt you to enter any installation information. See Chapter 2, "Preconfiguring System Configuration Information (Tasks)," for more information.**

- **If you did not preconfigure all the installation options, use the "Checklist for Installation" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade* to help you answer the installation questions.**

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

Note – When you boot the system after the installation, a GRUB menu lists the operating systems that are installed, including the newly-installed Solaris OS. Select which operating system you want to boot. The default selection loads if you do not make another selection.

More Information

Next Steps

If you install multiple operating systems on your machine, you need to instruct the GRUB boot loader to recognize these operating systems in order to boot. For more information, see "Modifying Boot Behavior by Editing the GRUB Menu at Boot Time" in *System Administration Guide: Basic Administration*.

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 With GRUB" in *Solaris 10 10/08 Installation Guide: Basic Installations*.

◆ ◆ ◆ CHAPTER 7

Patching the Miniroot Image (Tasks)

This chapter provides a step-by-step procedure and an example to patch the miniroot image when you are setting up an install server.

This chapter covers the following topics:

- "Patching the Miniroot Image (Tasks)" on page 107
- "Patching the Miniroot Image (Example)" on page 109

Patching the Miniroot Image (Tasks)

You might need to Patch the files that are located in the miniroot on the network installation image that was created by `setup_install_server`.

About the Miniroot Image (Overview)

The miniroot is a minimal, bootable root (`/`) file system that resides on the Solaris installation media. A miniroot consists of all the Solaris software that is required to boot the system to either install or upgrade the system. The miniroot software is used by the installation media to perform a full installation of the Solaris OS. The miniroot runs only during the installation process.

You might need to patch the miniroot before installation if the boot image has problems booting or if you need to add driver and hardware support. When you patch the miniroot image, the patch is not installed on the system where the Solaris OS installation occurs or on the system that the `patchadd` command is run. Patching the miniroot image is strictly used for adding driver and hardware support to the process that performs the actual installation of the Solaris OS.

Note – This procedure is only for patching the miniroot, not for patching the complete network installation image. If you need to patch the network installation image, perform the task after the installation completes.

▼ How To Patch the Miniroot Image

Follow these steps to patch a network installation miniroot image.

Note – These steps assume that you have a system on your network that is running the current Solaris release, and that system is accessible over the network.

1 On a system that is running the current Solaris release, log in as superuser or assume an equivalent role.

2 Change to the `Tools` directory of the installation image you created in Step 5.

```
# cd install-server-path/install-dir-path/Solaris_10/Tools
```

install-server-path Specifies the path to the install server system on your network, for example, `/net/installserver-1`.

3 Create a new installation image, and place that image on the system that is running the current Solaris release.

```
# ./setup_install_server remote_install_dir_path
```

remote_install_dir_path Specifies the path on the current Solaris release in which to create the new installation image.

This command creates a new installation image on the current Solaris release. In order to patch this image, you must temporarily place this image on a system that is running the current Solaris release.

4 On the current Solaris release, unpack the network installation boot archive.

```
# /boot/solaris/bin/root_archive unpackmedia remote_install_dir_path \
  destination_dir
```

remote_install_dir_path Specifies the path to the network installation image on the current Solaris release.

destination_dir Specifies the path to the directory to contain the unpacked boot archive.

5 On the current Solaris release, patch the unpacked boot archive.

```
# patchadd -C destination_dir path-to-patch/patch-id
```

path-to-patch Specifies the path to the patch that you want to add, for example, `/var/sadm/spool`.

patch-id Specifies the patch ID that you want to apply.

You can specify multiple patches with the `patchadd -M` option. For more information, see `patchadd(1M)`.

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

6 On the current Solaris release, pack the boot archive.

```
# /boot/solaris/bin/root_archive packmedia remote_install_dir_path \
  destination_dir
```

7 Copy the patched archives to the installation image on the install server.

```
# cd remote_install_dir_path
# find boot Solaris_10/Tools/Boot | cpio -pdum \
    install-server-path/install_dir_path
```

Next Steps After you have set up the install server and patched the miniroot, you might need to set up a boot server or add systems to be installed from the network.

- 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 69.
- 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 67.

Patching the Miniroot Image (Example)

This example describes the steps to patch a miniroot image to create a modified miniroot.

Patching the Miniroot Image

In this example, you perform the unpacking and packing of the miniroot on a system that is running the current release.

How to Modify the Miniroot (Example)

This procedure shows how to install a Kernel Update (KU) patch on a Solaris 10 10/08 miniroot image. On a system that is running the Solaris 10 OS, follow these steps, but note these details.

- jmp-start1 — Is a network installation server that is running the Solaris 9 OS
- v20z-1 — Is a system that is running the Solaris 10 OS, with GRUB implemented
- v20z-1:/export/mr — Is the unpacked miniroot location
- v20z-1:/export/u1 — Is the installation image that was created, so it could be modified

The network installation image is located at`/net/jmpstart1/export/images/solaris_10_u1/Solaris_10/Tools`.

1 On a system that is running the current Solaris release, log in as superuser or assume an equivalent role.

2 Change to the directory where you want to unpack the miniroot and place the network installation image.

```
# cd /net/server-1/export
```

3 Create the installation and miniroot directories.

```
# mkdir /export/u1 /export/mr
```

4 Change directories to the Tools directory where the Solaris 10 10/08 installation images are located.

```
# cd /net/jmp-start1/export/images/solaris_10/Solaris_10/Tools
```

5 Create a new installation image, and place the image on the system that is running the current Solaris release.

```
# ./setup_install_server /export/u1
Verifying target directory...
Calculating the required disk space for the Solaris_10 product
Calculating space required for the installation boot image
Copying the CD image to disk...
Copying Install Boot Image hierarchy...
Copying /boot  netboot hierarchy...
Install Server setup complete
```

The installation server setup is now complete.

6 Run the following command to unpack the miniroot.

```
 # /boot/solaris/bin/root_archive unpackmedia /export/u1 /export/mr
```

7 Change directories.

```
# cd  /export/mr/sbin
```

8 Make a copy of the `rc2` and the `sulogin` files.

```
# cp rc2 rc2.orig
# cp sulogin sulogin.orig
```

9 Apply all required patches to the miniroot.

`patchadd -C /export/mr /export` *patchid*

patchid specifies the patch ID that you want to apply.

In this example, five patches are applied to the miniroot.

```
# patchadd -C /export/mr /export/118344-14
# patchadd -C /export/mr /export/122035-05
# patchadd -C /export/mr /export/119043-10
# patchadd -C /export/mr /export/123840-04
# patchadd -C /export/mr /export/118855-36
```

10 Export the *SVCCFG_REPOSITORY* variable.

```
# export SVCCFG_REPOSITORY=/export/mr/etc/svc/repository.db
```

Caution – The *SVCCFG_REPOSITORY* variable must point to the location of the unpacked miniroot's `repository.db` file. In this example, that location is the `/export/mr/etc/svc` directory. The `repository.db` file is located in the directory `/etc/svc` under the unpacked miniroot. Failure to export this variable results in the modification of the live repository, which prevents the live system from booting.

11 Modify the miniroot's `repository.db` file.

```
# svccfg -s system/manifest-import setprop start/exec = :true
# svccfg -s system/filesystem/usr setprop start/exec = :true
# svccfg -s system/identity:node setprop start/exec = :true
# svccfg -s system/device/local setprop start/exec = :true
# svccfg -s network/loopback:default setprop start/exec = :true
# svccfg -s network/physical:default setprop start/exec = :true
# svccfg -s milestone/multi-user setprop start/exec = :true
```

For more information, see the `svccfg(1M)` man page.

12 Change directories. Then, restore the original copies of the `rc2.orig` and `sulogin.orig` files.

```
# cd /export/mr/sbin
# mv rc2.orig rc2
# mv sulogin.orig sulogin
```

13 Pack the modified miniroot that contains the changes you made. Place the modified miniroot in the `/export/u1` directory.

```
# /boot/solaris/bin/root_archive packmedia /export/u1 /export/mr
```

This step essentially replaces `/export/u1/boot/miniroot` directory, along with some other necessary files.

Next Steps After you have set up the install server and patched the miniroot, you might need to set up a boot server or add systems to be installed from the network.

- 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. You are finished. Proceed to "Adding Systems to Be Installed From the Network With a DVD Image" on page 69.
- 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 67.

◆ ◆ ◆ CHAPTER 8

Installing Over the Network (Examples)

This chapter provides examples that illustrate how to use DVD or CD media to install the Solaris OS over the network.

All examples in this chapter have the following conditions.

- The install server
  - Is a network installation image.
  - Runs the current Solaris release.
  - Is already part of the site's network and naming service.
- You have already gathered or preconfigured the information that you need in order to install. For further information, see Chapter 5, "Gathering Information Before Installation or Upgrade (Planning)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*.

Choose an example from one of the following additional options.

- "Network Installation Over the Same Subnet (Examples)" on page 114
  - The install client is on the same subnet as the install server. Therefore, you do not need to create a boot server.
  - The network installation uses a graphical user interface (GUI) in a desktop session.
- **Network Installation Over a Different Subnet (Examples TBD)**
  - The install client is on the different subnet from the install server. Therefore, you must create a boot server.
  - The network installation uses a text installer in a desktop session

Network Installation Over the Same Subnet (Examples)

This section includes the following examples.

- Example 8–1: **SPARC: Install on the Same Subnet (With DVD Media)**
- Example 8–2: **SPARC: Install on the Same Subnet (With CD Media)**
- Example 8–3: **x86: Install on the Same Subnet (With DVD Media)**
- Example 8–4: **x86: Install on the Same Subnet (With CD Media)**

EXAMPLE 8–1 SPARC: **Install Over the Same Subnet (With DVD Media)**

This example creates a SPARC install server with SPARC DVD media.

This example has the following conditions:

- The install client is on the same subnet as the install server.
- The network installation uses a graphical user interface (GUI) in a desktop session.
- General conditions for this example are listed at Chapter 8, "Installing Over the Network (Examples)."

1. **Create and set up a SPARC install server.**

   This example creates an install server by copying the Solaris DVD to the install server's `/export/home/dvdsparc` directory.

   a. Insert the Solaris DVD in the SPARC system's drive.

   b. Use the following command to create a directory to contain the DVD image. This command also changes to the `Tools` directory on the mounted disc. Then the command copies the DVD image in the drive to the install server's hard disk.

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

2. **Install the system with a network installation image.**

   In this example, you install with the Solaris interactive installation GUI.

   a. Boot the system from the network.

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

   ```
   ok bootnet - install
   ```

   The system installs from the network.

   c. 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.

   After you confirm the system configuration information, the Welcome to Solaris panel appears. The installation is complete.

EXAMPLE 8–1 SPARC: **Install Over the Same Subnet (With DVD Media)** *(Continued)*

For a more detailed explanation about the network installation procedures that are used in this example, see Chapter 5, "Installing From the Network With DVD Media (Tasks)."

EXAMPLE 8–2 SPARC: **Install Over the Same Subnet (With CD Media)**

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

This example has the following conditions:

- The install client is on the same subnet as the install server.
- The network installation uses a graphical user interface (GUI) in a desktop session.
- General conditions for this example are listed at Chapter 8, "Installing Over the Network (Examples)."

1. **Create and set up a SPARC install Server.**

   The following example illustrates how to create an install server by copying the CD media to the install server's `/export/home/cdsparc` directory.

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

   b. Use the following command to create a directory for the CD image. This command also changes to the `Tools` directory on the mounted disc, and copies the image in the drive to the install server's hard disk.

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

2. **Add systems to be installed from the network.**

   a. Insert the Solaris Software for SPARC Platforms - 2 CD in the CD-ROM drive.

   b. Use the following command. This command changes to the `Tools` directory on the mounted CD. The command copies the CD in the CD-ROM drive to the install server's hard disk. Then the command changes to the root (`/`) directory.

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

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

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

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

EXAMPLE 8–2 SPARC: **Install Over the Same Subnet (With CD Media)** *(Continued)*

e. Eject the CD.

f. Repeat the previous commands for each Solaris Languages for SPARC Platforms CD CD that you want to install.

3. **Install the system with a network installation image.**

   a. Boot the system from the network.

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

      ```
      ok boot net
      ```

      The system installs from the network.

   c. If you are prompted, answer the system configuration questions.

      After you confirm the system configuration information, the Welcome to Solaris panel appears. The installation is complete.

For a more detailed explanation about the network installation procedures that are used in this example, see Chapter 6, "Installing From the Network With CD Media (Tasks)."

EXAMPLE 8–3 x86: **Install Over the Same Subnet (With DVD Media)**

This example creates an x86 install server with x86 DVD media.

This example has the following conditions:

- The install client is on the same subnet as the install server.
- The network installation uses a graphical user interface (GUI) in a desktop session.
- General conditions for this example are listed at Chapter 8, "Installing Over the Network (Examples)."

1. **Create and set up an x86 install server.**

   The following examples illustrate how to create an x86 install server by copying the Solaris Operating System for x86 Platforms DVD to the install server's `/export/home/dvdx86` directory.

   a. Insert the Solaris DVD into the system's drive.

   b. Use the following command. This command creates a directory to contain the boot image. Then this command changes to the `Tools` directory on the mounted disc. Also, the command copies the disc in the drive to the install server's hard disk by using the `setup_install_server` command:

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

EXAMPLE 8–3 x86: **Install Over the Same Subnet (With DVD Media)** *(Continued)*

c. Make the install server available to the boot server.

Using the `share` command, add this entry to the `/etc/dfs/dfstab` file.

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

d. 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 /
```

Note – If the install server was running the Solaris 9 OS, or compatible version, you would type the following command instead.

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

For this older release, if the `nfsd` daemon was running, you would continue to the next step. If the `nfsd` daemon was not running, you would start it.

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

---

2. **Add systems to be installed from the network.**

The file system `/export/home/dvdx86/` contains the `add_install_client` command. The install client is named `basil`, which is an x86 system.

a. Add the client to the install server's `/etc/ethers` file.

On the client, find the ethers address. The `/etc/ethers` map is taken from the local file.

```
# ifconfig -a grep ether
ether 8:0:20:b3:39:1d
```

On the install server, open the `/etc/ethers` file in an editor. Add the address to the list.

b. Use the following command. This command changes to the `Tools` directory on the Solaris DVD image. Then, this command sets up the client system so that it can be installed from the network.

```
install_server# cd /export/home/dvdx86/Solaris_10/Tools
install_server# ./add_install_client basil i86pc
```

3. **Install the system with a network installation image.**

The Solaris installation programs for x86 based systems use the GRUB boot loader. This example installs an x86 based system over the network with the GRUB boot loader.

EXAMPLE 8–3 x86: **Install Over the Same Subnet (With DVD Media)** *(Continued)*

a. In the system BIOS, instruct the system to boot from the network.

   After you exit BIOS, the system installs from the network. The GRUB menu is displayed.

b. To install the Solaris OS from the network, select the appropriate Solaris entry on the menu, then press Enter.

   The installation selection screen is displayed.

c. To install with the Solaris interactive installation GUI, type 1, then press Enter.

   The installation program begins.

d. If you are prompted, answer the system configuration questions.

   After you confirm the system configuration information, the Welcome to Solaris panel appears.

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

Note – When you boot the system after the installation, a GRUB menu lists the operating systems that are installed, including the newly-installed Solaris OS. Select which operating system you want to boot. The default selection loads if you do not make another selection.

For further information, see the following references.

Procedure	Reference
For a more detailed explanation about the network installation procedures that are used in this example	Chapter 5, "Installing From the Network With DVD Media (Tasks)"
For information about how to complete an interactive installation with the Solaris installation GUI	"To Install or Upgrade With the Solaris Installation Program With GRUB" in *Solaris 10 10/08 Installation Guide: Basic Installations*
For overview information about the GRUB boot loader	Chapter 7, "SPARC and x86 Based Booting (Overview and Planning)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*

EXAMPLE 8–4 x86: **Install Over the Same Subnet (With CD Media)**

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

This example has the following conditions:

- The install client is on the same subnet as the install server.
- The network installation uses a graphical user interface (GUI) in a desktop session.
- General conditions for this example are listed at Chapter 8, "Installing Over the Network (Examples)."

1. **Create and set up an x86 install server.**

   The following steps create an install server by copying the following CDs to the install server's `/export/home/cdx86` directory.

   a. Insert the Solaris Software - 1 CD in the system's drive.

   b. Use the following command. This command creates a directory for the CD image and changes to the `Tools` directory on the mounted disc. This command then copies the image in the drive to the install server's hard disk.

   ```
   # mkdir -p /export/home/dvdx86
   # cd /cdrom/cdrom0/Solaris_10/Tools
   # ./setup_install_server /export/home/cdx86
   ```

   c. Insert the Solaris Software - 2 CD in the system's CD-ROM drive.

   d. Use the following command. This command changes to the `Tools` directory on the mounted CD. Then this command copies the CD in the CD-ROM drive to the install server's hard disk and changes to the root (`/`) directory.

   ```
   # cd /cdrom/cdrom0/Solaris_10/Tools
   # ./add_to_install_server /export/home/cdx86
   # cd /
   ```

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

   f. Insert the first Solaris Languages CD in the system's CD-ROM drive.

   g. Use the following command. This command changes to the `Tools` directory on the mounted CD. This command then copies the CD in the CD-ROM drive to the install server's hard disk.

   ```
   # cd /cdrom/cdrom0/Solaris_10/Tools
   # ./add_to_install_server /export/home/cdx86
   ```

   h. Eject the CD.

   i. Repeat the previous commands for each Solaris Languages for SPARC Platforms CD CD that you want to install.

2. **Add systems to be installed from the network.**

EXAMPLE 8–4 x86: **Install Over the Same Subnet (With CD Media)** *(Continued)*

In this example, the install client is named `basil`, which is an x86 system. The file system `/export/home/cdx86/Solaris_10/Tools` contains the `add_install_client` command.

a. Add the client to the install server's `/etc/ethers` file. On the client, find the ethers address. The `/etc/ethers` map is taken from the local file.

```
# ifconfig -a grep ether
ether 8:0:20:b3:39:1d
```

b. On the install server, open the `/etc/ethers` file in an editor. Add the address to the list.

c. Use the following command. This command changes to the `Tools` directory on the current Solaris release CD image on the install server. Then this command adds the client system to be installed from the network.

```
install_server# cd /export/home/cdx86/Solaris_10/Tools
install_server# ./add_install_client basil i86pc
```

3. **Install the system with a network installation image.**

This step describes how to install an x86 based system over the network with the GRUB boot loader.

a. In the system BIOS, instruct the system to boot from the network.

After you exit BIOS, the system installs from the network. The GRUB menu is displayed.

b. To install the Solaris OS from the network, select the appropriate Solaris entry on the menu, then press Enter.

The installation selection screen is displayed.

c. To install with the Solaris interactive installation GUI, type 1, then press Enter.

The installation program begins.

d. If you are prompted, answer the system configuration questions.

After you confirm the system configuration information, the Welcome to Solaris panel appears.

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

Note – When you boot the system after the installation, a GRUB menu lists the operating systems that are installed, including the newly-installed Solaris OS. Select which operating system you want to boot. The default selection loads if you do not make another selection.

For further information, see the following references.

EXAMPLE 8–4 x86: **Install Over the Same Subnet (With CD Media)** *(Continued)*

Procedure	Reference
For a more detailed explanation about the network installation procedures that are used in this example	Chapter 6, "Installing From the Network With CD Media (Tasks)"
For information about how to complete an interactive installation with the Solaris installation GUI	"To Install or Upgrade With the Solaris Installation Program With GRUB" in *Solaris 10 10/08 Installation Guide: Basic Installations*
For overview information about the GRUB boot loader	Chapter 7, "SPARC and x86 Based Booting (Overview and Planning)," in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade*

◆ ◆ ◆ CHAPTER 9

Installing From the Network (Command Reference)

This chapter lists the commands used to set up network installations. This chapter includes the following topics.

- "Network Installation Commands" on page 123
- "x86: GRUB Menu Commands for Installation" on page 124

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 current Solaris release 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 DVD or Solaris Software and Solaris Languages CDs. The `mount`(1M) man page contains more information.
`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.

Command	Platform	Description
`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. **Caution –** Don't use the `patchadd -C` command unless you have read the `Patch README` instructions or have contacted your local Sun support office. For more information, see the following references: ■ Chapter 7, "Patching the Miniroot Image (Tasks)" ■ The `patchadd(1M)` man page contains more information.
`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.

x86: GRUB Menu Commands for Installation

You can customize the network boot and installation of your system by editing the commands in the GRUB menu. This section describes several commands and arguments you can insert in the commands in the GRUB menu.

In the GRUB menu, you can access the GRUB command line by typing b at the prompt. A command line that is similar to the following output is displayed.

```
kernel /Solaris_10_x86/multiboot kernel/unix
-B install_media=192.168.2.1:/export/cdrom0/boot
module /platform/i86pc/boot_archive
```

You can edit this command line to customize your boot and installation. The following list describes several common commands you might want to use. For a complete list of boot arguments that you can use with the `-B` option, see the `eeprom(1M)` man page.

Note – To add multiple arguments with the `-B` option, separate the arguments with a comma.

TABLE 9–1 x86: GRUB Menu Commands and Options

Command/Option	Description and Examples

`install`	Insert this option before the `-B` option to perform a custom JumpStart installation.
	`kernel /Solaris_10_x86/multiboot` **`install`**
	`-B install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`

TABLE 9–1 x86: GRUB Menu Commands and Options *(Continued)*

Command/Option	Description and Examples

url|`ask`	Specifies the location of the custom JumpStart files or prompts you for the location. Insert either option with the `install` option.
	■ *url* - Specifies the path to the files. You can specify a URL for files that are located in the following places:
	■ Local hard disk
	`file://`*jumpstart_dir_path/compressed_config_file*
	`For example:`
	`kernel /Solaris_10_x86/multiboot` **`install`**
	`file://jumpstart/config.tar`
	`-B install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`
	■ NFS server
	`nfs://`*server_name:IP_address/jumpstart_dir_path/compressed_config_file*
	`For example:`
	`kernel /Solaris_10_x86/multiboot` **`install`**
	`myserver:192.168.2.1/jumpstart/config.tar`
	`-B install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`
	■ HTTP server
	`http://`*server_name:IP_address/jumpstart_dir_path/compressed_config_file&proxy_info*
	■ If you placed a `sysidcfg` file in the compressed configuration file, you must specify the IP address of the server that contains the file, as in the following example:
	`kernel /Solaris_10_x86/multiboot` **`install`**
	`http://192.168.2.1/jumpstart/config.tar`
	`-B install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`
	■ If you saved the compressed configuration file on an HTTP server that is behind a firewall, you must use a proxy specifier during boot. You do not need to specify an IP address for the server that contains the file. You must specify an IP address for the proxy server, as in the following example:
	`kernel /Solaris_10_x86/multiboot` **`install`**
	`http://www.shadow.com/jumpstart/config.tar&proxy=131.141.6.151`
	`-B install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`

TABLE 9–1 x86: GRUB Menu Commands and Options *(Continued)*

Command/Option	Description and Examples

url|`ask` *(continued)*	■ `ask` - When used with the `install` option, specifies that the installation program prompt you to type the location of the compressed configuration file after the system boots and connects to the network. If you use this option, you are not able to do a completely hands off JumpStart installation.
	If you bypass the prompt by pressing Return, the Solaris installation program interactively configures the network parameters. The installation program then prompts you for the location of the compressed configuration file.
	The following example performs a custom JumpStart and boots from a network installation image. You are prompted to input the location of the configuration file after the system connects to the network.
	`kernel /Solaris_10_x86/multiboot` **`install ask`**
	`-B install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`

`dhcp`	Insert this option before the `-B` option to instruct the installation programs to use a DHCP server to obtain network installation information that is needed to boot the system. If you do not specify to use a DHCP server by typing `dhcp`, the system uses the `/etc/bootparams` file or the naming service `bootparams` database. For example, you would not specify `dhcp` if you wanted keep a static IP address.
	`kernel /Solaris_10_x86/multiboot` **`dhcp`**
	`-B install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`

`- text`	Insert this option before the `-B` option to perform a text-based installation in a desktop session.
	`kernel /Solaris_10_x86/multiboot` **`- text`**
	`-B install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`

`- nowin`	Insert this option before the `-B` option to perform a text-based installation in a console session.
	`kernel /Solaris_10_x86/multiboot` **`- nowin`**
	`-B install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`

`console=`*serial-console*	Use this argument with the `-B` option to instruct the system to use a serial console, such as `ttya` (COM1) or `ttyb` (COM2).
	`kernel /Solaris_10_x86/multiboot -B` **`console=ttya`**
	`install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`

TABLE 9–1 x86: GRUB Menu Commands and Options *(Continued)*

Command/Option	Description and Examples

`ata-dma-enabled=[0|1]`	Use this argument with the `-B` option to enable or disable Advanced Technology Attachment (ATA) or Integrated Drive Electronics (IDE) devices and Direct Memory Access (DMA) during the installation.
	`kernel /Solaris_10_x86/multiboot -B` **`ata-dma-enabled=0`**
	`install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`

`acpi-enum=[0|1]`	Use this argument with the `-B` option to enable or disable Advanced Configuration and Power Interface (ACPI) power management.
	`kernel /Solaris_10_x86/multiboot -B` **`acpi-enum=0`**
	`install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`

`atapi-cd-dma-enabled=[0|1]`	Use this argument with the `-B` option to enable or disable DMA for CD or DVD drives during the installation.
	`kernel /Solaris_10_x86/multiboot -B` **`atapi-cd-dma-enabled=0`**
	`install_media=192.168.2.1:/export/cdrom0/boot`
	`module /platform/i86pc/boot_archive`
	Note – The DMA name *atapi* is the current variable name used for DMA. This variable is subject to change.

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 10

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 131
- "When to Use WAN Boot" on page 132
- "How WAN Boot Works (Overview)" on page 133
- "Security Configurations Supported by WAN Boot (Overview)" on page 137

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.

Note – The `flarcreate` command no longer has size limitations on individual files. You can create a Solaris Flash archive that contains individual files over 4 Gbytes.

For more information, see "Creating an Archive That Contains Large Files" in *Solaris 10 10/08 Installation Guide: Solaris Flash Archives (Creation and Installation)*.

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 133.

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 4, "Installing From the Network (Overview)."

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 10–1 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.

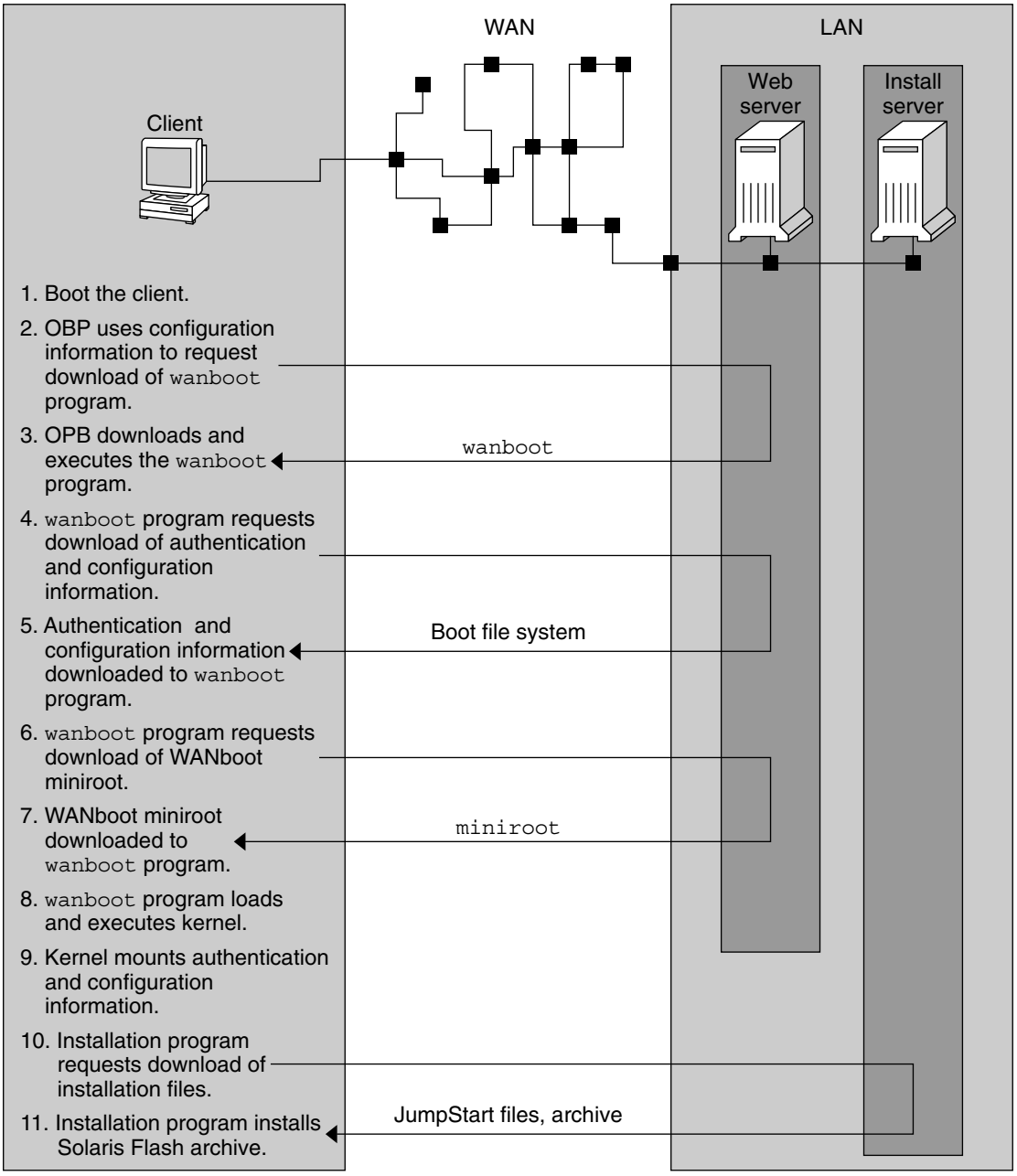

FIGURE 10–1 Sequence of Events in a WAN Boot Installation

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 170.

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 170.

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 147. 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 168.

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 137. For less critical systems, or systems on semi-private networks, consider the configuration that is described in "Insecure WAN Boot Installation Configuration" on page 138.

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 12–1.

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 12–2.

◆ ◆ ◆ CHAPTER 11

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 139
- "WAN Boot Security Limitations" on page 148
- "Gathering Information for WAN Boot Installations" on page 148

WAN Boot Requirements and Guidelines

The section describes the system requirements to perform a WAN boot installation.

TABLE 11–1 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 11–1 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 512 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 159.
(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 187. 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 14, "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 11–1 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.

  Note – The `flarcreate` command no longer has size limitations on individual files. You can create a Solaris Flash archive that contains individual files over 4 Gbytes.

  For more information, see "Creating an Archive That Contains Large Files" in *Solaris 10 10/08 Installation Guide: Solaris Flash Archives (Creation and Installation)*.

- 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 11–1 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.

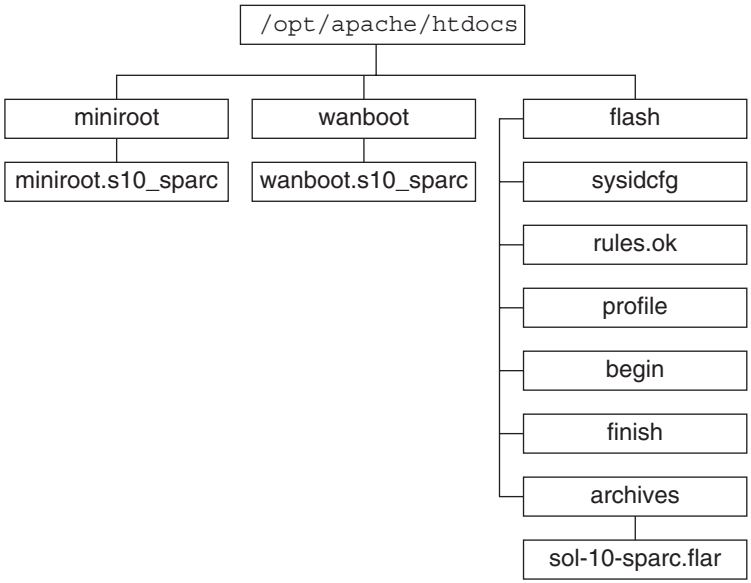

FIGURE 11–1 Sample Structure for Document Root Directory

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 current Solaris release 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 173.

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 144 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 181
- "To Create the `wanboot.conf` File" on page 183
- "(Optional) To Create a Hashing Key and an Encryption Key" on page 170
- "(Optional) To Use Digital Certificates for Server and Client Authentication" on page 168

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 11–2 demonstrates how you can set up the `/etc/netboot` directory to customize your WAN boot installations.

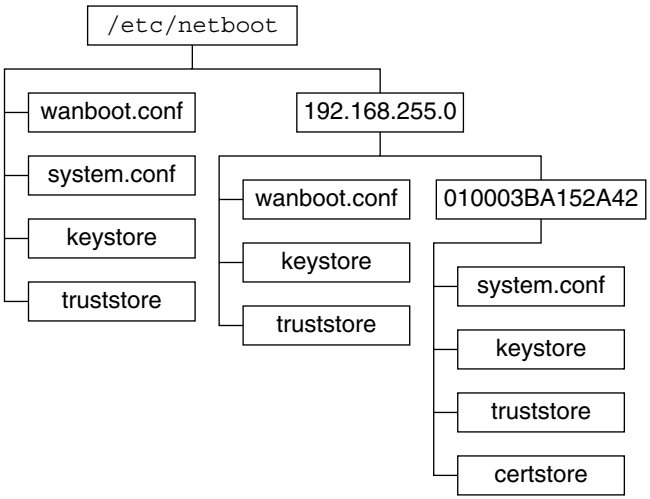

FIGURE 11–2 Sample `/etc/netboot` Directory

The `/etc/netboot` directory layout in Figure 11–2 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 168.

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 naming service** – If you use a naming 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 11–2
- Table 11–3

TABLE 11–2 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	

TABLE 11–2 Worksheet for Collecting Server Information *(Continued)*

Information Needed	Notes
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 11–3 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 12

Installing With WAN Boot (Tasks)

This chapter describes the following tasks that are necessary to prepare your network for a WAN boot installation.

- "Installing Over a Wide Area Network (Task Maps)" on page 151
- "Configuring the WAN Boot Server" on page 155
- "Creating the Custom JumpStart Installation Files" on page 173
- "Creating the Configuration Files" on page 181
- "(Optional) Providing Configuration Information With a DHCP Server" on page 187
- "(Optional) To Configure the WAN Boot Logging Server" on page 166

Installing 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 12–1.

  For a description of a secure WAN boot installation over HTTPS, see "Secure WAN Boot Installation Configuration" on page 137.
- For a list of the tasks you need to perform to prepare for an insecure WAN boot installation, see Table 12–2.

  For a description of an insecure WAN boot installation, see "Insecure WAN Boot Installation Configuration" on page 138.

To use a DHCP server or a logging server, complete the optional tasks that are listed at the bottom of each table.

TABLE 12–1 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 135 "Security Configurations Supported by WAN Boot (Overview)" on page 137
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 148
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 155
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 156
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 159
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 160
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 165
(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 166
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 162
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 167
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 168

TABLE 12–1 Task Map: Preparing to Perform a Secure WAN Boot Installation *(Continued)*

Task	Description	For Instructions
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 170
Create the Solaris Flash archive.	Use the `flarcreate` command to create an archive of the software that you want to install on the client.	"To Create the Solaris Flash Archive" on page 173
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 175 "To Create the Profile" on page 176 "To Create the `rules` File" on page 178 "(Optional) Creating Begin and Finish Scripts" on page 180
Create the system configuration file.	Set the configuration information in the `system.conf` file.	"To Create the System Configuration File" on page 181
Create the WAN boot configuration file.	Set the configuration information in the `wanboot.conf` file.	"To Create the `wanboot.conf` File" on page 183
(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 43

TABLE 12–2 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 135 "Security Configurations Supported by WAN Boot (Overview)" on page 137
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 148
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 155

TABLE 12–2 Task Map: Preparing to Perform an Insecure WAN Boot Installation *(Continued)*

Task	Description	For Instructions
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 156
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 159
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 160
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 165
(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 166
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 162
(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 170
Create the Solaris Flash archive.	Use the `flarcreate` command to create an archive of the software that you want to install on the client.	"To Create the Solaris Flash Archive" on page 173
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 175 "To Create the Profile" on page 176 "To Create the `rules` File" on page 178 "(Optional) Creating Begin and Finish Scripts" on page 180
Create the system configuration file.	Set the configuration information in the `system.conf` file.	"To Create the System Configuration File" on page 181

TABLE 12–2 Task Map: Preparing to Perform an Insecure WAN Boot Installation *(Continued)*

Task	Description	For Instructions
Create the WAN boot configuration file.	Set the configuration information in the `wanboot.conf` file.	"To Create the `wanboot.conf` File" on page 183
(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 43

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 11–1.

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 155
- "Creating the WAN Boot Miniroot" on page 156
- "Installing the `wanboot` Program on the WAN Boot Server" on page 160
- "Creating the `/etc/netboot` Hierarchy on the WAN Boot Server" on page 162
- "Copying the WAN Boot CGI Program to the WAN Boot Server" on page 165
- "(Optional) Protecting Data by Using HTTPS" on page 167

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 142.

For an example of how to set up this directory, see "Create the Document Root Directory" on page 213.

After you create the document root directory, create the WAN boot miniroot. For instructions, see "Creating the WAN Boot Miniroot" on page 156.

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 DVD or the Solaris 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*.

1 Become superuser or assume an equivalent role 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 naming service

  If you use a naming service, the system must already be in a naming service, such as NIS, NIS+, DNS, or LDAP. If you do not use a naming service, you must distribute information about this system by following your site's policies.

2 Insert the Solaris Software - 1 CD or the Solaris 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/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 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 12–1 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 Software media in the media drive that is attached to `wanserver-1`. Type the following commands.

```
wanserver-1# mkdir -p /export/install/cdrom0
wanserver-1# cd /cdrom/cdrom0/Solaris_10/Tools
wanserver-1# ./setup_install_server -w /export/install/cdrom0/miniroot \
/export/install/cdrom0
```

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/cdrom0/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 158.

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.

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. For those clients who do have an OBP that is capable of supporting WAN boot installations, see your system documentation for information about how to update the OBP.

    Note – Not all client OBPs support WAN Boot. For those clients use the next option.

  - After you complete the preparation tasks and are ready to install the client, perform the WAN boot installation from the Solaris Software CD1 or DVD. This option works in all cases when the current OBP does not provide WAN Boot support.

    For instructions about how to boot the client from CD1, see "To Perform a WAN Boot Installation With Local CD Media" on page 206. To continue preparing for the WAN boot installation, see "Creating the `/etc/netboot` Hierarchy on the WAN Boot Server" on page 162.

Example 12–2 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 160.

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 162.

See Also For additional information about the `setup_install_server` command, see Chapter 4, "Installing From the Network (Overview)."

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 159.
- 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 162 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*.

Before You Begin Verify that your client system supports WAN boot. See "To Check the Client OBP for WAN Boot Support" on page 159 for more information.

1 Become superuser or assume an equivalent role on the install server.

2 Insert the Solaris Software - 1 CD or the Solaris DVD in the install server's drive.

3 Change to the `sun4u` platform directory on the Solaris Software - 1 CD or the Solaris DVD.

```
# cd /cdrom/cdrom0/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.s10_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 12–3 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 Software media to the WAN boot server's document root directory.

Insert the Solaris DVD or the Solaris Software - 1 CD in the media drive that is attached to `wanserver-1` and type the following commands.

```
wanserver-1# cd /cdrom/cdrom0/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 162.

See Also For overview information about the `wanboot` program, see "What Is WAN Boot?" on page 131.

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 144.

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.

1. **Become superuser or assume an equivalent role 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 12–4 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 165.

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 144.

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 current Solaris release 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

1. **Become superuser or assume an equivalent role 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 166.

If you do not want to set up a separate logging server, see "(Optional) Protecting Data by Using HTTPS" on page 167 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 131.

▼ (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.

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 183.

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 12–5 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 167 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 137, 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 173.

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 168.
- 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 168.
- 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 170.
- (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 147.

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 173.

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 144.
- For instructions about how to create the `/etc/netboot` hierarchy, see "Creating the `/etc/netboot` Hierarchy on the WAN Boot Server" on page 162.

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 170.**
- **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 12–6 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 170.

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 173.

To create a hashing key and an encryption key, follow these 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 192.

Example 12–7 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 173.

See Also For overview information on hashing keys and encryption keys, see "Protecting Data During a WAN Boot Installation" on page 135.

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 173
- "To Create the `sysidcfg` File" on page 175
- "To Create the `rules` File" on page 178
- "To Create the Profile" on page 176
- "(Optional) Creating Begin and Finish Scripts" on page 180

For detailed information on the custom JumpStart installation method, see Chapter 2, "Custom JumpStart (Overview)," in *Solaris 10 10/08 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 10/08 Installation Guide: Solaris Flash Archives (Creation and Installation)*.
  - For detailed information about Solaris Flash archives, see Chapter 1, "Solaris Flash (Overview)," in *Solaris 10 10/08 Installation Guide: Solaris Flash Archives (Creation and Installation)*.
- File Size Issues:

  Check your web server software documentation to verify that the software can transmit files that are the size of a Solaris Flash archive.

- Check your web server software documentation to verify that the software can transmit files that are the size of a Solaris Flash archive.
- The `flarcreate` command no longer has size limitations on individual files. You can create a Solaris Flash archive that contains individual files over 4 Gbytes.

  For more information, see "Creating an Archive That Contains Large Files" in *Solaris 10 10/08 Installation Guide: Solaris Flash Archives (Creation and Installation)*.

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 `flarcreate` command.

`# flarcreate -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 `flarcreate` command to customize your Solaris Flash archive. For detailed descriptions of these options, see Chapter 5, "Solaris Flash (Reference)," in *Solaris 10 10/08 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 `flarcreate` 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 `flarcreate` command returns an exit code of 0.
- If the archive creation fails, the `flarcreate` command returns a nonzero exit code.

Example 12–8 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# flarcreate -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 175.

See Also

For detailed instructions about how to create a Solaris Flash archive, see Chapter 3, "Creating Solaris Flash Archives (Tasks)," in *Solaris 10 10/08 Installation Guide: Solaris Flash Archives (Creation and Installation)*.

For more information about the `flarcreate` 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 173 for detailed instructions.

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 22.

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 12–9

`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 naming 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 176.

See Also For more detailed information about `sysidcfg` keywords and values, see "Preconfiguring With the `sysidcfg` File" on page 18.

▼ 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 10/08 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 175 for detailed instructions.

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 10/08 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 10/08 Installation Guide: Custom JumpStart and Advanced Installations* contains information about testing profiles.

Example 12–10 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 178.

See Also

For more information about how to create a profile, see "Creating a Profile" in *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

For more detailed information about profile keywords and values, see "Profile Keywords and Values" in *Solaris 10 10/08 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 10/08 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 176 for detailed instructions.

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 10/08 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 current Solaris release 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 DVD or a Solaris 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 12–11 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 current Solaris release 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 180.

If you do not want to set up begin and finish scripts, see "Creating the Configuration Files" on page 181 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 10/08 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 10/08 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 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.
- For more information about creating finish scripts, see "Creating Finish Scripts" in *Solaris 10 10/08 Installation Guide: Custom JumpStart and Advanced Installations*.

To continue preparing for your WAN boot installation, see "Creating the Configuration Files" on page 181.

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 173 for detailed instructions.

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 175.

   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 176 and "To Create the `rules` File" on page 178.

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 12–12 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 12–13 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 183.

▼ 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 162.

If the WAN boot server is running the current Solaris release, 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
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 229.

To create the `wanboot.conf` file, follow these 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 229.

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 162.

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 12–14 `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 166.

`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 12–15 `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 166.

`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 187.

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 190 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 229 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 43.

For detailed information about setting up a Solaris DHCP server, see Chapter 14, "Configuring the DHCP Service (Tasks)," in *System Administration Guide: IP Services*.

To continue with your WAN boot installation, see Chapter 13, "SPARC: Installing With WAN Boot (Tasks)."

◆ ◆ ◆ CHAPTER 13

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 12, "Installing With WAN Boot (Tasks)."

This chapter describes the following tasks.

- "Preparing the Client for a WAN Boot Installation" on page 190
- "Installing the Client" on page 197

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 13–1 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 12, "Installing With WAN Boot (Tasks)"
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 190

TABLE 13–1 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 192
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 198 "To Perform an Interactive WAN Boot Installation" on page 200 "To Perform a WAN Boot Installation With a DHCP Server" on page 204 "To Perform a WAN Boot Installation With Local CD Media" on page 206

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 190
- "Installing Keys on the Client" on page 192

▼ 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.

1 Become superuser or assume an equivalent role 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 192 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 13–1 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 192.
- If you are performing a less secure installation without keys, see "Installing the Client" on page 197.

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 193.
- For instructions about how to install keys during the boot process, see "To Perform an Interactive WAN Boot Installation" on page 200.

- 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 195.

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.

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 197 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 13–2 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 197.

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`).

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 or assume an equivalent role 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 197 for instructions about how to install the client system.

Example 13–3 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 197.

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 13–2 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 192. ■ To perform a noninteractive installation, see "To Perform a Noninteractive WAN Boot Installation" on page 198.

TABLE 13–2 Methods to Install the Client *(Continued)*

Method	Description	Instructions
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 200
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 187. ■ To use a DHCP server during your installation, see "To Perform a WAN Boot Installation With a DHCP Server" on page 204.
Installing with local CD media	If your client OBP does not support WAN boot, boot the client from a local copy of the Solaris Software CD.	■ To determine if the client OBP supports WAN boot, see "To Check the Client OBP for WAN Boot Support" on page 159. ■ To install the client with a local copy of the Solaris Software CD, see "To Perform a WAN Boot Installation With Local CD Media" on page 206.

▼ 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 192.

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 13–4 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 192.

```
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, 512 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.

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 13–5 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, 512 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 187.

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 192.
- 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 43.

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 13–6 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, 512 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 187.

▼ 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 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.

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 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 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 the`bootserver` 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 13–7 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, 512 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 14

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" on page 212
- "Create the Document Root Directory" on page 213
- "Create the WAN Boot Miniroot" on page 213
- "Check the Client OBP for WAN Boot Support" on page 213
- "Install the `wanboot` Program on the WAN Boot Server" on page 214
- "Create the `/etc/netboot` Hierarchy" on page 214
- "Copy the `wanboot-cgi` Program to the WAN Boot Server" on page 215
- "(Optional) Configure the WAN Boot Server as a Logging Server" on page 215
- "Configure the WAN Boot Server to Use HTTPS" on page 215
- "Provide the Trusted Certificate to the Client" on page 215
- "(Optional) Use Private Key and Certificate for Client Authentication" on page 216
- "Create the Keys for the Server and the Client" on page 217
- "Create the Solaris Flash Archive" on page 217
- "Create the `sysidcfg` File" on page 218
- "Create the Client's Profile" on page 218
- "Create and Validate the `rules` File" on page 219
- "Create the System Configuration File" on page 219
- "Create the `wanboot.conf` File" on page 220
- "Check the `net` Device Alias in OBP" on page 221
- "Install Keys on the Client" on page 222
- "Install the Client" on page 223

Sample Site Setup

Figure 14–1 shows the site setup for this example.

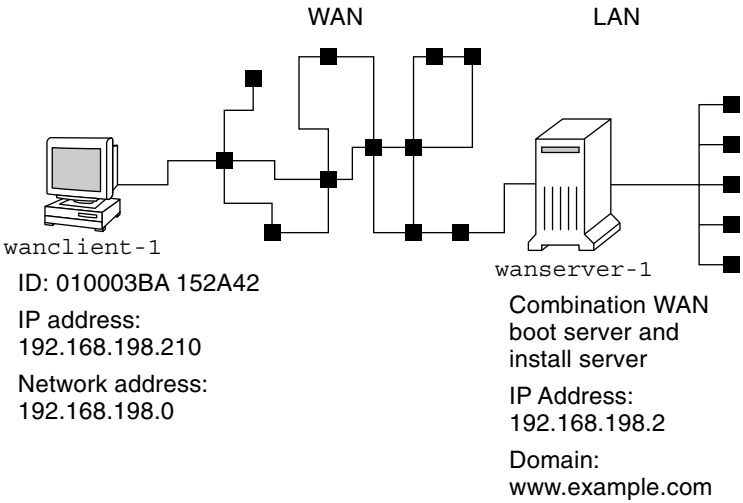

FIGURE 14–1 Sample Site for WAN Boot Installation

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 current Solaris release.
- `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 current Solaris release 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 Software media in the media drive that is attached to `wanserver-1`. Type the following commands.

```
wanserver-1# mkdir -p /export/install/cdrom0
wanserver-1# cd /cdrom/cdrom0/Solaris_10/Tools
wanserver-1# ./setup_install_server -w /export/install/cdrom0/miniroot \
/export/install/cdrom0
```

Move the WAN boot miniroot to the document root directory (`/opt/apache/htdocs/`) of the WAN boot server.

```
wanserver-1# mv /export/install/cdrom0/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 Software media to the WAN boot server's document root directory.

Insert the Solaris DVD or the Solaris Software - 1 CD in the media drive that is attached to wanserver-1 and type the following commands.

```
wanserver-1# cd /cdrom/cdrom0/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 current Solaris release, 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# flarcreate -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 14–1 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 naming 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 current Solaris release 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/cdrom0.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 current Solaris release 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 216, 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 215, 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 217, 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, 512 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 15

WAN Boot (Reference)

This chapter briefly describes the commands and files you use to perform a WAN installation.

- "WAN Boot Installation Commands" on page 225
- "OBP Commands" on page 228
- "System Configuration File Settings and Syntax" on page 229
- "`wanboot.conf` File Parameters and Syntax" on page 229

WAN Boot Installation Commands

The following tables describe the commands you use to perform a WAN boot installation.

- Table 15–1
- Table 15–2

TABLE 15–1 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 15–1 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	`flarcreate –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 15–2 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`
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*

TABLE 15–2 Preparing the WAN Boot Security Files *(Continued)*

Task and Description	Command
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`
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 15–3 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`
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 162.

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 10 10/08 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 235
- "Problems With Booting a System" on page 236
- "Initial Installation of the Solaris OS" on page 241
- "Upgrading the Solaris OS" on page 244

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 naming service.

Solution: Add the host *host_name* to the naming service and run the `add_install_client` command again.

`Error: <system name> does not exist in the NIS ethers map`

`Add it, and rerun the add_install_client command`

Description: When you run the `add_install_client` command, the command fails with the above error.

Cause: The client you are adding to the install server does not exist in the server's `/etc/ethers` file.

Solution: Add the needed information to the `/etc/ethers` file on the install server and run the `add_install_client` command again.

1. Become superuser or assume an equivalent role.
2. On the client, find the ethers address.

```
# ifconfig -a grep ethers
ether 8:0:20:b3:39:1d
```

3. On the install server, open the `/etc/ethers` file in an editor. Add the address to the list.
4. On the client, run `add_install_client` again as in this example.

```
# ./add_install_client bluegill sun4u
```

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 10 10/08 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 DVD or the Solaris 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 DVD or Solaris 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 DVD or the Solaris 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 the current Solaris release was installed (either through the Solaris installation program or custom JumpStart), no boot disk was selected. You now must edit the BIOS to boot the system.

Solution: 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 log file. For more information, see `syslogd(1M)`.

Solution: For SPARC based systems, at the `ok` prompt, type the following command.

```
ok boot net -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 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 at 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 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 naming 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 92.

`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 DVD, either the DVD or a copy of the DVD image on the install server
- Solaris Software - 1 CD image, either the Solaris 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 DVD or Solaris 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 DVD image or Solaris 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+ naming 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 naming service. If the system's host name is listed in the NIS or NIS+ naming 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 an `/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.

`The system does not boot from the network (`**`network installations with DHCP only`**`).`
Cause: The DHCP server is not configured correctly. This error might occur if the system is not configured as an installation client on the DHCP server.

Solution: In the DHCP manager software, verify that installation options and macros are defined for the client system. For more information, see "Preconfiguring System Configuration Information With the DHCP Service (Tasks)" on page 43.

Initial Installation of the Solaris OS

`Initial installation fails`
Solution: If the Solaris installation fails, you must restart the installation. To restart the installation, boot the system from the Solaris DVD, the Solaris 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/sol_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 Device Configuration Assistant 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 Device Configuration Assistant.

---

x86 only – If you are using the `locale` keyword to test a custom JumpStart profile for an initial installation, the `pfinstall -D` command fails to test the profile. For a workaround, see the error message "could not select locale," in the section, "Upgrading the Solaris OS" on page 244.

▼ 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.

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 Boot to the installation media.

3 When you are prompted to select an installation type, select option 6, Single user shell.

4 Start the `format`(1M) program.

```
# format
```

5 Specify the IDE disk drive on which you want to perform a surface analysis.

```
# cxdy
```

cx Is the controller number

d*y* Is the device number

6 Determine if you have an `fdisk` partition.

- If a Solaris `fdisk` partition already exists, proceed to Step 7.
- If a Solaris `fdisk` partition does not exist, use the `fdisk` command to create a Solaris partition on the disk.

```
format> fdisk
```

7. **To begin the surface analysis, type:**

```
format> analyze
```

8. **Determine the current settings, type:**

```
analyze> config
```

9. **(Optional) To change settings, type:**

```
analyze> setup
```

10. **To find bad blocks, type:**

```
analyze> type_of_surface_analysis
```

type_of_surface_analysis Is read, write, or compare

If `format` finds bad blocks, it remaps them.

11. **To exit the analysis, type:**

```
analyze> quit
```

12. **Determine if you want to specify blocks to remap.**

- If no, go to Step 13.
- If yes, type:

```
format> repair
```

13. **To exit the format program, type:**

```
quit
```

14. **Restart the media in multiuser mode by typing the following command.**

```
# exit
```

Upgrading the Solaris OS

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/bzcat not found`

Cause: Solaris Live Upgrade fails because of needing a patch cluster.

Solution: A patch is needed to install Solaris Live Upgrade. Ensure that you have the most recently updated patch list by consulting http://sunsolve.sun.com. Search for the info doc 72099 on the SunSolve web site.

`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 the Solaris Software - 1 CD 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 DVD, a net installation image, or JumpStart.

`ERROR: Could not select locale (x86 based systems only)`

Cause: When you test your JumpStart profile by using the `pfinstall -D` command, the dry run test fails under the following conditions:

- The profile contains the locale keyword.
- You're testing a release that contains GRUB software. **Starting with the Solaris 10 1/06 release**, the GRUB boot loader facilitates booting different operating systems installed on your system with the GRUB menu.

With the introduction of GRUB software, the miniroot is compressed. The software can no longer find the list of locales from the compressed miniroot. The miniroot is the smallest possible Solaris root (`/`) file system and is found on the Solaris installation media.

Solution: Perform the following steps. Use the following values.

- `MEDIA_DIR` is `/cdrom/cdrom0/`
- `MINIROOT_DIR` is `$MEDIA_DIR/Solaris_10/Tools/Boot`
- `MINIROOT_ARCHIVE` is `$MEDIA_DIR/boot/x86.miniroot`
- `TEMP_FILE_NAME` is `/tmp/test`

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. Uncompress the miniroot archive.

   ```
   # /usr/bin/gzcat $MINIROOT_ARCHIVE > $TEMP_FILE_NAME
   ```

3. Create the miniroot device by using the `lofiadm` command.

   ```
   # LOFI_DEVICE=/usr/sbin/lofiadm -a $TEMP_FILE_NAME
   # echo $LOFI_DEVICE
   /dev/lofi/1
   ```

4. Mount the miniroot with the `lofi` command under the Miniroot directory.

   ```
   # /usr/sbin/mount -F ufs  $LOFI_DEVICE  $MINIROOT_DIR
   ```

5. Test the profile.

   ```
   # /usr/sbin/install.d/pfinstall -D -c $MEDIA_DIR $path-to-jumpstart_profile
   ```

6. After the testing is completed, unmount the lofi device.

   ```
   # umount  $LOFI_DEVICE
   ```

7. Delete the `lofi` device.

   ```
   # lofiadm -d $TEMP_FILE_NAME
   ```

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" in *Solaris 10 10/08 Installation Guide: Planning for Installation and Upgrade* 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.

1 **Reboot the system from the Solaris DVD, the Solaris 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.

x86: Problems With Solaris Live Upgrade When You Use GRUB

The following errors can occur when you use Solaris Live Upgrade and the GRUB boot loader on an x86 based system.

`ERROR: The media product tools installation directory` *path-to-installation-directory* `does not exist.`

`ERROR: The media` *dirctory* `does not contain an operating system upgrade image.`

Description: The error messages are seen when using the `luupgrade` command to upgrade a new boot environment.

Cause: An older version of Solaris Live Upgrade is being used. The Solaris Live Upgrade packages you have installed on your system are incompatible with the media and the release on that media.

Solution: Always use the Solaris Live Upgrade packages from the release you are upgrading to.

Example: In the following example, the error message indicates that the Solaris Live Upgrade packages on the system are not the same version as on the media.

```
# luupgrade -u -n s10u1 -s /mnt
    Validating the contents of the media </mnt>.
    The media is a standard Solaris media.
    ERROR: The media product tools installation directory
</mnt/Solaris_10/Tools/Boot/usr/sbin/install.d/install_config> does
not exist.
    ERROR: The media </mnt> does not contain an operating system upgrade
image.
```

`ERROR: Cannot find or is not executable: </sbin/biosdev>.`

`ERROR: One or more patches required by Solaris Live Upgrade has not been installed.`

Cause: One or more patches required by Solaris Live Upgrade are not installed on your system. Beware that this error message does not catch all missing patches.

Solution: Before using Solaris Live Upgrade, always install all the required patches. Ensure that you have the most recently updated patch list by consulting http://sunsolve.sun.com. Search for the info doc 72099 on the SunSolve web site.

`ERROR: Device mapping command </sbin/biosdev> failed. Please reboot and try again.`

Cause: *Reason 1*: Solaris Live Upgrade is unable to map devices because of previous administrative tasks.

Solution: *Reason 1*: Reboot the system and try Solaris Live Upgrade again

Cause: *Reason 2*: If you reboot your system and get the same error message, you have two or more identical disks. The device mapping command is unable to distinguish between them.

Solution: *Reason 2*: Create a new dummy `fdisk` partition on one of the disks. See the `fdisk(1M)` man page. Then reboot the system.

`Cannot delete the boot environment that contains the GRUB menu`

Cause: Solaris Live Upgrade imposes the restriction that a boot environment cannot be deleted if the boot environment contains the GRUB menu.

Solution: Use `lumake(1M)` or `luupgrade(1M)` commands to reuse that boot environment.

`The file system containing the GRUB menu was accidentally remade. However, the disk has the same slices as before. For example, the disk was not re-sliced.`

Cause: The file system that contains the GRUB menu is critical to keeping the system bootable. Solaris Live Upgrade commands do not destroy the GRUB menu. But, if you accidentally remake or otherwise destroy the file system containing the GRUB menu with a command other than a Solaris Live Upgrade command, the recovery software attempts to reinstall the GRUB menu. The recovery software puts the GRUB menu back in the same file system at the next reboot. For example, you might have used the `newfs` or `mkfs` commands on the file system and accidentally destroyed the GRUB menu. To restore the GRUB menu correctly, the slice must adhere to the following conditions:

- Contain a mountable file system
- Remain a part of the same Solaris Live Upgrade boot environment where the slice resided previously

Before rebooting the system, make any necessary corrective actions on the slice.

Solution: Reboot the system. A backup copy of the GRUB menu is automatically installed.

`The GRUB menu's menu.lst file was accidentally deleted.`

Solution: Reboot the system. A backup copy of the GRUB menu is automatically installed.

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.

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. **Create an inactive boot environment. See "Creating a New Boot Environment" in *Solaris 10 10/08 Installation Guide: Solaris Live Upgrade and Upgrade Planning*.**

3. **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 /mnt
      ```

   b. **Change to the directory that contains the `vfstab`, for example:**

      ```
      # cd /mnt/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 /mnt/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. **Create the Veritas `install-db` file, for example:**

```
# touch vx/reconfig.d/state.d/install-db
```

j. **Unmount the inactive boot environment.**

```
# luumount inactive_boot_environment_name
```

4 **Upgrade the inactive boot environment. See Chapter 5, "Upgrading With Solaris Live Upgrade (Tasks)," in *Solaris 10 10/08 Installation Guide: Solaris Live Upgrade and Upgrade Planning*.**

5 **Activate the inactive boot environment. See "Activating a Boot Environment" in *Solaris 10 10/08 Installation Guide: Solaris Live Upgrade and Upgrade Planning*.**

6 **Shut down the system.**

```
# init 0
```

7 **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.

8 **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
```

9 **Restore the original `vfstab` and system files:**

```
# cp /etc/vfstab.original /etc/vfstab
# cp /etc/system.original /etc/system
```

10 **Reboot the system.**

```
# init 6
```

x86: Service Partition Not Created by Default on Systems With No Existing Service Partition

If you install the current Solaris release 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 re-create the service partition before you install the current Solaris release.

If you installed the Solaris 8 2/02 OS 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 OS, you might not be able to re-create the service partition and upgrade to the current Solaris release.

If you want to include a service partition on the disk that contains the Solaris partition, choose one of the following workarounds.

▼ To Install Software From a Network Installation Image or From the Solaris DVD

To install the software from a net installation image or from the Solaris 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 Solaris partition.

▼ To Install From the Solaris Software - 1 CD or From a Network Installation Image

To use the Solaris installation program to install from the Solaris 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. **The installation program prompts you to choose a method for creating the Solaris partition.**
4. **Boot the system.**
5. **Select the `Use rest of disk for Solaris partition` option.**
   The installation program preserves the service partition and creates the Solaris partition.
6. **Complete the installation.**

◆ ◆ ◆ APPENDIX B

Installing or Upgrading Remotely (Tasks)

This appendix describes how to use the Solaris installation 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*.

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*.

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 DVD or the Solaris 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/Solaris_10/Tools
     ```

   - For CD media, type:

     ```
     remote system# cd /cdrom/cdrom0
     ```

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 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`

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 DVD or Solaris 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`
remote system`# unshare /cdrom/cdrom0`

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 Software - 2 CD, repeat Step 9 through Step 11 to unshare the Solaris Software - 1 CD and to export and install the Solaris Software - 2 CD.

13 If the Solaris installation program prompts you to insert additional Solaris Software CDs, repeat Step 9 through Step 11 to unshare the Solaris Software CDs and to export and install the additional CDs.

14 If the Solaris installation program prompts you to insert the first Solaris Languages CD, repeat Step 9 through Step 11 to unshare the Solaris Software CDs and to export and install each Solaris Languages CD.

When you export a Solaris Languages CD, an installer window appears on the machine where the CD-ROM is mounted. Ignore the installer window while you install the Solaris Languages CD. After you complete the installation of the Solaris Languages CDs, close the installer window.

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 archive **x86 only**: A boot archive is a collection of critical files that is used to boot the Solaris OS. These files are needed during system startup before the root (`/`) file system is mounted. Two boot archives are maintained on a system:

- The boot archive that is used to boot the Solaris OS on a system. This boot archive is sometimes called the primary boot archive.
- The boot archive that is used for recovery when the primary boot archive is damaged. This boot archive starts the system without mounting the root (`/`) file system. On the GRUB menu, this boot archive is called failsafe. The archive's essential purpose is to regenerate the primary boot archive, which is usually used to boot the system.

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.

boot loader — **x86 only**: The boot loader is the first software program that runs after you turn on a system. This program begins the booting process.

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.

`bootlog-cgi` **program** — The CGI program that enables a web server to collect and store remote client-booting and installation console messages during a WAN boot installation.

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` **file** — 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 install 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` file of the active and inactive boot environments. Example file systems 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*.

dataset A generic name for the following ZFS entities: clones, file systems, snapshots, or volumes.

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 the `pfinstall` command 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 release.

Entire Solaris Software Group Plus OEM Support A software group that contains the entire Solaris release plus additional hardware support for OEMs. This software group is recommended when installing Solaris software on SPARC based servers.

`/etc` **directory** 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` **file system** 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.

failsafe boot archive **x86 only**: A boot archive that is used for recovery when the primary boot archive is damaged. This boot archive starts the system without mounting the root (`/`) file system. This boot archive is called failsafe on the GRUB menu. The archive's essential purpose is to regenerate the primary boot archive, which is usually used to boot the system. See *boot archive*.

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*.

GRUB **x86 only**: GNU GRand Unified Bootloader (GRUB) is an open source boot loader with a simple menu interface. The menu displays a list of operating systems that are installed on a system. GRUB enables you to easily boot these various operating systems, such as the Solaris OS, Linux, or Microsoft Windows.

GRUB edit menu **x86 only**: A boot menu that is a submenu of the GRUB main menu. GRUB commands are displayed on this menu. These commands can be edited to change boot behavior.

GRUB main menu **x86 only**: A boot menu that lists the operating systems that are installed on a system. From this menu, you can easily boot an operating system without modifying the BIOS or `fdisk` partition settings.

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 called a *media server*). You can create an install server by copying the Solaris DVD or CD images to the server's hard disk.

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.

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` **file** 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 for the purposes 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*.

`menu.lst` **file** **x86 only**: A file that lists all the operating systems that are installed on a system. The contents of this file dictate the list of operating systems that is displayed on the GRUB menu. From the GRUB menu, you can easily boot an operating system without modifying the BIOS or `fdisk` partition settings.

metadevice See *volume*.

miniroot A minimal, bootable root (`/`) file system that is included in Solaris installation media. A miniroot consists of the Solaris software that is required to install and upgrade systems. On x86 based systems, the miniroot is copied to the system to be used as the failsafe boot archive. See *failsafe boot archive*.

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 naming service to systems on a network.

naming 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 naming service, the system information can be maintained, managed, and accessed on a network-wide basis. Without a naming service, each system has to maintain its own copy of the system information in the local `/etc` files. Sun supports the following naming services: LDAP, NIS, and NIS+.

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*.

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.

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 file system 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.

pool A logical group of devices describing the layout and physical characteristics of the available ZFS storage. Space for datasets is allocated from a pool.

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, the Power Management software is installed by default. A `sun4u` SPARC based system is an example of a system that has Power Management 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.

primary boot archive A boot archive that is used to boot the Solaris OS on a system. This boot archive is sometimes called the primary boot archive. See *boot archive*.

private key The decryption key used in public-key encryption.

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.

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-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.

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-Z storage pool A virtual device that stores data and parity on multiple disks that can be used as a ZFS storage pool. RAID-Z is similar to RAID-5.

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 The top level of a hierarchy of items. Root is the one item from which all other items are descended. See *root directory* or *root* (`/`) *file system*.

root (`/`) file system The top-level file system from which all other file systems stem. The root (`/`) file system is the base on which all other file systems are mounted, and is never unmounted. The root (`/`) file system 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.

root directory The top-level directory from which all other directories stem.

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` file 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.

snapshot A read-only image of a ZFS file system or volume at a given point in time.

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, called 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 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 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.

superuser A special user who has privileges to perform all administrative tasks on the system. The superuser has the ability to read and write to any file, run all programs, and send kill signals to any process.

swap space A slice or file that temporarily holds the contents of a memory area till it can be reloaded in memory. Also called the `/swap` or `swap` volume.

`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` **file** 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 preserves 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. 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` **file system** 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` **file system** 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.

virtual device A logical device in a ZFS pool, which can be a physical device, a file, or a collection of devices.

volume Agroup 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 for the purposes 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-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.

`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.

ZFS A file system using storage pools to manage physical storage.

zone See *non-global zone*

Index

Numbers and Symbols

3DES encryption key
  installing with `wanboot` program, 202
  encrypting data for WAN boot installation, 136

A

`add_install_client`, description, 123
add_install_client command
  example
    boot server for DVD media, 73
    for boot server for CD media, 96
    same subnet for CD media, 95
    specifying serial console, 74, 96
    with DHCP for CD media, 95, 96
    with DHCP for DVD media, 72, 73
  example for specifying a serial console, 74, 96
`add_to_install_server`, description, 123
adding
  dataless clients
    with CD media, 92
    with DVD media, 69
  locale.org_dir table entries, 42
  systems from network, 63, 85
AES encryption key
  installing
    with `wanboot` program, 202
  encrypting data for WAN boot installation, 136
archive
  creating an archive, WAN boot installation, 173
  installing with WAN boot, 197-209
archive *(Continued)*
  storing in document root directory for WAN boot installation, 143
  WAN boot profile example, 177

B

`banner` command, 124
boot: cannot open /kernel/unix message, 236
`boot` command syntax for WAN boot installations, 228
`boot_file` parameter, 230
`boot_logger` parameter, 232
boot server
  creating on subnet
    with DVD media, 67
  creating on subnet with CD media, 90
  creating with DVD media, example, 69
  description, 58
  requirement for network installation, 58
`bootconfchk` command, syntax, 226
booting the system, resetting terminals and display first, 124
`bootlog-cgi` program, specifying in `wanboot.conf` file, 232
`bootlog` file, directing to logging server, 166
bootparams file, updating, 241
`bootserver` variable, 201

C

-c option, add_install_client command, 94
Can't boot from file/device message, 236
certificates, *See* digital certificates
`certstore` file
  description, 145
  inserting client certificate, 216
CHANGE DEFAULT BOOT DEVICE message, 242
check script
  syntax for WAN boot installations, 226
  testing rules, 179
client, requirements for WAN boot installation, 140
client and server authentication, configuring for WAN boot installation, 216
`client_authentication` parameter, 231
CLIENT MAC ADDR error message, 241
client_name, description, 95
clock gained xxx days message, 236
color depth, preconfiguring, 39
commands to start an installation, x86 based systems, 80, 102
comments, in `wanboot.conf` file, 229
configuring
  DHCP server to support installation tasks, DVD media, 63, 85
  DHCP service for WAN boot installation, 187-188
  WAN boot server, 155-166
corrupted binaries, with WAN boot installations, 148
CPUs (processors), WAN boot installation requirements, 140
creating
  boot server on a subnet with CD media, 85, 90
  boot server on a subnet with DVD media, 63, 67
  /etc/locale file, 41
  install server with CD media, 85, 86, 115, 119
  install server with DVD media, 63, 64, 114, 116
  WAN boot
    custom JumpStart files, 173-180
    document root directory, 155
    `/etc/netboot` directory, 162-165
    installation files, 173-180
    Solaris Flash archive, 173
    WAN boot miniroot, 156-158
custom JumpStart installation
  examples, WAN boot installation profile, 177
  with WAN boot installation, 173-180

D

-d option, add_install_client command, 94
date and time, preconfiguring, 38
denial of service attacks, with WAN boot installations, 148
`devalias` command, syntax, 228
device drivers, installing, 80, 102
DHCP (Dynamic Host Configuration Protocol), preconfiguring, 38
DHCP service
  configuring for WAN boot installation, 187-188
  creating macros for Solaris install, 48
  creating options for Solaris installation, 44
  description, 43
  sample script for adding options and macros, 52
  Solaris network boot and install, 44
  Sun vendor options for WAN boot installation, 187-188
  WAN boot installation requirements, 140
`dhtadm` command, using in script, 52
digital certificates
  description, 136, 147
  preparing for WAN boot installations, 215-216, 216
  protecting data during WAN boot installation, 136
  requirements for WAN boot installation, 147
directories
  document root
    creating, 155, 213
    description, 142
    example, 142, 213
  `/etc/netboot`
    configuration and security files, description, 145
    description, 144-147
    example, 146
    sharing configuration and security files, 145-147
    sharing configuration and security files among clients, 144
    storing configuration and security files, 144
  `/etc/netboot` directory, 162-165

disk space, requirements for WAN boot
installation, 140
display resolution, preconfiguring, 39
displaying
mounted file systems, 123
platform name, 124
shared file systems, 123
system information, 124
document root directory
creating, 155
description, 142
example, 142, 213
domain name, preconfiguring, 38

E

`eeprom` command, checking OBP support of WAN boot installations, 226
encrypting data during WAN boot installation
with digital certificate, 215-216, 216
with HTTPS, 167-172
with private key, 216
encrypting data with HTTPS, WAN boot installation, 136-137
encryption key
creating, 217
description, 136
encrypting data during WAN boot installation, 136
installing
example, 194, 196, 222
methods to install, 192-197
with `wanboot` program, 202
specifying in `wanboot.conf` file, 230
`encryption_type` parameter, 230
`/etc/bootparams` file, enabling JumpStart directory access, 241
/etc/locale file, 41
`/etc/netboot` directory
configuration and security files, description, 145
configuring client and server authentication, 216
creating, 162-165, 214
description, 144-147
example, 146
`/etc/netboot` directory *(Continued)*
inserting
client private key, 216
digital certificate, 216
trusted certificate, 215-216
permissions, 162-165
sharing configuration and security files among clients, 144, 145-147
storing configuration and security files
entire network installations, 144, 162
entire subnet installations, 144, 162
single-client installations, 144, 163

F

failed upgrade, rebooting problems, 246
`file` variable, 199
files and file systems
displaying mounted file systems, 123
displaying shared file systems, 123
system configuration syntax, 229
WAN boot file system, 132
`wanboot.conf`
description, 229-232
syntax, 229-232
`flarcreate` command, syntax for WAN boot installations, 226

G

graphical user interface (GUI), command to start (x86 based systems), 80, 103
graphics card, preconfiguring, 39
GRUB based booting
command reference, 124-128
installing x86 clients over the network with (DVD), 77, 100

H

hard disks, size, space available, 65

hashing key
creating, 217
description, 136
installing
example, 222
methods to install, 192-197
with `wanboot` program, 202
protecting data during WAN boot installation, 136
specifying in `wanboot.conf` file, 230
HMAC SHA1 hashing key, *See* hashing key
`host-ip` variable, 199
host name, preconfiguring, 38
`hostname` variable, 199
HTTP over Secure Sockets Layer, *See* HTTPS
`http-proxy` variable, 199
HTTPS
description, 136-137
protecting data during WAN boot installation, 136-137
requirements to use with WAN boot, 167-172

I

install server
creating with CD media, 86
creating with CD media, example, 89, 115, 119
creating with DVD media, 64
creating with DVD media, example, 66, 114, 116
on subnet, 66, 109, 112
system types applicable, 57-59
WAN boot installation requirements, 140
install time updates (ITUs), installing, 80, 102
installation, WAN boot, description, 131-132
installing
device drivers, 80, 102
install time updates (ITUs), 80, 102
IP addresses
preconfiguring, 38
preconfiguring a default route, 38
IPv6, preconfiguring, 38
IRQ level, preconfiguring, 39

K

Kerberos, preconfiguring, 38
keyboard language and layout, preconfiguring, 39
keys, *See* encryption key, hashing key
`keystore` file
description, 145
inserting client private key, 216
keywords, sysidcfg file, 22-36

L

le0: No carrier - transceiver cable problem message, 236
`list-security-keys` command, syntax, 228
locale file, 41
locale.org_dir table, adding entries, 42
log files, for WAN boot installation, 166
logging server
configuring for WAN boot installation, 215
description, 140
location of log messages, 166
WAN boot installation requirements, 140
logging server, specifying in `wanboot.conf` file, 232

M

Makefile file, 40
memory, WAN boot installation requirements, 140
monitor type, preconfiguring, 38
`mount` command, 123
mounting, displaying mounted file systems, 123

N

name server, preconfiguring, 38
names/naming
host name, 95
system configuration file for WAN boot installation, 181
system platform name determination, 124
naming service, preconfiguring, 38
`net` device alias, checking and resetting, 191, 221-222

netmask, preconfiguring, 38
network-boot-arguments OBP variables
  setting in WAN boot installations, 201
  syntax, 228
network installation
  See also WAN boot installation
  description, 57-59
  preparing, 57-59
  requirements, 57-59
  using CD media, 86, 91
  using DVD media, 64, 67
  using PXE, 59-60
  WAN boot installation example, 211-223
network interface, preconfiguring, 38
nistbladm command, 42, 43
No carrier - transceiver cable problem message, 236
Not a UFS filesystem message, 236
nvalias command, syntax, 228

O

OBP
  checking for WAN boot support, 159, 213
  checking net device alias, 191, 221-222
  setting net device alias, 191
  setting variables in WAN boot installations, 201
  WAN boot installation requirements, 140
OpenBoot PROM, *See* OBP
output files, bootlog file for WAN boot
  installation, 166

P

-p option of check script, 179
permissions, /etc/netboot directory, 164
PKCS#12 file
  preparing for WAN boot installation, 216
  requirements for WAN boot installation, 147
planning
  WAN boot installation
    information required to install, 148-149
    server layout, 141-142
    sharing configuration and security files, 145-147
planning, WAN boot installation *(Continued)*
    storing configuration and security files, 144-147
    storing installation files, 142
    storing wanboot-cgi program, 147
    system requirements, 139
    web server requirements, 141
platforms
  install server setup, 95
  name determination, 124
pointing device, preconfiguring, 39
Power Management, 36
Preboot Execution Environment (PXE)
  BIOS setup requirements, 77, 100
  description, 60
  guidelines, 60
preconfiguring system configuration information
  advantages, 17-18
  choosing a method, 37-39
  with DHCP, 43
  Power Management, 36
  using a naming service, 39
  using sysidcfg file, 39
preparing for installation
  client for WAN boot installation, 190-197
  preconfiguring system information
    advantages, 17-18
    methods, 37-39
  WAN boot installation, 151-188
primary document directory, *See* document root
  directory
printenv command, checking for WAN boot
  support, 213
privacy issues with WAN boot installations, 148
processors, WAN boot installation requirements, 140
profiles
  examples
    WAN boot installation, 177
  naming, 176
protecting data during WAN boot installation
  with encryption key, 136
  with hashing key, 136
  with HTTPS, 136-137
PXE (Preboot Execution Environment)
  BIOS setup requirements, 77, 100

PXE (Preboot Execution Environment) *(Continued)*
description, 60
guidelines, 60

R

requirements
network installation, servers, 57-59
WAN boot installation, 139
`reset` command, 124
resetting display and terminal after I/O interrupts, 124
`resolve_hosts` parameter, 231
`root_file` parameter, 230
root password, preconfiguring, 38
`root_server` parameter, 230
`router-ip` variable, 199
RPC Timed out message, 240
rules, validating for WAN boot installation, 179
rules file, validating for WAN boot installation, 179

S

`SbootURI` DHCP option
description, 47
using with WAN boot installations, 187
screen size, preconfiguring, 39
secure HTTP, *See* HTTPS
Secure Sockets Layer, using with WAN boot installation, 167-172
security
WAN boot installation
description, 135-137
security issues for WAN boot installations, 148
security policy, preconfiguring, 38
serial console, 79, 101
specifying with `add_install_client` command, 74, 96
`server_authentication` parameter, 231
servers
network installation setup with CD media
standalone installation, 92
network installation setup with DVD media
standalone installation, 69
servers *(Continued)*
requirements for network installation, 57-59
WAN boot installation
configuration options, 141-142
descriptions, 139
requirements, 139
web server software requirements, 141
`set-security-key` command
installing keys on WAN boot client, 222
syntax, 228
`setenv` command, syntax, 228
setting up a serial console, 79, 101
`setup_install_server`
description, 123
for WAN boot installation, 156-158
syntax for WAN boot installations, 225
sharing, WAN boot configuration
information, 145-147
`showmount` command, 123
`SHTTPproxy` DHCP option
description, 48
using with WAN boot installations, 187
`signature_type` parameter, 230
size, hard disk, space available, 65
`SjumpsCF` parameter, 181, 229
Solaris installation program
graphical user interface (GUI), command to start (x86 based systems), 80, 103
text installer
command to start in console session (x86 based systems), 80, 103
command to start in desktop session (x86 based systems), 80, 103
SSL, using with WAN boot installation, 167-172
`SsysidCF` parameter, 181, 229
starting an installation, x86 based systems, 80, 103
subnet
boot server creation on, with CD media, 90
boot server creation on, with DVD media, 67
`subnet-mask` variable, 199
`sysidcfg` file
guidelines and requirements, 18-36
`keyboard` keyword, description, 23-24
keywords, 22-36

sysidcfg file *(Continued)*
name_service keyword, description, 24-27
network_interface keyword, description, 27-32
root_password keyword, description, 33
security_policy keyword, description, 33-34
service_profile keyword, description, 34
syntax, 21-22
system_locale keyword, description, 35
terminal keyword, description, 35
timeserver keyword, description, 35-36
timezone keyword, description, 35
WAN boot, example, 175
system.conf file, *See* system configuration file
system_conf parameter, 232
system configuration file
creating for WAN boot installation, 219-220
description, 145
examples
insecure WAN boot installation, 182
secure WAN boot installation, 182, 219-220
SjumpsCF setting, 229
specifying in wanboot.conf file, 232
SsysidCF setting, 229
syntax, 229
system information, displaying, 124

T

terminal type, preconfiguring, 38
testing
WAN boot
rules file, 179
wanboot.conf file, 184
text installer
command to start in console session (x86 based systems), 80, 103
command to start in desktop session (x86 based systems), 80, 103
time and date, preconfiguring, 38
time zone, preconfiguring, 38
timed out RPC error, 240
token ring card, booting error with, 240
transceiver cable problem message, 236
Triple DES encryption key, *See* 3DES encryption key
troubleshooting
booting from network with DHCP, 241
booting from wrong server, 241
general installation problems
booting from the network with DHCP, 241
booting the system, 241
trust anchor, *See* trusted certificate
trusted certificate, inserting in truststore file, 215-216
truststore file
description, 145
inserting trusted certificate, 215-216

U

uname command, 124
Unknown client error message, 235
upgrade, failed upgrade, 246

V

validating
rules files, for WAN boot installation, 179
wanboot.conf file, 184
/var/yp/make command, 42
/var/yp/Makefile, 40

W

WAN boot file system, description, 132
WAN boot installation
bootlog-cgi program, specifying in wanboot.conf file, 232
checking rules file, 179
client requirements, 140
client authentication
requirements, 137-138
specifying in wanboot.conf file, 231
commands, 225-227
configuration and security files, description, 145
configuring
client and server authentication, 216

WAN boot installation, configuring *(Continued)*
DHCP service support, 187-188
WAN boot server, 155-166
copying `wanboot-cgi` program, 165
corrupted binaries, 148
creating
begin scripts, 180
finish scripts, 180
Solaris Flash archive, 173
denial of service attacks, 148
description, 131-132
digital certificates, requirements, 147
document root directory
description, 142
example, 142
files, 142
encrypting data
with encryption key, 136
with HTTPS, 136-137, 167-172
encryption key
displaying value, 192-197
installing, 192-197
specifying in `wanboot.conf` file, 230
encryption key privacy issues, 148
`/etc/netboot` directory
creating, 162-165
description, 144-147
example, 146
setting permissions, 163
examples
checking client OBP support, 159, 213
checking `net` device alias, 191, 221-222
configuring logging server, 166, 215
copying `wanboot-cgi` program, 215
creating `/etc/netboot` directory, 164
creating custom JumpStart profile, 218-219
creating encryption key, 172, 217
creating hashing key, 172, 217
creating `rules` file, 219
creating Solaris Flash archive, 217
creating `sysidcfg` file, 218
creating system configuration file, 219-220
creating the `/etc/netboot` directory, 214
creating the WAN boot miniroot, 213
WAN boot installation, examples *(Continued)*
custom JumpStart profile, 177
document root directory, 213
enabling client authentication, 216
enabling server authentication, 169, 216
`/etc/netboot` directory, 146
inserting client certificate, 169, 216
inserting client private key, 169, 216
inserting trusted certificate, 169, 215-216
installing encryption key in OBP, 194, 222
installing encryption key on running client, 196
installing from local CD media, 208
installing hashing key in OBP, 194, 222
installing hashing key on running client, 196
installing `wanboot` program, 214
installing with DHCP service, 205
interactive installation, 202
network setup, 212
noninteractive installation, 199, 223
preparing digital certificates, 216
setting `net` device alias, 191
`sysidcfg` file, 175
system configuration file, 182
unattended installation, 199, 223
using encryption, 217
`wanboot.conf` file, 184, 186, 220-221
hashing key
displaying value, 192-197
installing, 192-197
specifying in `wanboot.conf` file, 230
hashing key privacy issues, 148
information required to install, 148-149
insecure configuration, 138
installing a client
methods to install, 197
required tasks, 189
installing encryption key, 192-197
installing hashing key, 192-197
installing the `wanboot` program, 160-162
logging server, specifying in `wanboot.conf` file, 232
noninteractive installation, 223
planning
document root directory, 142
`/etc/netboot` directory, 144-147

WAN boot installation, planning *(Continued)*
server layout, 141-142
sharing configuration and security files, 144
storing configuration and security files, 144-147
storing installation files, 142
system requirements, 139
protecting data, 136
requirements
client CPU, 140
client disk space, 140
client memory, 140
DHCP service, 140
digital certificates, 147
install server disk space, 140
logging server, 140
OBP for client, 140
operating system for web server, 141
SSL version support, 141
WAN boot server, 139
web proxy, 141
web server, 141
secure configuration
description, 137-138
requirements, 137-138
tasks to install, 151
security configurations, description, 137-138
security issues, 148
sequence of events, 133-135
server authentication
requirements, 137-138
specifying in `wanboot.conf` file, 231
server configurations, description, 141-142
sharing configuration and security files
entire network, 144, 162
entire subnet, 144, 162
specific client, 144, 163
storing the `wanboot-cgi` program, 147
system requirements, 139
system configuration file
specifying in `wanboot.conf` file, 232
syntax, 229
unattended installation, 223
WAN boot miniroot
creating, 156-158
WAN boot installation, WAN boot miniroot *(Continued)*
description, 132
specifying in `wanboot.conf` file, 230
storing in document root directory, 143
`wanboot-cgi` program, 165
copying to WAN boot server, 165
specifying in `wanboot.conf` file, 230
`wanboot.conf` file
parameters, 229-232
syntax, 229-232
validating, 184
`wanboot` program
description, 131
installing, 160-162
specifying in `wanboot.conf` file, 230
storing in document root directory, 143
`wanbootutil` command
creating encryption key, 217
creating hashing key, 217
creating private key, 168
creating trusted certificate, 168
web server requirements, 141
when to use, 132
WAN boot miniroot
creating, 156-158, 213
description, 132
specifying in `wanboot.conf` file, 230
storing in document root directory, 143
WAN boot server
configuring, 155-166
copying `wanboot-cgi` program, 165
description, 139
requirements, 139
web server requirements, 141
`wanboot-cgi` program
copying to WAN boot server, 165, 215
description, 144
order of search through `/etc/netboot` directory, 145
selecting client configuration information, 145
specifying in `wanboot.conf` file, 230
storing, 147

wanboot.conf file
creating for WAN boot installation, 220-221, 229-232
description, 145, 229-232
examples
insecure WAN boot installation, 186
secure WAN boot installation, 184, 220
syntax, 229-232
validating for WAN boot installation, 184, 220-221
wanboot program
description, 131
installing keys for WAN boot installation, 202
installing on WAN boot server, 160-162, 214
storing in document root directory, 143
tasks performed during WAN boot installation, 135
wanboot program, specifying in wanboot.conf file, 230
wanbootutil command
configuring client and server authentication, 168, 216, 217
creating a hashing key, 217
creating an encryption key, 217
displaying a hashing key value, 222
displaying an encryption key value, 222
inserting client digital certificate, 168, 216
inserting client private key, 168, 216
inserting trusted certificate, 168, 215-216
splitting a PKCS#12 file, 168, 215-216, 216
WARNING: CHANGE DEFAULT BOOT DEVICE, 242
WARNING: clock gained xxx days message, 236
web proxy, WAN boot installation requirements, 141
web proxy, preconfiguring, 38