

Sun Solaris 10

Deployment Guide for HP ProLiant Servers



Legal notices

© Copyright 2007, 2009 Hewlett-Packard Development Company, L.P.

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Microsoft and Windows are U.S. registered trademarks of Microsoft Corporation. UNIX is a registered trademark of The Open Group. Intel is a trademark of Intel Corporation in the U.S. and other countries. Java is a U.S. trademark of Sun Microsystems, Inc. AMD is a trademark of Advanced Micro Devices, Inc.

Contents

Introduction	6
Applicable servers and OS versions.....	6
Using this guide.....	7
Basic installations.....	8
Prerequisites	8
Installing Solaris interactively.....	9
Architecture of an installation.....	10
General process flow	10
Primary installation components.....	10
GRUB	10
Miniroot.....	11
Configuration information	11
Packages	11
Network deployment infrastructure.....	11
DHCP server.....	12
TFTP server	12
JumpStart server.....	12
Install server	12
Deploying ProLiant servers	12
Additional drivers	13
Miniroot.....	13
JumpStart	14
iLO virtual media	14
Rapid Deployment Pack.....	14
Flash archives.....	15
Deployment resource kit.....	15
Deployment planning.....	16
Selecting a deployment strategy	16
Boot methods.....	16
Configuration methods	17
Software delivery methods	17
Implementing a PXE boot-based deployment	17
Implementing an Rapid Deployment Pack -initiated deployment	18
Implementing a custom ISO boot-based deployment	18
Deployment tasks	19
Install server preparation	19
Prerequisites for preparing an install server.....	19
Preparing the install server	20
Miniroot preparation	21
Prerequisites for preparing a miniroot	21
Preparing a miniroot	21
JumpStart preparation	23
Overview of JumpStart preparation.....	23
Prerequisites for JumpStart preparation.....	23
Example: JumpStart preparation	24
DHCP/PXE preparation	26

Prerequisites for DHCP/PXE preparation.....	26
Preparing the DHCP/PXE server	27
Rapid Deployment Pack server preparation.....	28
Overview of Rapid Deployment Pack server preparation	29
Prerequisites for Rapid Deployment Pack server preparation	29
Configuring the Rapid Deployment Pack deployment server	29
GRUB preparation	30
Overview of GRUB preparation	30
Booting Solaris from GRUB	31
Examples: booting Solaris from GRUB.....	32
Custom bootable ISO image creation.....	33
Prerequisites for creating a custom bootable ISO image.....	33
Creating a custom bootable ISO image.....	33
Optional configuration tasks	34
Enabling iLO virtual serial port	34
Mirroring disks with DiskSuite	34
Unmirroring disks with DiskSuite.....	36
Troubleshooting	38
Known issues.....	38
Write cache not flushed on shutdown.....	38
Solaris stops responding on HP ProLiant DL580 G5 server	38
RAID mode not supported	39
Kernel Panic after disabling PCI devices in RBSU.....	39
Network configuration fails during interactive installation	39
iLO boot hang	40
AMD memory scrubber.....	40
SNMP master agent	40
bcme fails to attach	41
System panics and reboots during installation	42
Embedded Ethernet not configured	42
Hard Drives not seen after CPQary3 added to Miniroot.....	42
Basic troubleshooting techniques	43
Shell prompt.....	43
Network verification.....	43
GRUB command line	44
Kernel debugger.....	44
Common issues.....	44
System does not boot to GRUB	44
GRUB does not display a menu	44
GRUB cannot boot Solaris	45
JumpStart deployment ignores GRUB command line arguments.....	45
System becomes unresponsive using remote console.....	45
Rapid Deployment Pack job fails	45
Installation terminates prematurely	46
JumpStart reverts to interactive configuration.....	46
Installation succeeds but server fails to boot.....	47
Installation succeeds but no network availability after reboot	47
Additional documentation	48
Where to go for additional help	48
HP documentation.....	48
Free Software Foundation documentation	48
Intel documentation	48

Sun Microsystems documentation.....	48
Glossary	50
Index.....	52

Introduction

This guide describes requirements for Solaris installation that are specific to ProLiant, how to apply Solaris deployment mechanisms to ProLiant servers, and how to use specific ProLiant technologies for Solaris installation.

Applicable servers and OS versions

Information in this guide applies to the following releases of Sun Solaris:

- Solaris 10 1/06
- Solaris 10 6/06
- Solaris 10 11/06
- Solaris 10 8/07
- Solaris 10 5/08
- Solaris 10 10/08
- Solaris 10 5/09

Information in this guide applies to the ProLiant server models listed in Table 1.

Table 1 Applicable HP ProLiant servers

BL20p G4	DL140 G2	DL385
BL25p	DL145	DL385 G2
BL25p G2	DL145 G2	DL385 G5
BL260c G5	DL160 G5	DL385 G5p
BL2x220c G5	DL165 G5	DL580 G3
BL35p	DL180 G5	DL580 G4
BL45p	DL185 G5	DL580 G5
BL45p G2	DL360 G4p SCSI	DL585
BL460c	DL360 G5	DL585 G2
BL460c G6	DL360 G6	DL585 G5
BL465c	DL365	DL785 G5
BL480c	DL365 G5	ML370 G4
BL490c G6	DL370 G6	ML370 G5
BL680c G5	DL380 G4	ML370 G6
BL685c	DL380 G5	ML570 G3
BL685c G5	DL380 G6	ML570 G4
DL385 G6	DL585 G6	BL465c G6

Table 1 Applicable HP ProLiant servers

BL280c G6	DL320 G5p	DL320 G6
-----------	-----------	----------

Using this guide

For effective use of this guide, follow these guidelines:

- For interactive installations using the Solaris media, see “Basic installations.”
- For an introduction to the processes and components used in advanced Solaris deployments, see “Architecture of an installation.” For deployment information specific to ProLiant, see “Deploying ProLiant servers.”
- For common deployment setup procedures, see “Deployment tasks.”
- To integrate ProLiant servers into an existing Solaris deployment environment, see the following sections:
 - For readers new to Solaris on ProLiant servers, see “Deploying ProLiant servers.”
 - For specific examples of deployment setup tasks required for ProLiant servers, see “Miniroot preparation” and “JumpStart preparation.”
- To plan a new deployment environment, see the following sections:
 - For those new to Solaris or Solaris on ProLiant servers, see “Architecture of an installation” and “Deploying ProLiant servers.”
 - For information about deployment strategies and guidance on selecting a strategy based on various environmental requirements, see “Deployment planning.”
 - For information on various deployment setup procedures, see “Deployment tasks.”
- For troubleshooting information, see “Troubleshooting.”
- For a list of other related documentation, see “Additional documentation.”
- For a list of terms, see “Glossary.”

NOTE: In this guide, citations of pages from the Solaris reference manual appear in italics next to a number in parentheses, for example, *sysidcfg(4)*. The number in parentheses represents the section of the manual where the cited page is located.

This guide focuses on initial deployment of the Solaris operating environment. Topics beyond the scope of this guide include automated post-installation configuration, application deployment, and operating system upgrades. For more information about these topics, see the following documentation:

- Solaris 10 11/06 Release and Installation Collection, at <http://docs.sun.com/app/docs/coll/1236.5>
- *Solaris 10 11/06 Installation Guide: Custom JumpStart and Advanced Installations*, at <http://docs.sun.com/app/docs/doc/819-6397>
- *Solaris 10 11/06 Installation Guide: Solaris Flash Archives (Creation and Installation)*, at <http://docs.sun.com/app/docs/doc/819-6398>

Basic installations

This chapter covers basic ProLiant deployments using the Solaris media. For more information on other types of deployments, see “Using this guide.”

Prerequisites

To perform a basic installation of Sun Solaris on HP ProLiant servers, the following prerequisites must be met:

- HP ProLiant Server—Use the ProLiant server support matrix for Solaris to determine which releases of Solaris 10 are supported on your ProLiant.
- Sun Solaris 10 Distribution Media—Obtain a copy of the appropriate release of Solaris 10 from Sun Microsystems. Both CD and DVD media are supported.
- Additional Drivers—Depending on your ProLiant model, one or more of the following additional drivers that are not found on the Solaris media might be required. These drivers are available from <http://www.hp.com> on the “Software and Drivers” page associated with each ProLiant model.
 - HP Smart Array controller driver—If you are installing to disks attached to a Smart Array controller, the Smart Array driver cpqary3 is required during installation. For information about additional requirements, including the use of a Driver Update diskette or CD, see the *HP Smart Array Controller Driver for Solaris 10 User Guide* at ftp://ftp.compaq.com/pub/softlib2/software1/pubsw-windows/p308167783/v40878/HP_Smart_Array_Controller_Driver_User_Guide.pdf. The Smart Array driver is listed as “HP Driver Update (DU) for Solaris (x86) 10” on the “Software and Drivers” page.
 - Broadcom NetXtreme II driver—If you are using network interface card (NIC) models NC370x, NC371x, NC373x, NC374x, or NC380x, the bnx driver is required. This driver is applied post-installation.
- CD or DVD drive—A CD-ROM or DVD-ROM drive is required to load the Solaris software. A Universal Serial Bus (USB) drive or iLO virtual media can be used if an internal drive is not available.
- HP SmartStart CD—For advanced Smart Array configuration, HP SmartStart is required to set up the arrays. For basic array setup, the Option ROM Configuration for Arrays (ORCA) utility can be used instead.
- Firmware updates—HP recommends using the latest system ROM and firmware. System ROM, Smart Array controller firmware, and iLO firmware updates should be applied when available.
- iLO client—If using iLO to perform the installation, a supported iLO client is required.
- iLO Advanced license—If using iLO to perform the installation, an iLO Advanced license is required. This license is included with all blades.

The most efficient method to deploy Solaris on ProLiant server blades is through the network. If network deployment is not available, a USB CD/DVD drive can be connected to the blade or iLO virtual media can be used.

You can install Sun Solaris software using only virtual media, but it is not recommended. For best results, verify that the iLO firmware is up-to-date. Connecting media images over HTTP using the iLO secure shell interface can yield improved performance over the Java applet browser interface.

Installing Solaris interactively

The following procedure for installing Solaris on a ProLiant server might require adaptation to fit your target environment or as recommended by the ProLiant server support matrix at <http://www.hp.com/wwsolutions/solaris>, relevant HP advisories, or HP support personnel.

To install Solaris on a ProLiant server:

1. Ensure you have met all the requirements for installation and have all the necessary tools available for installation, including a Smart Array Driver Update (DU) diskette or CD, if necessary. For more information, see "Prerequisites."
2. Apply system ROM and other firmware updates, as necessary.
3. If installing to disks attached to a Smart Array controller, configure the arrays as necessary. The ORCA utility can be used to configure basic arrays. Use the array configuration utility available on the HP SmartStart CD for a more complex array configuration.



IMPORTANT: If the ORCA utility is used to create a new array, the system continues booting after exiting the utility. However, the system must be rebooted (for example, by pressing Ctrl+Alt+Delete) after exiting the utility and before booting the installation media. If the system is not rebooted, the Smart Array driver might not recognize the newly created logical volume..

4. Choose one of the following:
 - Insert the first Solaris CD or DVD into the drive.
 - If using virtual media, attach the CD, DVD, or ISO image using iLO, and then boot the server.
5. If installing to disks attached to a Smart Array controller, follow the procedure described in the *HP Smart Array Controller Driver for Solaris 10 User Guide* at ftp://ftp.compaq.com/pub/softlib2/software1/pubsw-windows/p308167783/v40878/HP_Smart_Array_Controller_Driver_User_Guide.pdf to load the driver. This step is time-sensitive because a timeout might occur on the menu that appears shortly after boot. The root disk must be SCSI target 0, LUN 0 (that is, `cxt0d0`).
6. Follow the onscreen instructions and the *Solaris 10 11/06 Installation Guide: Basic Installations* at <http://docs.sun.com/app/docs/doc/819-6394>. The following known issues or workarounds might apply:
 - If a driver update such as the Smart Array driver was applied at the beginning of the installation, and you are configuring the network using a static IP address, select **non-networked** when prompted during the installation. Configure the network post-installation, as described in step 7.
 - If using a NIC that requires the bnx driver, select **non-networked** when prompted during the installation. Install the driver and configure the network post-installation, as described in step 7.
7. If the bnx driver is required, then after installation is complete, install the bnx driver according to the documentation that accompanies the driver distribution.

If you selected non-networked in step 6, after installation is complete, you can:

- Use `sys-unconfig(1M)` to clear the system identification and reboot the system. Upon restart, Solaris prompts you for system identification, at which time you can select networked.
- Manually configure the network instead of running `sys-unconfig`.

Architecture of an installation

This chapter describes the components, processes, and general architecture of a Solaris installation. To successfully construct more complex Solaris deployments, review the following information.

General process flow

A Solaris deployment consists of the following phases:

- **Boot**—The first phase of deployment is booting a miniroot on the target system. The miniroot contains the Solaris installation software. The system can be booted from a variety of sources, including the Solaris media (CD or DVD), the network (PXE), a custom-built ISO image/CD, and hard drive.
- **Configuration**—The second phase of deployment is configuring of the installation, which includes system identification (hostname, network parameters, root password, and so on), target root disk, and so on. This identification can be performed manually or automatically using JumpStart.
- **Software installation**—The final phase of deployment is the actual loading of the Solaris operating environment software onto the system. The software packages can be loaded from the Solaris media, over the network through a Network File System (NFS), or in the form of a flash archive, which can be loaded from disk or other storage media or over the network.
- **Optional configuration**—After installation, make ProLiant-specific configuration updates to the deployment.

Primary installation components

All Solaris installations follow the same flow as described previously, and are composed of the same primary components. Deployments differ primarily in the way these components are delivered. For example, in a CD-ROM-based interactive installation, the Administrator interactively supplies the system configuration, and the packages that contain the Solaris operating environment are loaded from the physical media. By contrast, for a fully automated installation, the system configuration can be obtained from a JumpStart server, and the packages can be loaded through a NFS.

The following sections briefly describe the primary installation components, their roles, and delivery mechanisms. A key aspect of this architecture is that the delivery mechanisms for each component are generally independent of the others. This component independence provides a great amount of flexibility and allows for the creation deployment scenarios to fit most environments. Other sections discuss how various types of deployments can be constructed using these mechanisms.

GRUB

Grand Unified Bootloader (GRUB) is the default bootloader for Solaris starting with the Solaris 10 1/06 release. GRUB loads the Solaris multiboot program and the miniroot during installations or in failsafe mode. GRUB can also be used to pass boot options and parameters to Solaris.

GRUB is loaded by the system BIOS or option ROMs (such as NICs and Fibre Channel adapters), and can be loaded from a wide variety of devices, including CDs, DVDs, diskettes, iLO virtual media, hard drives, USB storage, Storage Area Networks (SANs), and Preboot Execution Environment/Trivial File Transfer Protocol (PXE/TFTP).

Miniroot

A Solaris miniroot is a compressed disk image of a Unix File System (UFS) root file system that is similar to a Linux `initrd` file. A miniroot is a small but functional Solaris system that contains a 32-bit kernel, device drivers, the Solaris installation program, and various tools and utilities.

GRUB loads the miniroot into memory, along with the Solaris multiboot program. GRUB passes control to the multiboot program, which loads and executes the Solaris kernel from the in-memory miniroot, passing along various options and parameters from GRUB. The multiboot program and miniroot can be loaded through most media that GRUB supports, such as a local media or storage device (through BIOS calls) and PXE/TFTP.

A Solaris miniroot also assists with recovery from various errors or failures because it contains a comprehensive set of tools and utilities. Other important components of the miniroot include:

- Drivers—The miniroot must contain all device drivers that are required for installation, or those drivers must be loaded before installation begins. For ProLiant servers, these drivers usually include the Smart Array driver and network drivers for network-based deployments. For more information about installing additional drivers into the miniroot, see “Miniroot preparation.”
- Installer—The miniroot contains the Solaris installer, which orchestrates the actual installation of Solaris onto the target disks. Boot parameters can be used to provide the installer with deployment configuration information. For more information, see “GRUB preparation.”

Configuration information

The configuration information necessary to complete an installation includes target disk, partition, file system layout, packages to be installed, network configuration, root password, and so on.

This information can be supplied interactively by the Administrator at the time of installation or automatically through NFS using a JumpStart profile and the `sysidcfg(4)` file.

Packages

Packages on the Solaris media contain the software, documentation, and other data that comprise the Solaris operating environment. Some or all of these packages are loaded onto the system by the installer according to the selected configuration.

These packages can be loaded from various storage media (CD, DVD, and hard disk) or through the network (NFS and iLO virtual media).

Packages and other system configuration information can also be delivered in a flash archive. Flash archives can be loaded through FTP and HTTP, in addition to the methods previously listed.

Network deployment infrastructure

Solaris network-based deployments require several services to boot the target servers, deliver configuration information, and provide the Solaris software. The following sections briefly introduce the required services and the role of each.

DHCP server

A Dynamic Host Configuration Protocol (DHCP) server is required to provide systems with network configuration parameters for use during network deployments. If deployments are booted through PXE, then the DHCP server can also provide boot information. If the HP ProLiant Essentials Rapid Deployment Pack is used, a separate DHCP server for PXE boot is not required.

TFTP server

When PXE booting is used, a TFTP server must provide the booting systems with the files that are necessary to properly boot. For Solaris this server includes `pxegrub`, the GRUB menu file `menu.lst`, the Solaris `multiboot` program, and a miniroot. The DHCP/PXE server often hosts the TFTP server. If the Rapid Deployment Pack is used, a separate TFTP server is not required.

JumpStart server

JumpStart consists of a set of files which the Solaris installation software uses to automatically configure a deployment. The JumpStart files included profiles and `sysidcfg(4)` files. A JumpStart server is a set of directories and files on a networked system that provides JumpStart profiles and `sysidcfg(4)` files to clients during network deployments. The profile and `sysidcfg` files can provide the Solaris installation software with all the information necessary to perform an automatic deployment of Solaris. JumpStart files are typically served through an NFS share. For information about setting up a JumpStart server, see “JumpStart preparation.”

Install server

An install server is a networked system that contains a copy of the Solaris installation media that it makes available to other systems on the network through NFS. During a network deployment, a client obtains the Solaris operating environment software from the install server instead of from the physical media such as a CD or DVD. For information about setting up an install server, see “Install server preparation.”

Deploying ProLiant servers

While you can use traditional Solaris deployment mechanisms to deploy ProLiant servers, some requirements are ProLiant-specific. In addition, you can use some HP technologies to enhance Solaris deployments. The following sections provide an overview of these requirements and technologies.

Information in the following sections is for use with network-based or other advanced deployment methods. For basic interactive installation information, see “Basic installations.” For more information about the tasks, see “Deployment tasks.”

Additional drivers

Every new ProLiant generation is enhanced with newer device technologies, and ProLiant servers often include some devices specific to HP. Frequently, drivers for one or more devices are not available on the Solaris media. The following section describes the current set of drivers not available on the Solaris media.

The following drivers are available for download from <http://www.hp.com> on the “Software and Drivers” page associated with each ProLiant server model:

- Smart Array Driver—Most ProLiant servers are equipped with an HP Smart Array RAID controller. This driver for this RAID controller is not currently included on the Solaris media. To install Solaris to disks attached to the controller, the cpqary3 driver must be loadable at installation time. To boot from the controller, the driver must be installed on the system during deployment. For more information, see “Miniroot” and “JumpStart.”

NOTE: On the “Software and Drivers” load page, the cpqary3 driver is listed as “HP Driver Update (DU) for Solaris (x86) 10.”

- Broadcom NetXtreme II driver—Current ProLiant generations use newer Broadcom NIC chipsets for embedded NICs. The driver for these chipsets does not currently ship on the Solaris media. To perform network deployments using these NICs, the bnx driver must be loadable at installation time. To use the NICs after installation is complete, the driver must be installed on the system. For more information, see “Miniroot” and “JumpStart.”

The following NIC models require the bnx driver:

- NC370x
- NC371x
- NC373x
- NC374x
- NC380x

Miniroot

All device drivers required during installation, such as the disk driver for targeted storage, must be loaded into the miniroot. You can load the device drivers manually at run-time. Alternatively, you can install the driver into the miniroot before booting the client for automated deployments or ease of interactive installations.

When loading device drivers into the miniroot, follow these guidelines:

- For most ProLiant servers, the Smart Array driver, cpqary3, is installed into the miniroot so that the Solaris software can be installed onto disks attached to a Smart Array controller.
- For network-based deployments, the miniroot must have a driver for the network interface to be used. For current ProLiant generations, the Broadcom NetXtreme II driver, bnx, must be installed into the miniroot.
- For network-based deployments that do not boot from the network, such as those that are initiated by Rapid Deployment Pack, or those that boot from a custom CD ISO, the HPQnetstrategy package must be installed into the miniroot. For more information about the HPQnetstrategy package, see “Deployment resource kit.”

For more information about preparing a miniroot for use in deploying ProLiant server and blades, see “Miniroot preparation.”

JumpStart

JumpStart configuration for ProLiant servers is similar to configuration of other vendor platforms, with the following requirements that are specific to ProLiant:

- Drivers—Most ProLiant servers require certain drivers that are not included with Solaris. In addition to having these drivers in the miniroot, you must also install the drivers to the target disks during deployment. This installation is accomplished using JumpStart.
JumpStart offers several mechanisms for installing additional packages not included with Solaris. For more information, see *Solaris 10 11/06 Installation Guide: Custom JumpStart and Advanced Installations* at <http://docs.sun.com/app/docs/doc/819-6397>.
- Rapid Deployment Pack—When using Rapid Deployment Pack, the boot disk is modified before starting the operating system installation to facilitate the booting of the miniroot. After the installation is booted, this modification is no longer necessary. When using Rapid Deployment Pack, the JumpStart profile must be configured to remove this modification.

For more information about configuring JumpStart for ProLiant deployments, see “JumpStart preparation.”

iLO virtual media

Virtual media support, part of the iLO Advanced feature set, provides administrators with a virtual diskette drive, USB key drive, and a virtual CD/DVD drive that connect through the network to the managed server, as if they were physically connected to each other.

Using a custom-built bootable ISO, a Solaris installation can be booted remotely with virtual media, without the use of a PXE server. This ability to boot remotely can be useful in environments where PXE cannot be used or is otherwise not available.

For more information about iLO, see <http://www.hp.com/go/ilo>. For more information about creating a custom ISO image, see “Custom bootable ISO image creation.”

Rapid Deployment Pack

Rapid Deployment Pack is a server deployment solution that facilitates the installation, configuration, and deployment of high volumes of servers through either a GUI-based or a web-based console, using either scripting or imaging technology. Typically used to deploy Microsoft® Windows® and Linux, Rapid Deployment Pack can also be used to initiate Solaris deployment.

Using Rapid Deployment Pack provides the following advantages:

- Rapid Deployment Pack consolidates deployment management of multiple operating systems to a single tool.
- Rapid Deployment Pack can automatically configure the hardware, including Smart Array logical volumes.
- Rapid Deployment Pack has a simple drag-and-drop interface that enables you to select from multiple JumpStart profiles based on intended server usage.

For more information about Rapid Deployment Pack, see <http://www.hp.com/go/rdp>. For more information about using Rapid Deployment Pack to deploy Solaris, see “Rapid Deployment Pack server preparation.”

Flash archives

Flash archives can be used in ProLiant deployment. The primary consideration when using flash archives with ProLiant servers is to ensure that all the required drivers are included in the archive.

To ensure that all required drivers are included, before creating the archive on the master system, install all the drivers required by all the intended clone systems, even if the master system does not have any of those devices or otherwise does not need the drivers. When the flash archive is created, all the installed drivers (whether or not they are in use) are included.

Deployment resource kit

The Solaris on ProLiant Deployment Resource Kit contains the following tools and examples for use in deploying Solaris on ProLiant servers:

- `HPQnetstrategy`—The `HPQnetstrategy` package enhances the miniroot to allow automatic network configuration when the miniroot is not booted from the network. It is required for deployments initiated by Rapid Deployment Pack and custom ISO boot.
- Example Rapid Deployment Pack job—This example of an Rapid Deployment Pack job for deploying Solaris can be imported into a deployment server. When imported, the job must be customized to fit the intended environment. For more information, see “Rapid Deployment Pack server preparation.”
- JumpStart examples—Several sample JumpStart configuration files are included.

This resource kit is available from <http://www.hp.com> on the “Software and Drivers” page associated with each ProLiant server model.

For more information, see the `README` file in the kit.

Deployment planning

Solaris installations proceed through three phases: boot, configuration, and software installation. Each phase offers flexibility in implementation and can be customized for a variety of environments.

This chapter describes several common methods of implementing each phase and provides guidance on selecting strategies that meet the requirements of most organizations.

This section describes the planning of deployment strategies where none exist. If ProLiant servers are being integrated into an existing Solaris deployment environment, see “Deploying ProLiant servers,” “Miniroot preparation,” and “JumpStart preparation.”

Selecting a deployment strategy

For purposes of this guide, a complete deployment strategy consists of a method of booting and delivering the configuration and software packages. Use the following sections to select a deployment strategy for the target environment.

Boot methods

PXE-based booting:

- Can be used in Solaris-only environments because Solaris can provide all the required services
- Can also be used to deploy other x86-based platforms that are not ProLiant platforms
- Can be used with existing x86 or Scalable Processor Architecture (SPARC) deployment environments
- Manages deployment from the DHCP server
- Requires a PXE (DHCP/TFTP) server

Rapid Deployment Pack-initiated deployment:

- Can be used in mixed-operating system ProLiant environments
- Can be used to integrate Solaris deployments into existing ProLiant Windows and Linux deployment environments
- Can be used to automate Smart Array configuration during deployment
- Can be used with an existing x86- or SPARC-hosted JumpStart server
- Manages deployment from the Rapid Deployment Pack deployment console
- Requires an Rapid Deployment Pack deployment server

Custom ISO-based booting:

- Can be used when neither PXE nor Rapid Deployment Pack are available nor desirable
- Can be used with existing SPARC deployment environments that do not require PXE boot services
- Can be used in Solaris-only environments
- Can be performed remotely using iLO virtual media
- Manages deployment from the physical or virtual system console
- Does not require a PXE or Rapid Deployment Pack server

Configuration methods

For Solaris deployments, use of JumpStart profiles and *sysidcfg(4)* files is the recommended method of supplying installation configuration for automated deployments.

Software delivery methods

- NFS—The traditional method of delivering the Solaris software packages to clients. The packages are provided in the same form as they appear on the Solaris media. Additional packages that are not a part of the Solaris distribution, such as the Smart Array driver, can also be delivered through NFS. Setting up an NFS installation image offers flexibility in the variety of installation configurations that a single image can support.
- Flash Archive—A portable, flexible image of a reference installation. A flash archive enables you to clone a master system to any number of additional systems. The clones can be made to mirror not only the Solaris operating environment and configuration of the master system, but also any additional third-party applications that might be necessary. In addition, differential archives can be employed to simplify clone upgrades.

Flash archives are useful for environments in which a large number of systems must be almost identically deployed, including software that is not part of the Solaris distribution. For more information, see *Solaris 10 11/06: Solaris Flash Archives (Creation and Installation)* at <http://docs.sun.com/app/docs/doc/819-6398>.

Flash archives can be used with any of the boot methods previously described.

Implementing a PXE boot-based deployment

PXE boot-based deployments require the following:

- Install server
- JumpStart server
- Modified miniroot
- DHCP/PXE server

Use the following procedure outline as a basis for implementing PXE boot-based deployments in the target environment:

1. Prepare the install server as described in “Install server preparation.”
2. Prepare the miniroot as described in “Miniroot preparation.”
3. Prepare the JumpStart server as described in “JumpStart preparation.”
4. Prepare the DHCP/PXE server as described in “DHCP/PXE preparation.”
5. To deploy Solaris using PXE boot:
 - a. Add the client to the install server as described in “DHCP/PXE preparation.”
 - b. PXE boot the client by booting the server and pressing F12 when prompted during POST.

Implementing an Rapid Deployment Pack -initiated deployment

Rapid Deployment Pack -initiated deployments require the following components:

- Install server
- JumpStart server
- Modified miniroot
- Rapid Deployment Pack deployment server

Use the following procedure outline as a basis for implementing Rapid Deployment Pack-initiated deployments in the target environment:

1. Prepare the install server as described in “Install server preparation.”
2. Prepare the miniroot as described in “Miniroot preparation.”
3. Prepare the JumpStart server as described in “JumpStart preparation.”
4. Prepare the Rapid Deployment Pack deployment server as described in “Rapid Deployment Pack server preparation.”
5. To deploy Solaris using Rapid Deployment Pack, drag-and-drop the **Solaris deployment** job onto the target client.

Implementing a custom ISO boot-based deployment

Custom ISO boot-based deployments require the following components:

- Install server
- JumpStart server
- Modified miniroot
- Custom ISO image

Use the following procedure outline as a basis for implementing custom ISO boot based deployments in the target environment:

1. Prepare the install server as described in “Install server preparation.”
2. Prepare the miniroot as described in “Miniroot preparation.”
3. Prepare the JumpStart server as described in “JumpStart preparation.”
4. Prepare the GRUB menu as described in “GRUB preparation.”
5. Prepare the custom ISO image as described in “Custom bootable ISO image creation.”
6. To deploy Solaris using a custom ISO, boot the client from the ISO through either physical media or iLO virtual media.

Deployment tasks

This chapter describes common deployment tasks. For information about using one or more of these tasks to construct a complete deployment environment, see “Deployment planning.”

In the examples and instructions that follow, file path references often end with `.../subpath`. Replace the ellipsis (...) with the correct path for the current environment. For example, in the following path:

```
.../sol-10-u3-ga-x86-v1.iso
```

replace the ellipsis (...) with a path to the directory that contains the ISO images for the Solaris 10 11/06 installation media.

Install server preparation

This section describes a procedure for preparing an install server, which provides the Solaris installation media to installing clients as part of a network-based deployment.

This procedure is also described in *Solaris 10 11/06 Installation Guide: Network-Based Installations* at <http://docs.sun.com/app/docs/doc/819-6395>, and Solaris man pages (for example, `setup_install_server(1M)`).

Prerequisites for preparing an install server

To prepare a Solaris install server, the following prerequisites must be met:

- Solaris 10 media—This media is available from Sun. The install server can be prepared from the ISO images; they do not need to be burned to physical media.
- System running Solaris—The install server must be a system running Solaris.
- Disk space—Enough disk space must be available on the install server to store the contents of the installation media (approximately 3 GB to 5 GB).

Preparing the install server

In the following procedure, the installation media is assumed to be mounted at `/mnt`, and the installation media root directory is `/export/home/install`. Substitute the installation media as necessary to fit the target environment.

1. Ensure you have met all the requirements for installation and have all the necessary tools available for installation. For more information, see “Prerequisites for preparing an install server.”
2. Mount the DVD or the first CD on the install server.
 - If using physical media (or iLO virtual media), insert or connect the media. Solaris automatically mounts the disc. The autoselected mount point can be determined from the output of, for example, `df(1)`. If Solaris does not automatically mount the disc, it can be manually mounted using `mount(1M)`. To gather more information regarding your media drive, you can enter the following:

```
# rmformat -l
```
 - If using ISO images, mount the DVD or the first CD by entering the following:

```
# lofiadm -a ../sol-10-u3-ga-x86-v1.iso
/dev/lofi/1
# mount -F hsfs /dev/lofi/1 /mnt
```
3. Copy the contents of the DVD or the first CD to disk by entering the following:

```
# cd /mnt/Solaris_10/Tools
# ./setup_install_server /export/home/install
```
4. Unmount the DVD or the first CD.
 - If using physical or iLO virtual media, unmount the media using `eject(1)` (if automounted) or `umount(1M)` (if manually mounted).
 - If using ISO images, unmount the DVD or the first CD by entering the following:

```
# umount /mnt
# lofiadm -d /dev/lofi/1
```
5. If using CD media, copy the remaining CDs. For each additional CD:
 - a. Mount the next CD or ISO image.
 - b. Copy the contents of the CD to disk by entering the following:

```
# cd /mnt/Solaris_10/Tools
# ./add_to_install_server /export/home/install
```
 - c. Unmount the CD.
6. If necessary, share the installation media root directory through NFS for client access.
 - a. Verify that the directory is shared:

```
# share | grep /export/home/install
```
 - b. If the installation media root directory is not listed, or if it is not shared with `anon=0`, then:
 - i. Add the following line to `/etc/dfs/dfstab`:

```
share -F nfs -o ro,anon=0 /export/home/install
```
 - ii. Activate the NFS share:

```
# shareall
```

Miniroot preparation

This section describes the procedure for customizing a miniroot and provides an example set of modifications that are typically required for ProLiant servers.

Prerequisites for preparing a miniroot

To prepare a miniroot for ProLiant server deployments, the following prerequisites must be met:

- System must be running Solaris 10 1/06 or higher—An x86-based system running Solaris is required for use as the staging system.
- Disk space on the staging system—Disk space must be approximately two to three times the size of an uncompressed miniroot (approximately 300 MB to 500 MB).
- Miniroot—The miniroot you are modifying must be available to the staging system. A stock Solaris miniroot is available on the media (DVD or the first CD), and under the installation media root directory on the install server as `.../boot/x86.miniroot`.
- Modifications—The set of packages you are installing, or other modifications, must be available from the staging system. For a list of required modifications for ProLiant servers, see “Deploying ProLiant servers.”

Preparing a miniroot

A general procedure for customizing a miniroot is provided here. An example for ProLiant servers is provided in the following section.

1. Unpack the miniroot to a temporary directory (in this example, `/var/tmp/mr`):

```
# /boot/solaris/bin/root_archive unpack ../x86.miniroot
/var/tmp/mr
```
2. The miniroot is unpacked and expanded under `/var/tmp/mr` (the root of the Solaris instance), and can be modified as necessary. For example, packages can be installed by adding a `-R /var/tmp/mr` option to `pkgadd(1M)`.

NOTE: The stock miniroot does not contain a record of which Solaris packages are already installed. As a result, `pkgadd(1M)` can report that dependencies have not been met when installing additional packages. You must verify that dependencies are met. However, in all the specific examples provided in this guide, you can ignore dependency warnings from `pkgadd(1M)` because the stock miniroot meets all such dependencies.

3. When modifications are complete, repack the modified miniroot under `/var/tmp/mr` into `/var/tmp/x86.miniroot.new`:

```
# /boot/solaris/bin/root_archive pack /var/tmp/x86.miniroot.new \
> /var/tmp/mr
```

`/var/tmp/x86.miniroot.new` is the modified miniroot, packed and ready for use.

Repacking the Miniroot might take several minutes to complete.

NOTE: There is a known issue with Solaris 10 10/08 for which a miniroot with a name other than `x86.miniroot` cannot be unpacked after creation. If you need to make modifications to a custom miniroot, you must temporarily rename it to `x86.miniroot` to use the `root_archive` tool.

NOTE: For Solaris 5/09, a known issue prevents drivers from being installed in the miniroot. If disks attached to a HP Smart Array Controller are not available during installation, see the procedure outlined in the Troubleshooting section “Hard Drives Not Seen After CPQary3 Added to Miniroot” to resolve this issue.

Example: miniroot preparation

The following procedure is an example of a miniroot preparation for ProLiant servers.

1. Unpack the stock Solaris miniroot:

```
# /boot/solaris/bin/root_archive unpack ../boot/x86.miniroot \  
> /var/tmp/mr
```

The miniroot is expanded under `/var/tmp/mr`.

2. Install the Smart Array driver into the miniroot, as described in the *HP Smart Array Controller Driver for Solaris 10 User Guide* at ftp://ftp.compaq.com/pub/softlib2/software1/pubsw-windows/p308167783/v40878/HP_Smart_Array_Controller_Driver_User_Guide.pdf. However, do not repack the miniroot at this time.

3. Install the Broadcom NetXtreme II driver into the miniroot. Choose the version of the driver (gldv2 or gldv3) which is appropriate for your deployment:

```
# cd /var/tmp  
# gunzip -c ../BRCMbnx-x.y.z-solaris10-i386.tar.gz | tar vxf -  
# cd BRCMbnx-x.y.z-solaris10-i386/bnx/gldvN  
# pkgadd -R /var/tmp/mr -d `pwd`/BRCMbnx.pkg BRCMbnx
```

4. For Rapid Deployment Pack or custom ISO images, install the HPQnetstrategy package:

```
# cd /var/tmp  
# gunzip -c ../HPQdeploykit-1.0.0-solaris10-i386.tar.gz | tar vxf -  
# cd HPQdeploykit-1.0.0-solaris10-i386/tools  
# pkgadd -R /var/tmp/mr -d `pwd`/HPQnetstrategy.pkg HPQnetstrategy
```

5. Repack the miniroot:

```
# cd /var/tmp  
# /boot/solaris/bin/root_archive pack  
/var/tmp/x86.miniroot.proliant \  
> /var/tmp/mr
```

The new miniroot `/var/tmp/x86.miniroot.proliant` is now ready for use with ProLiant servers.

JumpStart preparation

This section describes some prerequisites for preparing JumpStart that are specific to ProLiant. For more information about Solaris JumpStart, see *Solaris 10 11/06 Installation Guide: Custom JumpStart and Advanced Installations* at <http://docs.sun.com/app/docs/doc/819-6397>.

This section also includes descriptions of a JumpStart configuration, requirements specific to ProLiant, and a complete configuration example.

Overview of JumpStart preparation

JumpStart automates Solaris deployments by providing configuration information for which the Solaris installer would ordinarily interactively prompt the user.

A basic JumpStart configuration consists of the following components:

- `rules`—The `rules` file provides instructions for selecting a JumpStart profile based on various user-selectable criteria such as platform architecture, system model, and so on, as well as custom criteria.
- Profile—A JumpStart profile file provides certain installation configuration information such as file system layout, disk partitioning, and which packages are to be installed.
- `sysidcfg(4)`—A `sysidcfg` file provides the system identification information such as root password, networking parameters, time zone, and so on.
- Begin and finish scripts—These are optional scripts that run before and immediately after installation and can be customized to perform pre- and post-installation configuration.

The `rules` file, `profile` file, and begin and finish scripts must be in the same directory, and are typically made available to clients through NFS.

Prerequisites for JumpStart preparation

This section describes ProLiant-specific prerequisites for installing JumpStart.

Additional required packages

Some drivers that are not included on the Solaris media are usually required for proper use of Solaris on ProLiant servers. These drivers include the Smart Array driver and the Broadcom NetXtreme II network driver. For automated deployments, provisions must be made for these drivers to be installed on the system during operating system installation. To make these provisions, perform either of the following steps:

- Use the `package` profile keyword to install the additional packages through NFS. See “Example: JumpStart preparation.”
- Use a finish script to install the additional packages using `pkgadd(1M)`. For a complete example, see the Sun documentation.

Rapid Deployment Pack

When deploying Solaris using Rapid Deployment Pack, a small, non-Solaris partition is created on the boot volume to facilitate booting the installation. By default, Solaris avoids disks with existing partitions, so this temporary partition must be deleted. This deletion can be accomplished using the `fdisk` profile keyword to delete all type 0x83 partitions. You must create a new Solaris partition to replace the 0x83 partitions. For example:

```
fdisk rootdisk 0x83 delete
fdisk rootdisk solaris all
```

Example: JumpStart preparation

This section provides a complete JumpStart example. In this example, the profile is named “proliant,” and the entire Solaris distribution is installed, along with the `CPQary3` package. Optional configuration for use with Rapid Deployment Pack is also listed. If installing to disks attached to a Smart Array controller, the root disk must be SCSI target 0, LUN 0 (that is, `cxt0d0`).

NOTE: The example `sysidcfg(4)` file contains entries that are new as of Solaris 10 11/06 and might not be supported by previous releases.

1. Prepare the configuration files as follows:

- Contents of the directory structure:

```
/jumpstart/
/jumpstart/rules
/jumpstart/proliant
/jumpstart/sysidcfg
```

- Contents of the `rules` file:

```
arch i386 - proliant -
```

- Contents of the `proliant` file:

```
install_type initial_install
system_type standalone
partitioning default
cluster SUNWCXall
package CPQary3 add nfs://jssip/jumpstart/pkg
```

where `jssip` is the IP address of the JumpStart server.

If using Rapid Deployment Pack, also include the following lines:

```
fdisk rootdisk 0x83 delete
fdisk rootdisk solaris all
```

- Contents of the `sysidcfg` file:

```
system_locale=en_US.UTF-8
name_service=none
network_interface=if {dhcp protocol_ipv6=no}
security_policy=none
terminal=ibm-pc
timezone=US/Central
timeserver=localhost
nfs4_domain=dynamic
root_password=9E417cFYuXjIY
service_profile=limited_net
```

where `if` is the network interface to configure, such as `bnx0` or `bge0`. You can also use the keyword value 'primary'. In this example, the password for user `root` is `root`.

- 2. Prepare additional packages using `pkgtrans(1)`:

```
# mkdir /jumpstart/pkg
# pkgtrans ../CPQary3-x.y-solaris10-i386/CPQary3.pkg \
> /jumpstart/pkg CPQary3
# pkgtrans ../BRCMbnx-x.y.z-solaris10-i386/bnx/BRCMbnx.pkg \
> /jumpstart/pkg BRCMbnx
# pkginfo -d /jumpstart/pkg
system BRCMbnx Broadcom NetXtreme II Gigabit Ethernet Adapter
driver
system CPQary3 Third Generation Array Controllers
#
```

- 3. Validate the configuration using the check script from the Solaris media (or install server):

```
# cd /jumpstart
# ../Solaris_10/Misc/jumpstart_sample/check
Validating rules...
Validating profile proliant...
The custom JumpStart configuration is ok.
#
```

The check script creates the `rules.ok` file, which is required by JumpStart.

- 4. If not already performed, share the configuration through NFS for client access:

- a. Verify that the directory is shared:

```
# share | grep /jumpstart
```

- b. If the JumpStart directory is not listed, or if it is not shared with `anon=0`, then:

- i. Add the following line to `/etc/dfs/dfstab`:

```
share -F nfs -o ro,anon=0 /jumpstart
```

- ii. Activate the NFS share:

```
# shareall
```

DHCP/PXE preparation

DHCP administration is a complex subject, much of which is beyond the scope of this guide. However, PXE booting is an effective means of deploying Solaris. This section provides instructions for the quick setup of a Solaris DHCP server for use in PXE-booting network clients. For more information, see *System Administration Guide: IP Services* at <http://docs.sun.com/app/docs/doc/816-4554> and *add_install_client(1M)*. For more information about PXE, see *Preboot Execution Environment (PXE) Specification Version 2.1* at <http://download.intel.com/design/archives/wfm/downloads/pxespec.pdf>.

DHCP server configuration for PXE booting of ProLiant servers can be similar to other x86-based servers. There are no DHCP configuration requirements specific to ProLiant.

The following procedure is for the Solaris DHCP server software. If you are using other DHCP server software, see the documentation for that software.

Prerequisites for DHCP/PXE preparation

To prepare a DHCP/PXE server, the following prerequisites must be met:

- X Windows—Use of the Solaris DHCP Manager, which is documented in this guide, requires X Windows. For information about configuring the DHCP server from the command line, see *System Administration Guide: IP Services* at <http://docs.sun.com/app/docs/doc/816-4554>.
- Install server is a DHCP server—The following instructions assume that the install server is also the DHCP server, and that the media has already been copied to disk as described in “Install server preparation.” For information about configuring a DHCP server as a boot-only server, see *add_install_client(1M)*. These instructions also assume that this server is on the same subnet as its intended clients, and that the network is configured for the DHCP server. If this network has not been configured, you can configure it by using *sys-unconfig*, and restarting the server.
- Modified the miniroot, if required—For more information, see “Miniroot preparation.”

Preparing the DHCP/PXE server

1. Configure the DHCP server as follows:
 - a. Start the DHCP Manager by entering the following command:

```
# /usr/sadm/admin/bin/dhccpmgr
```

NOTE: If the DHCP Manager crashes during setup, the manager might be missing network device configuration. You can configure the network devices using *sys-unconfig*.

- b. If starting with no configuration, `dhccpmgr` prompts you for the type of configuration, either DHCP server or BOOTP relay. Select **DHCP server** to start the DHCP Configuration Wizard.
 - c. In the wizard, choose to store the DHCP configuration data as text files, and then click the right arrow.
 - d. Specify a location to store the configuration data.
 - e. Choose the option to not manage host records.
 - f. Enter a lease length, and then choose to enable clients to renew their leases.
 - g. Do not specify a DNS domain.
 - h. Select a network from the list, or enter a network address and subnet mask.
 - i. Specify local area network as the network type, and then choose to use the router discovery protocol.
 - j. Do not configure an NIS domain.
 - k. Do not configure an NIS+ domain.
 - l. Verify the configuration, and then click **Finish** to complete the server configuration.
 - m. Start the Address Wizard.
 - n. Enter the number of IP addresses to add.
 - o. Enter the DHCP server name and the first IP address in the range.
 - p. For the configuration macro, select the DHCP server name option.
 - q. Select the dynamic lease type.
 - r. Verify the configuration, and then click **Finish** to complete the address configuration.
2. If a modified miniroot is to be used, replace `x86.miniroot` under `.../Solaris_10/boot/` with the custom miniroot, and save a copy of the original in the process.

The preceding procedure enables the DHCP server, but does not enable network booting to Solaris installations. To enable network booting for installations, add installation clients by performing the following steps:

3. Use `add_install_client(1M)` to add the client configuration to the DHCP server:

```
# cd .../Solaris_10/Tools
# ./add_install_client -d -e ethmac \
> -s installpath \
> -c jsnfspath \
> -p idsnfspath \
> -t mrpath \
i86pc
```

where:

- *ethmac* is the Ethernet MAC address of the client, in the form aa:bb:cc:dd:ee
 - *installpath* is the NFS path to the Installation Server, such as:
10.10.10.10:/export/home/install
 - *jsnfspath* is the NFS path to the JumpStart configuration directory, such as:
10.10.10.10:/jumpstart
 - *idnfspath* is the NFS path to the directory that contains the *sysidcfg(4)* file, such as:
10.10.10.10:/jumpstart
 - *mrpath* is the local directory of the miniroot, such as:
/export/home/install/boot
4. After adding the installation client, *add_install_client* prints the modifications that must be made to the DHCP server configuration to PXE boot the client. For the Solaris 10 1/06 release and later, only the `BootFile` and `BootSrvA` options are required. Use *dhtadm(1M)* to configure the server as follows:

```
# /usr/sbin/dhtadm -A -m "macname" \  
> -d ':BootSrvA=ipaddr:BootFile=path:'
```

where *macname*, *ipaddr*, and *path* are the values for the macro name, boot server IP address, and boot file that were printed by *add_install_client*.

5. For Solaris 10 10/08, *add_install_client* can produce invalid `menu.lst` files for GRUB. You might need to modify the `menu.lst` file. These files are stored in the `/tftpboot` directory, in the format `menu.lst.dhcpserveridentifier`, where *dhcpserveridentifier* is the identifier produced by *add_install_client* for the DHCP Server above. If the contents of this file are in the following format:

```
default=0  
timeout=30  
title Solaris_10 Jumpstart  
  
kernel /I86PC.Solaris_10-1/multiboot kernel/unix -B -install,  
install_config=jsnfspath,sysid_config=idnfspath,install_media=  
installpath,install_boot=nfsserver:mrpath  
  
module /I86PC.Solaris_10-1/x86.miniroot
```

You will need to modify the kernel line in the following manner:

```
kernel /I86PC.Solaris_10-1/multiboot kernel/unix - install -B  
install_config=jsnfspath,sysid_config=idnfspath,install_media=  
installpath,install_boot=nfsserver:mrpath
```

If you are using a custom miniroot with a name different than `x86.miniroot`, you must modify the module line in the following manner:

```
module /I86PC.Solaris_10-1/customminirootname
```

The DHCP/PXE server configuration is complete.

Rapid Deployment Pack server preparation

This section describes how Rapid Deployment Pack can be used to deploy Solaris on ProLiant servers, and explains the Rapid Deployment Pack deployment server setup required.

Overview of Rapid Deployment Pack server preparation

Deployments of Solaris based on Rapid Deployment Pack are modeled after Rapid Deployment Pack deployments of Linux.

For Linux deployments, the target server PXE boots from the Rapid Deployment Pack deployment server to the Linux preinstall environment. This environment is used for pre-installation configuration of the hardware, such as updating BIOS settings and creating Smart Array logical volumes. After pre-installation configuration, a small ext2-type file system is created on the boot volume, and GRUB, the Linux kernel, and the Linux initrd are placed on the partition. Rapid Deployment Pack then instructs the server to boot off of the disk, which loads Linux, and the installation proceeds from a Linux KickStart server.

Using Rapid Deployment Pack to deploy Solaris is similar to using Rapid Deployment Pack to deploy Linux. The Linux pre-install environment is used for hardware configuration and to install GRUB on the boot volume. However, instead of installing a Linux kernel and initrd, the Solaris multiboot program and installation miniroot are installed, and an appropriate GRUB configuration file is created. After the server reboots off of the disk, Solaris loads and the installation proceeds from a Solaris JumpStart server.

NOTE: At the start of the Solaris installation, the temporary Linux partition remains on the boot volume. By default, Solaris avoids existing partitions. Remove this partition using the appropriate JumpStart profile commands. For an example, see “JumpStart preparation.”

Prerequisites for Rapid Deployment Pack server preparation

This section specifically covers deployment server configuration. Other components are required for a complete deployment solution based on Rapid Deployment Pack. For more information, see “Implementing an Rapid Deployment Pack -initiated deployment.”

The following resources are required to configure an Rapid Deployment Pack deployment server for Solaris deployments:

- Rapid Deployment Pack Deployment Server—An Rapid Deployment Pack 3.1 or greater deployment server is required.
- Deployment Resource Kit—The Deployment Resource Kit contains a sample Rapid Deployment Pack job and associated scripts. For more information, see “Deployment resource kit.”
- Modified miniroot—This miniroot must include the HPQnetstrategy package. For more information, see “Miniroot preparation.”
- Solaris multiboot program—The Solaris multiboot program is available on the media (DVD or the first CD), and under the installation media root directory on the install server as `.../boot/multiboot`

Configuring the Rapid Deployment Pack deployment server

1. Copy the example script `examples/rdp/sol.sh` from the resource kit to `...\\lib\\bin32\\linux\\` on the deployment server. This script is called by the Create Boot Environment task script in the Solaris Rapid Deployment Pack job.
2. Create a new directory under `...\\lib\\osdist` on the deployment server to store the Solaris multiboot program and miniroot, for example `...\\lib\\osdist\\sol110_1106`. Copy the Solaris multiboot program and the modified miniroot into the new directory. The multiboot program and miniroot must be named `multiboot` and `x86.miniroot`, respectively.

3. Import the example job included in the resource kit (`examples/rdp/deploy-sol.bin`) to the deployment server console.
4. Modify the script in the Create Boot Environment task in the newly imported example job to match the target environment.

GRUB preparation

This section describes GRUB menu configuration file preparation for use in deploying Solaris. For more information about GRUB, see *GRUB and the Solaris 10 1/06: The New Bootloader for x86 Platforms* and *GNU GRUB Manual 0.97* at

http://www.gnu.org/software/grub/manual/html_node/index.html.

Overview of GRUB preparation

When GRUB loads, it scans for a configuration file that usually contains the definition of the menu that GRUB should display. Each menu item is associated with a series of GRUB commands that GRUB executes when that menu option is selected. For Solaris, the GRUB configuration file is usually named `menu.lst`.

The easiest way to create a custom configuration file is to use the `menu.lst` from the Solaris media as a template. This list is on the DVD or the first CD, or under the installation media root directory on the install server, as `.../boot/grub/menu.lst`. Each title line defines a menu option, and the commands that follow the title line are the commands that GRUB executes when that menu option is selected.

NOTE: For commands in `menu.lst`, each command list is implicitly terminated with a `boot` command. When entering commands manually at the GRUB command line, the `boot` command must be explicitly given.

To boot a Solaris miniroot, the Solaris multiboot program and miniroot module must be loaded. To boot into a network deployment, the proper boot options for installation media location and JumpStart configuration must be provided.

The following sections describe the various boot options and provide examples.

Booting Solaris from GRUB

This section provides an overview of and describes the options for booting Solaris from GRUB.

Synopsis

```
kernel pathtomultiboot kerneltoboot [options]  
module pathtominiroot
```

- *pathtomultiboot*—Path to the multiboot program. This path is relative to the boot media, either CD or network (TFTP). When booting from CD, this must be a full path.
- *pathtominiroot*—Path to the miniroot. This path is relative to the boot media, either CD or network (TFTP). When booting from CD, this must be a full path.
- *kerneltoboot*—Path to the kernel that multiboot should load. Unlike the other paths, this path is relative to the miniroot (multiboot loads the kernel from the miniroot, not the boot media). When booting a miniroot, the value of *kerneltoboot* is usually `kernel/unix`.

Options

The *options* take this form:

```
[- install] [-B param1=value1[,param2=value2[...]]]
```

The `- install` option must be specified for noninteractive JumpStart installations.

The following parameters can be specified following `-B`:

- *install_media*—Specifies an NFS mount path to the installation media root directory. For example, “10.10.10.10:/export/home/install.” This path must map to the same directory that was passed to *setup_install_server(1M)* when the media was copied to disk.
- *install_config*—Specifies an NFS mount path to a JumpStart configuration directory. For example, a parameter value of 10.10.10.10:/jumpstart instructs the installer to use the rules and profiles found in /jumpstart on an NFS server with IP address 10.10.10.10.
- *sysid_config*—Specifies an NFS mount path to a directory containing the *sysidcfg(4)* file to use (which must itself be named *sysidcfg*). For example, a parameter value of 10.10.10.10:/jumpstart instructs the installer to use the file /jumpstart/sysidcfg from the NFS server with IP address 10.10.10.10.

HPQnetstrategy

The `HPQnetstrategy` package, when installed in the miniroot, is used to direct the Solaris installation to configure the network even when the miniroot is not booted from the network, such as when booting from an ISO image or when using Rapid Deployment Pack.

To instruct `HPQnetstrategy` on the desired network configuration, specify a *netstrategy* boot parameter in the same manner as the installation parameters described previously. Parameter usage takes the following form:

```
netstrategy=if:strategy
```

- *if*—The network interface that must be used, such as `bnx0`. The *if* interface can also be the value `any` to instruct `HPQnetstrategy` to search for a network interface that can be configured using DHCP.
- *strategy*—The configuration method that must be used. The only supported value for *strategy* is `dhcp`.

Examples: booting Solaris from GRUB

NOTE: In the following examples, the kernel line is shown broken into multiple lines. In practice, the parameters must all be on the same line, and there must not be any spaces in the comma-separated `-B` parameter list.

- For a JumpStart installation when PXE booting:

```
title Solaris 10 JumpStart Install (PXE boot)
kernel multiboot kernel/unix - install
  -B install_media=10.10.10.10:/export/home/install,
  install_config=10.10.10.10:/jumpstart,
  sysid_config=10.10.10.10:/jumpstart,
  install_boot=10.10.10.10:/export/home/install/boot
module x86.miniroot
```

- For a JumpStart installation with ISO boot or Rapid Deployment Pack, using the first network interface to receive a DHCP response:

```
title Solaris 10 JumpStart Install (ISO boot/RDP)
kernel /multiboot kernel/unix - install
  -B install_media=10.10.10.10:/export/home/install,
  install_config=10.10.10.10:/jumpstart,
  sysid_config=10.10.10.10:/jumpstart,
  install_boot=10.10.10.10:/export/home/install/boot,
  netstrategy=any:dhcp
module /x86.miniroot
```

- For an interactive network installation, booting from ISO, using a specific network interface (`bnx0` in this example) with DHCP:

```
title Solaris 10 Interactive Install (ISO boot)
kernel /multiboot kernel/unix
  -B install_media=10.10.10.10:/export/home/install,
  netstrategy=bnx0:dhcp
module /x86.miniroot
```

Custom bootable ISO image creation

This section describes a procedure for creating a custom bootable ISO image with a modified miniroot for use in booting a ProLiant server to a network-based, possibly automated Solaris installation.

Prerequisites for creating a custom bootable ISO image

To create and use a custom bootable ISO image, the following prerequisites must be met:

- DHCP Service—Booting a network installation from a custom ISO image requires a DHCP service on the network to provide networking configuration for the client.
- System running Solaris 10 1/06 (or higher)—Use a system running Solaris x86 as the staging system.
- A modified miniroot—If the CD is used to boot to a network-based installation, then the miniroot must include the `HPQnetstrategy` package. For more information, see “Miniroot preparation.”
- Solaris multiboot program—The Solaris multiboot program is available on the media (DVD or the first CD), and under the installation media root directory on the install server as `.../boot/multiboot`.
- GRUB `menu.lst`—If using the directory layout described in the next section, the value for `pathtomultiboot` in `menu.lst` should be `/multiboot` and `pathtominiroot` must be `/x86.miniroot`. For more information, see “GRUB preparation.”

If the CD is used to boot to a network-based installation, then the GRUB “kernel” line must include a directive for `HPQnetstrategy`.

Creating a custom bootable ISO image

To create a custom bootable ISO image:

1. Ensure you have met all the requirements for installation and have all the necessary tools available for installation. For more information, see “Prerequisites for creating a custom bootable ISO image.”
2. Organize the CD as follows (in this example, `/var/tmp/iso` is the root of the CD):

```
# mkdir /var/tmp/iso
# mkdir -p /var/tmp/iso/boot/grub
# cp /boot/grub/stage2_eltorito /var/tmp/iso/boot/grub
# cp .../menu.lst /var/tmp/iso/boot/grub
# cp .../x86.miniroot /var/tmp/iso/x86.miniroot
# cp /boot/multiboot /var/tmp/iso/multiboot
```

3. Create the ISO image using `mkisofs(8)`:

```
# cd /var/tmp
# mkisofs -R -b boot/grub/stage2_eltorito -no-emul-boot \
> -boot-load-size 4 -boot-info-table -o solarisboot.iso iso
```

The resulting ISO image is `/var/tmp/solarisboot.iso`.

Optional configuration tasks

This chapter details optional configuration tasks specific to ProLiant servers.

Enabling iLO virtual serial port

By default, Solaris does not recognize the iLO virtual serial port. If enabled, this virtual serial port can be used as a serial console (by appending `"-B console=ttya"` to the GRUB kernel line). The procedure to enable the virtual serial port can potentially cause configuration issues with Solaris. Review the failsafe mode before configuring the virtual serial port. To enable Solaris to see the iLO virtual serial port, make the following changes to `/kernel/drv/asy.conf`:

1. Comment out (by pre-pending a '#') the "interrupt-priorities" line.
2. Add one of the following lines, depending on which serial port has been designated as virtual in RBSU:

For a virtual COM1 add:

```
name="asy" parent="isa" reg=1,0x3f8,0x8 interrupts=0x4;
```

For a virtual COM2 add:

```
name="asy" parent="isa" reg=1,0x2f8,0x8 interrupts=0x3;
```

Do not add them both (or have both uncommented).

3. Reboot.

Mirroring disks with DiskSuite

Solaris supports disk mirroring through DiskSuite. In this example, `c0t0d0` is mirrored by `c0t1d0`.

1. Inspect the current partition table of `c0t0d0`.

```
Part Tag Flag Cylinders Size Blocks
0 root wm 0 - 812 400.15MB (813/0/0) 819504
1 swap wu 813 - 1333 256.43MB (521/0/0) 525168
2 backup wm 0 - 17659 8.49GB (17660/0/0) 17801280
3 unassigned wm 1334 - 1354 10.34MB (21/0/0) 21168
4 var wm 1355 - 8522 3.45GB (7168/0/0) 7225344
5 usr wm 8523 - 14764 3.00GB (6242/0/0) 6291936
6 unassigned wm 14765 - 16845 1.00GB (2081/0/0) 2097648
7 home wm 16846 - 17658 400.15MB (813/0/0) 819504
```

2. Copy the partition table of the first disk to its mirror.

```
# prtvtoc /dev/rdisk/c0t0d0s2 | fmthard -s - /dev/rdisk/c0t1d0s2
```

3. Create at least two *DiskSuite* state database replicas on each disk. A state database replica contains *DiskSuite* configuration and state information.

```
# metadb -a -f -c2 /dev/dsk/c0t0d0s3 /dev/dsk/c0t1d0s3
```

4. Mirror the root slice.

```
# metainit -f d10 1 1 c0t0d0s0
# metainit -f d20 1 1 c0t1d0s0
```

```
# metainit d0 -m d10
# metaroot d0
```

Use the *metaroot d0* command only on the root slice.

5. Mirror the swap slice.

```
# metainit -f d11 1 1 c0t0d0s1
# metainit -f d21 1 1 c0t1d0s1
# metainit d1 -m d11
```

6. Mirror the var slice.

```
# metainit -f d14 1 1 c0t0d0s4
# metainit -f d24 1 1 c0t1d0s4
# metainit d4 -m d14
```

7. Mirror the usr slice.

```
# metainit -f d15 1 1 c0t0d0s5
# metainit -f d25 1 1 c0t1d0s5
# metainit d5 -m d15
```

8. Mirror the opt slice.

```
# metainit -f d16 1 1 c0t0d0s6
# metainit -f d26 1 1 c0t1d0s6
# metainit d6 -m d16
```

9. Mirror the home slice. Run *metastat metadvice* for information on a metadvice. If you want to delete a metadvice, run *metaclear metadvice*.

```
# metainit -f d17 1 1 c0t0d0s7
# metainit -f d27 1 1 c0t1d0s7
# metainit d7 -m d17
```

10. Update */etc/vfstab* to mount the mirrors after boot. The following is an example of */etc/vfstab* before changes:

```
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#
#/dev/dsk/c1d0s2 /dev/rdisk/c1d0s2 /usr ufs 1 yes -
fd - /dev/fd fd - no -
/proc - /proc proc - no -
/dev/dsk/c0t0d0s1 - - swap - no -
/dev/md/dsk/d30 /dev/md/rdisk/d30 / ufs 1 no logging
/dev/dsk/c0t0d0s5 /dev/rdisk/c0t0d0s5 /usr ufs 1 no ro,logging
/dev/dsk/c0t0d0s4 /dev/rdisk/c0t0d0s4 /var ufs 1 no nosuid,logging
/dev/dsk/c0t0d0s7 /dev/rdisk/c0t0d0s7 /home ufs 2 yes
nosuid,logging
/dev/dsk/c0t0d0s6 /dev/rdisk/c0t0d0s6 /opt ufs 2 yes nosuid,logging
swap - /tmp tmpfs - yes -
```

Make the following changes so that */etc/vfstab* reads as follows:

```
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#
#/dev/dsk/c1d0s2 /dev/rdisk/c1d0s2 /usr ufs 1 yes -
fd - /dev/fd fd - no -
/proc - /proc proc - no -
/dev/md/dsk/d1 - - swap - no -
/dev/md/dsk/d0 /dev/md/rdisk/d0 / ufs 1 no logging
/dev/md/dsk/d5 /dev/md/rdisk/d5 /usr ufs 1 no ro,logging
/dev/md/dsk/d4 /dev/md/rdisk/d4 /var ufs 1 no nosuid,logging
/dev/md/dsk/d7 /dev/md/rdisk/d7 /home ufs 2 yes nosuid,logging
```

```
/dev/md/dsk/d6 /dev/md/rdisk/d6 /opt ufs 2 yes nosuid,logging
swap - /tmp tmpfs - yes -
```

11. Reboot the system.

```
# lockfs -fa
# init 6
```

12. Attach the second submirror to the mirror. This causes the data from the boot disk to be synchronized with the mirrored drive. You can run *metastat* to track the mirroring progress.

```
# metattach d0 d20
# metattach d1 d21
# metattach d4 d24
# metattach d5 d25
# metattach d6 d26
# metattach d7 d27
```

13. Change the crash dump device to the *DiskSuite* swap metadevice (Solaris 7 and later).

```
# dumpadm -d `swap -l | tail -1 | awk '{print $1}'`
```

14. Determine the physical device path of the mirror disk.

```
$ ls -l /dev/dsk/c0t1d0s0
lrwxrwxrwx 1 root root 50 Jan 16 10:20 /dev/rdisk/c0t1d0s0 ->
../../devices/pci@1f,0/pci@1,1/ide@3/dad@1,0:a
```

Replace the major name (e.g. dad) in the physical device path with disk when creating the device alias below.

15. Create a mirror device alias for the mirror disk:

```
# eeprom "nvrarc=devalias mirror
/pci@1f,0/pci@1,1/ide@3/disk@1,0:a"
# eeprom "use-nvrarc?=true"
```

You can also configure the device alias from the Open Boot (ok) prompt:

```
nvalias mirror /pci@1f,0/pci@1,1/ide@3/disk@1,0:a
```

You might want to add the mirror device alias to the boot-device Open Boot parameter in case of a problem with the standard boot device. For instance:

```
# eeprom boot-device
boot-device=disk net
```

In case of an issue with the disk device alias, the system attempts to boot from the mirror device alias and then the network with the following change.

```
# eeprom "boot-device=disk mirror net"
```

From the Open Boot prompt, type boot mirror to boot from the mirror disk.

16. If you only have two internal disks (for example, the boot disk and mirror disk), consider adding the following parameter to */etc/system* to bypass the SVM/DiskSuite quorum rule.

```
set md:mirrored_root_flag = 1
```

Without this parameter, the system will not boot in multiuser mode if half or fewer of its state database replicas are available.

Unmirroring disks with DiskSuite

1. Detach the mirror from its sub-mirror using
`metadetach mirror sub-mirror`
2. Check the status of the mirror using `metastat`.
3. Detach the mirror. `metastat -p` will list the sub-mirrors and the mirrors associated.

4. Detach the mirror:
metadetach

Troubleshooting

This chapter describes common problems and probable causes, known issues, and provides general troubleshooting advice.

To begin troubleshooting deployment problems, review the list of Known Issues below first to see if any issues match the environment and symptoms. If none of the known issues apply, see “Common ” for issue diagnosis. For general troubleshooting advice, see “Basic troubleshooting techniques.”

Known issues

The following is a list of known issues associated with Solaris deployment and ProLiant servers.

Write cache not flushed on shutdown

Solaris does not flush hard drive controller cache to disk on shutdown.

- OS releases affected—Solaris 10 update 5/10
- Hardware affected—HP Third Generation Smart Array Controllers with Battery Backed Write Cache.
- Description—There is a known issue causing Solaris to not properly flush the disk controller caches to disk at shutdown. As a result of this issue, the following error message may be displayed on a ProLiant server running Sun Solaris 10 and configured with a Smart Array Controller equipped with Battery-Backed Write Cache (BBWC):

```
POST Error: 1792 - Drive Array Reports Valid Data Found in Array Accelerator
```

For more information please refer advisory

<http://h20000.www2.hp.com/bizsupport/TechSupport/Document.jsp?lang=en&cc=us&taskId=110&prodSeriesId=1137825&prodTypeId=15351&objectID=c01519161>

- Workaround—Ensure that the cache is properly flushed prior to shutdown. Do this by executing the following command before every shutdown:

```
# swap -d
```

Optionally, shutdown the system using these commands:

```
# init 0  
# halt
```

Solaris stops responding on HP ProLiant DL580 G5 server

Solaris may stop responding, or will respond poorly if a PCI-Express Controller is configured in PCI-Express slots 1, 2, 4, 5, 7, or 8 on a DL580 G5 server.

- OS releases affected—Solaris 10 8/07, 5/08
- Hardware affected—HP ProLiant DL580 G5
- Description—Solaris may stop responding, or will not respond properly during deployment if a PCI-Express Controller, such as a Smart Array P800 is installed in slots 1, 2, 4, 5, 7, or 8. The issue occurs because hot swap is enabled but not supported by the DL580 G5 server. For more

information, please refer to advisory

<http://h20000.www2.hp.com/bizsupport/TechSupport/Document.jsp?lang=en&cc=us&taskId=110&prodSeriesId=3454575&prodTypeId=15351&prodSeriesId=3454575&objectID=c01479945>

- Workaround—Edit `/platform/i86pc/kernel/drv/pcie_pci.conf` and add the entry `"pci-hotplug-type=0"`, then reboot the server.
- For Solaris 10 5/08, the issue is fixed with patch 137122-04.
- This issue is fixed in Solaris 10 10/08.

RAID mode not supported

Solaris 10 operating system will not deploy on an HP ProLiant BL260c G5 or BL280c G6 blade server when the SATA controller is configured in RAID mode.

- OS releases affected—Solaris 10
- Hardware affected— HP ProLiant BL260c G5, BL280c G6
- Description—Cannot install SUN Solaris 10 operating system on HP ProLiant when the SATA controller is configured in RAID mode. Solaris 10 operating system does not have the device driver support for the SATA Controller in RAID mode.
- Workaround—Solaris does not support this controller in RAID mode. Deploy Solaris in IDE mode.

Kernel Panic after disabling PCI devices in RBSU

Solaris stops responding or halts due to kernel panic when one or more PCI devices have been disabled using ROM-Based Setup Utility (RBSU).

- OS releases affected—All
- Servers affected—All supported ProLiant servers
- Description—Disabling PCI devices in RBSU might cause Solaris to stop responding, halt due to kernel panic, or produce other undesirable behavior.
- Workaround—Unless otherwise indicated, HP does not support using RBSU to disable PCI devices while running Solaris. Add-on PCI devices should be removed from the system instead of disabling them in RBSU.

Network configuration fails during interactive installation

Solaris installation fails during network configuration or Solaris is unable to configure network interfaces after the post-installation reboot.

- OS releases affected—Solaris 10 1/06, 6/06, 11/06, and 5/08
- Servers affected—All supported ProLiant servers
- Description—There is a known issue with network setup at installation time, and the installation might fail, or Solaris will be unable to initialize the network after rebooting. This issue occurs if driver updates (DUs) or installation time updates (ITU), such as the Smart Array driver, are added during installation.

This issue is not reported if the extra drivers are installed into the miniroot prior to deployment.

- Workaround—When prompted during the Solaris installation, select *non-networked*. After the installation is complete and the system has rebooted, run `sys-unconfig(1M)`. The system reboots and the network can be configured properly.
- This issue is fixed with Solaris 10 10/08.

iLO boot hang

Solaris becomes unresponsive intermittently, but frequently at boot, just after the Solaris copyright banner appears. If the console is viewed from iLO, the virtual console goes blank. A power cycle or reset is required to restore the system.

- OS releases affected—Solaris 10 6/06
- Servers affected—All supported ProLiant servers
- Description—A change to the Solaris device enumeration during boot that conflicts with iLO was introduced in the Solaris 10 6/06 release. It causes the system to become unresponsive intermittently during boot.

This issue is also documented as Customer Advisory c00775941.

- Resolution—This issue has been corrected in the Solaris 10 11/06 release.
- Workaround—Disable iLO

AMD memory scrubber

System becomes unresponsive after a fixed amount of uptime (usually between 1 and 6 hours), regardless of system load.

- OS releases affected—Solaris 10 1/06 with patches, 6/06, 11/06, 8/07, 5/08, 10/08, 5/09
- Servers affected—HP ProLiant DL385, DL585, BL25p, BL35p, BL45p
- Description—Beginning with the Solaris 10 6/06 release and equivalent kernel patches, Solaris is enabling the AMD™ memory scrubber. However, there is a known issue with AMD processors in which the memory scrubber causes the system to become unresponsive with certain memory configurations. First-generation AMD-based ProLiant servers with 4GB or more of system memory are subject to this issue.
- Workaround— for Solaris 10 8/07 or earlier, disable the memory scrubber by adding the following line to `/etc/system` and then reboot:

```
set cpu\.\AuthenticAMD\.\15:ao_scrub_rate_dram = 0
```
- Workaround— for Solaris 10 5/08 and later, add the following to `/etc/system` and reboot:

```
set mc-amd:mc_no_attach=1
```

SNMP master agent

The System Management Homepage shows no data and other SNMP clients are unable to get data from the Insight Management Agents.

- OS releases affected—Solaris 10 6/06, 11/06, 8/07, 5/08
- Servers affected—All supported ProLiant servers
- Description—A change was introduced with the Solaris 10 6/06 release that causes SNMP subagents, such as the HP Insight Management Agents, to lose communication with the master agent when the system is running in 64-bit mode.

With the Solaris 10 6/06 release, Sun has introduced a 64-bit version of the master agent, which is started if the system is running in 64-bit mode. Prior to this release, only a 32-bit version of the master agent was available. The 64-bit master agent is unable to communicate with any subagent, including the HP Insight Management Agents, which results in SNMP clients, such as the System Management Homepage, being unable to collect MIB data provided by subagents. The 32-bit master agent does not have this issue.

- Workaround— Disable the 64-bit master agent and enable the 32-bit version as follows:

```
# svcadm disable sma
# perl -pi -e
's/\usr\sfw\sbin\snmpd\/usr\sfw\sbin\i86\snmpd/' >
/lib/svc/method/svc-sma
# svcadm enable sma
```

bcme fails to attach

Broadcom gigabit Ethernet NIC fails to attach.

- OS releases affected—Solaris 10 8/07, 5/08, 10/08
- Servers affected—HP ProLiant BL685c, BL685c G5, BL680c, BL480c, BL260c G5
- Description—The Broadcom NetXtreme Gigabit Ethernet Driver (BRCMbcme) v10.4.5 or later successfully installs but fails to attach on a system with NC326i/NC325m (BCM5715S chipset) NIC devices. The “bcme fails to attach” message is seen during the installation of Broadcom NetXtreme Gigabit Ethernet Driver (BRCMbcme) v10.4.5 or later for NC326i and NC325m Network Controllers on Solaris10 8/07, Solaris 10 5/08, and Solaris 10 10/08.
- Workaround—For Solaris 10 8/07, follow these steps.

1. Remove the “bge” entries from the /etc/path_to_inst file.

```
The bge entries should look similar to the following for NC326i
"/pci@0, 0/pci8086, 65e2@2/pci1166, 103@0/pci103c, 703c@4" 0 "bge"
"/pci@0, 0/pci8086, 65e2@2/pci1166, 103@0/pci103c, 703c@4, 1" 1
"bge"
```

For NC325m:

```
"/pci@0, 0/pci8086, 25e2@2/pci8086, 3500@0/pci8086, 3514@1/pci1166, 103
@0/pci103c, 703c@4" 0 "bge"
"/pci@0, 0/pci8086, 25e3@3/pci111d, 801c@0/pci111d, 801c@0/pci1166, 103
@0/pci103c, 170c@4" 2 "bge"
```

2. Reboot with “-r” option:

```
# reboot -- -r
```

3. Run `fmadm faulty` (this will list the faulty/degraded devices)

```
# fmadm faulty
```

4. Repair the faulty devices using the `fmadm` command as follows :

```
# fmadm repair <faulty_device_ids>
```

The `faulty_device_ids` can be obtained from the step 3 above. The faulty PCI device ids should be similar to the following.

For NC326i:

```
"dev:///pci@0, 0/pci8086, 65e2@2/pci1166, 103@0/pci103c, 703c@4"
```

```
"dev:/// pci@0, 0/pci8086, 65e2@2/pci1166, 103@0/pci103c, 703c@4, 1"
```

For NC325m:

```
dev:///pci@0, 0/pci8086, 25e3@3/pci111d, 801c@0/pci111d, 801c@0/pci11
66, 103@0/pci103c, 170c@4, 1
```

```
dev:///pci@0, 0/pci8086, 25e2@2/pci8086, 3500@0/pci8086, 3514@1/pci11
66, 103@0/pci103c, 703c@4
```

- Workaround— For Solaris 10 5/08 and 10/08, the chipset is supported with the native bge driver provided by the patch 140120-01.
- Workaround— For Solaris 10 5/09, the chipset is supported with native bge driver.

System panics and reboots during installation

- OS releases affected—Solaris 10 10/08, 5/09
- Servers affected—HP ProLiant DL485 G1, DL585 G1
- Description—System panics and reboots during the OS installation of Solaris 10 10/08 and Solaris 10 5/09 on HP ProLiant DL385 G1 and DL585 G1.
- Workaround—During OS installation:
 - a. Edit the GRUB menu by pressing the **E** key.
 - b. Edit the kernel entry by pressing the **E** key.
 - c. Add `-kd` at the end of the line, and then press **ENTER**.
 - d. Press **b**. The system will boot to the kdb prompt appears similar to "`[0]>`".
 - e. Type the following entry at the kdb prompt:

```
[0]>lgrp_plat_srat_enable?W 0
```
 - f. Type `:c`.
 - g. Proceed with the installation.
 - h. Repeat the above steps after the server reboots to get to the OS.
- Workaround—After the OS installation:

Become the super user & type the following at the "# " shell prompt:

```
# mdb -w /platform/i86pc/kernel/unix
lgrp_plat_srat_enable?W 0
Press CTRL + d.
# mdb -w /platform/i86pc/kernel/amd64/unix
lgrp_plat_srat_enable?W 0
Press CTRL + d.
```

Embedded Ethernet not configured

- OS Releases affected—Solaris 10 5/08, 10/08, 6/09
- Servers affected—HP ProLiant BL280c G6
- Description—There is no native driver available for the NC362i Dual Port Gigabit Server Adapter in Solaris 10 5/08, 10/08, and 5/09.
- Workaround—After OS installation, install patch 138175-02 to configure the NC362i Dual Port Gigabit Server Adapter.

Hard Drives not seen after CPQary3 added to Miniroot

- OS Releases affected—Solaris 10 5/09
- Description—"Add_drv/Rem_drv Currently Busy; Try Later" message displayed when modifying the miniroot to add drivers to Solaris 10 5/09. Alternately, disks attached to the Smart Array Controller are not available during installation after Smart Array Controller Driver is added to the miniroot. When modifying the miniroot to add drivers to perform a PXE installation of Solaris 10 5/09, the post install script in the package calls `add_drv` to install the driver. The `add_drv` fails and displays the following message:

```
add_drv/rem_drv currently busy; try later
```

This occurs due to the presence of the `/tmp/AddDrEm.lock` file in the miniroot that prevents the driver package from being installed successfully to the miniroot .

- Workaround—Remove file `/tmp/AdDrEm.lck` file from the miniroot before adding the CPQary3 driver package, or any other driver package.
1. Unpack the PXE miniroot to some temporary location such as `/var/tmp/mr`

```
# /boot/solaris/bin/root_archive unpack <SUN Solaris 10
5/09_PXE_image_path>/boot/x86.miniroot /var/tmp/mr
```
 2. Remove `/tmp/AdDrEm.lck` from the PXE miniroot

```
# rm /var/tmp/mr/tmp/AdDrEm.lck
```
 3. Add the CPQary3 driver package

```
# pkgadd -d CPQary3.pkg -R /var/tmp/mr
```

NOTE: Ignore any of the following warnings:

WARNING: The `<SUNWcar>` package "Core Architecture, (Root)" is a prerequisite package and should be installed.

WARNING: The `<SUNWkvm>` package "Core Architecture, (Kvm)" is a prerequisite package and should be installed.

WARNING: The `<SUNWcsr>` package "Core Solaris, (Root)" is a prerequisite package and should be installed.

WARNING: The `<SUNWcsu>` package "Core Solaris, (Usr)" is a prerequisite package and should be installed.

WARNING: The `<SUNWcsd>` package "Core Solaris Devices" is a prerequisite package and should be installed.

WARNING: The `<SUNWcsl>` package "Core Solaris Libraries" is a prerequisite package and should be installed.

4. Pack the PXE image miniroot:

```
#/boot/solaris/bin/root_archive pack <SUN Solaris 10
5/09_PXE_image_path>/boot/x86.miniroot /var/tmp/mr
```

Basic troubleshooting techniques

The following basic tools and techniques are useful for investigating issues with Solaris deployment.

Shell prompt

An interactive shell is the primary tool for troubleshooting deployment issues that occur after the miniroot has booted. In some cases, early failures cause the console to automatically display a shell prompt. Otherwise, a shell can usually be started when necessary.

If X Windows is running, a terminal window can be opened by right-clicking on the background, and then selecting the appropriate item from the menu. If the character mode installer is running, proceed through the prompts until an option to exit appears. Exiting drops the console to a shell prompt.

Network verification

Problems with network deployments can often be traced to errors in network configuration or typographical errors in network paths. The Solaris miniroot contains a variety of tools and utilities that can be used to diagnose such problems.

For example, *ifconfig(1M)* can be used to verify network parameters and *ping(1M)* can be used to test network connectivity to an install server. Utilities such as *df(1M)* or *mount(1M)* can be used to determine which NFS filesystems have been mounted. If an NFS share is unexpectedly not mounted, an attempt to manually mount it can be used to determine why the mount is failing.

GRUB command line

For investigating issues with GRUB, the GRUB command line is often useful. The command line can also be used for fine-tuning commands in `menu.lst` without having to rebuild ISO images or restart Rapid Deployment Pack jobs.

At the GRUB menu, press the `c` key to display the GRUB command line. Under certain circumstances, GRUB might not display a menu and is already in command-line mode.

For more information about using GRUB, see the GRUB documentation at <http://www.gnu.org>.

Kernel debugger

In rare cases, the Solaris kernel debugger can be employed to determine the source of a problem. Some types of configuration problems can result in a kernel panic shortly after the Solaris copyright banner appears. In these cases, Solaris does not produce a crash dump and immediately reboots the system. Often, there is no indication of the nature of the problem. By enabling the kernel debugger, Solaris displays the debugger during a panic, instead of rebooting.

To enable the Solaris debugger, add `-k` to the GRUB kernel line. When the debugger has assumed control following the occurrence of a significant error, the `::msgbuf` command displays the kernel message buffer, the end of which contains the panic string, if any. The `::quit` command reboots the system.

Common issues

This section describes some symptoms encountered frequently during implementation of a deployment environment, along with the possible causes of each. Use this section to determine where troubleshooting efforts must be focused.

System does not boot to GRUB

If PXE booting, verify network connectivity and *dhtadm(1M)* usage, or the DHCP/TFTP server configuration. Verify that the DHCP service is available on the same subnet. Also, verify using RBSU that the correct NIC port is used for PXE boot and the correct Ethernet MAC address was given to *add_install_client(1M)* and *dhtadm*. For more information, see "DHCP/PXE preparation."

If booting from a custom ISO, verify *mkisofs(8)* usage and that `stage2_eltorito` is on the media. For more information, see "Custom bootable ISO image creation."

If booting from Smart Array, verify the boot controller order in RBSU. If installation is complete, verify that the Solaris root disk is SCSI target 0, LUN 0 (that is, `cxt0d0`).

GRUB does not display a menu

Most likely, GRUB is unable to find its configuration file `menu.lst`.

If booting from PXE, verify `add_install_client(1M)` and `dhtadm(1M)` usage. The Ethernet MAC address given to `add_install_client` is significant. For more information, see “DHCP/PXE preparation.”

If booting from a custom ISO, verify that the configuration file is in the proper location on the media (such as `/boot/grub/menu.lst`). For more information, see “Custom bootable ISO image creation.”

GRUB cannot boot Solaris

To boot Solaris, GRUB must be able to access the multiboot program and the miniroot.

If booting from PXE, verify `add_install_client(1M)` usage. For more information, see “DHCP/PXE preparation.”

If booting from a custom ISO, verify that multiboot and the miniroot are on the media and that the correct pathnames are given to GRUB.

If booting a Proliant server blade from a USB drive attached directly to the blade and GRUB is reporting `Cannot mount selected partition`, verify that the drive in use is supported for use with blades.

JumpStart deployment ignores GRUB command line arguments

The installation setup script, `add_install_client`, has produced an invalid `menu.lst`. See “DHCP/PXE preparation” for resolution.

System becomes unresponsive using remote console

Check for applicability of known issue iLO Boot Hang. For more information, see “iLO boot hang.”

If using a serial console (physical or virtual) and GRUB is not seen, verify that a `splashimage` is not specified in `menu.lst`.

If using the browser-based iLO remote console, do not specify a `tty` console. The iLO remote console is the server embedded video adapter.

Rapid Deployment Pack job fails

The sample Rapid Deployment Pack job for Solaris deployment is composed of multiple tasks, most of which are common Rapid Deployment Pack tasks used in other types of deployments. The Create Boot Environment task is unique to Solaris.

If the Create Boot Environment task is failing, use the reported error code or status value to determine the cause. For additional information, see “Rapid Deployment Pack server preparation.”

Error codes include:

- 0—The task script completed successfully.
- 127—Occurs if the task script is unable to find the `sol.sh` script. Verify that `sol.sh` is available on the deployment server.
- 250—The `sol.sh` script was unable to create the GRUB configuration file `grub.conf`.
- 251—The `sol.sh` script was unable to find `multiboot` and/or `x86.miniroot` in `...\osdist\%dist` on the deployment server. Verify that these required files have been put on the deployment server.

- 252—The `sol.sh` script was unable to copy `multiboot` and/or `x86.miniroot` to the temporary disk partition.
- 253—The `sol.sh` script was unable to mount the temporary disk partition.
- 254—The `$dist` parameter was not specified in the Create Boot Environment task script. Verify the setting in the task script.

Installation terminates prematurely

If installation is reporting `No disks found`, verify that the `CPQary3` package is installed in the miniroot (for more information, see “Miniroot preparation”) and use ORCA to verify that at least one logical drive has been created.

If installation is reporting `Unable to configure network interface`, verify that any necessary NIC drivers have been installed in the miniroot. If using JumpStart or a custom ISO, verify that your deployment configuration includes or targets the correct miniroot on the GRUB module line. If using Rapid Deployment Pack or booting a custom ISO, verify that the `HPQnetstrategy` package is installed in the miniroot, and that a `netstrategy` parameter has been given on the GRUB kernel line. For more information, see “Miniroot preparation” and “GRUB preparation.”

If booting a JumpStart installation from iLO virtual media using a custom ISO, and installation is reporting an error similar to `Could not preserve type 79 fdisk partition (c0t0d0)`, but has selected a different root disk, then the partition that is causing the error (`c0t0d0` in this case) is most likely the virtual media. Add a `dontuse` keyword to the JumpStart profile to cause the installation to ignore the virtual media partition.

JumpStart reverts to interactive configuration

If a JumpStart profile and a `sysidcfg(4)` file are provided, but the Solaris installation begins prompting you for information, then it either cannot find one of the files, one of the files has a syntax error, or the configuration is incomplete.

To resolve this issue:

1. Verify network connectivity to the JumpStart server.
2. Verify the NFS pathnames in one of the following locations, as appropriate:
 - `add_install_client(1M)` (if you are using PXE)
 - GRUB (if you are using a custom ISO)
 - Create Boot Environment task script (if you are using Rapid Deployment Pack)
3. Verify that those paths can be mounted on the target server. Ensure that the directory specified by the `sysid_config` parameter contains a file named `sysidcfg`.
4. Verify the syntax of `rules.ok` and the profile file. The `check` script included on the Solaris media verifies `rules` and any profile files it specifies, and creates `rules.ok`.
5. Verify the syntax of `sysidcfg` and verify that all required information is provided. The example provided in this guide is complete for those Solaris releases to which this guide applies.

For more information, see “JumpStart preparation.”

Installation succeeds but server fails to boot

If the system is not booting to GRUB, verify the standard boot order and boot controller order in RBSU. If booting from Smart Array, the Solaris root disk must be SCSI target 0, LUN 0 (that is, `cxt0d0`).

If the system is booting to GRUB, but reboots shortly after the Solaris copyright banner appears, then verify that the `CPQary3` package was installed to the disk during installation (for more information, see “JumpStart preparation”). Another symptom of this problem is the kernel debugger showing that Solaris is panicking with “cannot mount root.”

Installation succeeds but no network availability after reboot

If Solaris is unable to configure network interfaces after the post-installation reboot, see “Network configuration fails during interactive installation.”

In addition, verify that all required NIC drivers were installed to the disk during installation. For more information, see “JumpStart preparation.”

Additional documentation

This section provides technical support information.

Where to go for additional help

While this guide provides guidance about Solaris installation on ProLiant servers, it is not a replacement for the appropriate documentation from HP and others. The following sections list this documentation.

HP documentation

- ProLiant server support matrix, at <http://www.hp.com/wwsolutions/solaris>
- *HP Smart Array Controller Driver for Solaris 10 User Guide*, at ftp://ftp.compaq.com/pub/softlib2/software1/pubsw-windows/p308167783/v40878/HP_Smart_Array_Controller_Driver_User_Guide.pdf
- *HP ROM-Based Setup Utility User Guide*, at <http://h20000.www2.hp.com/bc/docs/support/SupportManual/c00191707/c00191707.pdf>

Free Software Foundation documentation

- *GNU GRUB Manual 0.97*, at http://www.gnu.org/software/grub/manual/html_node/index.html

Intel documentation

- *Preboot Execution Environment (PXE) Specification Version 2.1*, at <http://download.intel.com/design/archives/wfm/downloads/pxespec.pdf>

Sun Microsystems documentation

- Solaris 10 11/06 Release and Installation Collection, at <http://docs.sun.com/app/docs/coll/1236.5>.
- *Solaris 10 11/06 Installation Guide: Basic Installations*, at <http://docs.sun.com/app/docs/doc/819-6394>
- *Solaris 10 11/06 Installation Guide: Network-Based Installations*, at <http://docs.sun.com/app/docs/doc/819-6395>.
- *Solaris 10 11/06 Installation Guide: Custom JumpStart and Advanced Installations*, at <http://docs.sun.com/app/docs/doc/819-6397>
- *Solaris 10 11/06 Installation Guide: Solaris Flash Archives (Creation and Installation)*, at <http://docs.sun.com/app/docs/doc/819-6398>
- *System Administration Guide: IP Services*, at <http://docs.sun.com/app/docs/doc/816-4554>
- *GRUB and the Solaris 10 1/06 OS: The New Bootloader for x86 Platforms*, at http://www.sun.com/bigadmin/features/articles/grub_boot_solaris.html

- *GRUB and the Solaris Operating System on x86 Platforms – A Guide to Creating a Customized Boot DVD*, at <http://www.sun.com/blueprints/0806/819-7546.pdf>

Glossary

Deployment server	The component of an Rapid Deployment Pack environment that provides the foundation for deploying and managing servers using Rapid Deployment Pack.
Flash archive	From <i>flar(1M)</i> , A flash archive is an easily transportable version of a reference configuration of the Solaris operating environment, plus other optional software.
GRand Unified Bootloader (GRUB)	The bootloader used by Solaris 10 1/06 and later.
HP ProLiant Essentials Rapid Deployment Pack	A multiserver deployment tool that enables IT administrators to deploy large volumes of servers in an unattended, automated fashion. For more information, see http://www.hp.com/go/rdp .
Install Media Root Directory	The directory to which the Solaris installation media has been copied. This directory was given to <i>setup_install_server(1M)</i> when the media was copied to disk.
Install server	An NFS server that serves the Solaris installation media to clients undergoing network-based development.
Integrated Lights-Out (iLO)	The remote management facility on HP ProLiant servers. For more information, see http://www.hp.com/go/ilo .
JumpStart profile	A configuration file used in automated deployments that provides certain installation configuration information, such as filesystem layout, disk partitioning, and which packages are to be installed.
JumpStart server	A server that makes JumpStart profiles and <i>sysidcfg(4)</i> files available through NFS to installation clients.
Miniroot	A compressed disk image of a small but functional single-file system Solaris instance that can be booted by GRUB and is used during installation and in failsafe mode.
Multiboot	The Solaris multiboot program that is loaded and run by GRUB, and loads and transfers control to the Solaris kernel.
Option ROM Configuration for Arrays (ORCA)	A utility that can be used to perform simple Smart Array administration. ORCA can be accessed by pressing F8 when prompted during POST.
Power On Self-Test (POST)	A portion of a server run cycle that occurs after power-on (or reboot) but before a bootloader (such as GRUB) is run, and includes, for example, optional ROM initialization.
Pre-installation environment	A static, standalone instance of an operating system that is booted over the network and is used by Rapid Deployment Pack to perform pre-installation hardware configuration. Rapid Deployment Pack uses both Windows-based and Linux-based pre-installation environments.
Preboot eXecution Environment (PXE)	A specification commonly used on x86-architecture systems that provides for booting such a system from the network. On ProLiant servers, a PXE boot can be initiated by pressing F12 when prompted during POST.
ROM-Based Setup Utility (RBSU)	A utility that can be accessed by booting the server and pressing F9 when prompted during the POST.
Staging system	A system that prepares an object that can be used or stored elsewhere, such as a miniroot or custom ISO image. Such a system might be necessary if the system that ultimately stores the object is not capable of producing it. Conceptually, the staging system is only required during the initial preparation of an object, not during the normal use of that object.

Sysidcfg(4)

A file that provides the system identification information such as root password, networking parameters, time zone, and so on, during an automated installation. For more information, see *sysidcfg(4)*.

Trivial File Transfer Protocol (TFTP)

A protocol used during a PXE-based boot to transfer the network boot program from the server to the boot system.

Virtual media

An HP iLO feature that enables local media on the iLO client or network-based CD/diskette images to be connected as virtual USB devices on ProLiant servers in remote locations and data centers.

Index

A

- about this guide, 6
- additional drivers, 13
- AMD memory scrubber, 40
- applicable OS versions, 6
- applicable servers, 6

B

- basic installation
 - interactive, 9
 - overview, 8
 - prerequisites, 8
- basic troubleshooting techniques
 - GRUB command line, 44
 - kernel debugger, 44
 - network verification, 44
 - shell prompt, 43
- bcme fails to attach, 41
- boot methods, 16
- booting Solaris from GRUB
 - HPQnetstrategy, 31
 - options, 31
 - overview, 31

C

- common problems
 - GRUB cannot boot Solaris, 45
 - GRUB does not display a menu, 45
 - installation succeeds but no network availability after reboot, 47
 - installation succeeds but server fails to boot, 47
 - installation terminates prematurely, 46
 - JumpStart reverts to interactive configuration, 46
 - Rapid Deployment Pack job fails, 45
 - system becomes unresponsive using remote console, 45
 - system does not boot to GRUB, 44
- components
 - configuration information, 11
 - GRUB, 10
 - installation, 10

- miniroot, 11
- overview, 10
- packages, 11
- configuration information, 11
- configuration methods, 17
- custom bootable ISO image creation
 - prerequisites, 33
 - procedure, 33

D

- deploying ProLiant servers, 12
 - additional drivers, 13
 - deployment resource kit, 15
 - flash archives, 15
 - iLO virtual media, 14
 - JumpStart, 14
 - miniroot, 13
 - Rapid Deployment Pack, 14
- deployment planning
 - implementing a custom ISO-boot-based deployment, 18
 - implementing a PXE-boot-based deployment, 17
 - implementing an RDP-initiated deployment, 18
 - overview, 16
 - selecting a strategy, 16
- deployment resource kit, 15
- deployment tasks
 - custom bootable ISO image creation, 33
 - GRUB preparation, 30
 - install server preparation, 19
 - JumpStart preparation, 23
 - miniroot preparation, 21
 - overview, 19
 - Rapid Deployment Pack preparation, 28
- DHCP server, 12
- DHCP/PXE preparation
 - overview, 26
 - prerequisites, 26
 - procedure, 27
- disabled PCI devices, 39

- DiskSuite, 34
- drivers
 - additional ProLiant, 13
- E**
- Embedded Ethernet not configured, 42
- Enabling iLO virtual serial port
 - Optional configuration tasks, 34
- F**
- flash
 - archives, 15
- flash archives, 15
- G**
- GRUB, 10
- GRUB cannot boot Solaris, 45
- GRUB command line, 44
- GRUB does not display a menu, 45
- GRUB preparation
 - booting Solaris from GRUB, 31
 - examples, 32
 - overview, 30
- H**
- help resources
 - Free Software Foundation documentation, 48
 - HP documentation, 48
 - Intel documentation, 48
 - overview, 48
 - Sun Microsystems documentation, 48
- HP ProLiant Essentials Rapid Deployment Pack, 14
- I**
- iLO boot hang, 40
- iLO virtual media, 14
- install server, 12
- install server preparation
 - prerequisites, 19
 - procedure, 20
- installation
 - general process flow, 10
- installation succeeds but no network availability after reboot, 47
- installation succeeds but server fails to boot, 47
- installation terminates prematurely, 46
- interactive installation, 9
- ISO-boot-based deployment, 18

- J**
- JumpStart, 14
- JumpStart preparation
 - example, 24
 - overview, 23
 - prerequisites, 23
- JumpStart preparation prerequisites
 - additional required packages, 23
 - Rapid Deployment Pack, 24
- JumpStart reverts to interactive configuration, 46
- JumpStart server, 12
- K**
- kernel debugger, 44
- known issues
 - AMD memory scrubber, 40
 - disabled PCI devices, 39
 - iLO boot hang, 40
 - network configuration fails during interactive installation, 39
 - SNMP master agent, 40
- M**
- miniroot, 11, 13
- miniroot preparation
 - example, 22
 - prerequisites, 21
 - procedure, 21
- Mirroring disks, 34
- N**
- network configuration fails during interactive installation, 39
- network deployment infrastructure
 - DHCP server, 12
 - install server, 12
 - JumpStart server, 12
 - overview, 11
 - TFTP server, 12
- network verification, 44
- O**
- Optional configuration tasks, 34
- P**
- packages, 11
- prerequisites, 8
- PXE-boot-based deployment, 17

R

- Rapid Deployment Pack (RDP), 14
- Rapid Deployment Pack job fails, 45
- Rapid Deployment Pack preparation
 - overview, 29
 - prerequisites, 29
 - procedure, 29
- RDP, 14
- RDP-initiated deployment, 18

S

- selecting a deployment strategy
 - boot methods, 16
 - software delivery methods, 17
- shell prompt, 43
- SNMP master agent, 40
- software delivery methods, 17
- Solaris installation
 - components, 10
 - deploying ProLiant servers, 12

- general process flow, 10

- network deployment infrastructure, 11

- overview, 10

- Solaris media packages, 11

- system becomes unresponsive using remote console, 45

- system does not boot to GRUB, 44

T

- technical support, 48

- TFTP server, 12

- troubleshooting

- basic techniques, 43

- common issues, 44

- known issues, 38

- overview, 38

U

- using this guide, 7

V

- virtual media, 14