Transitioning From Oracle® Solaris 10 to Oracle Solaris 11.1

Part No: E36219–02
September 2013

ORACLE

130903@25097

Contents

Preface

Transitioning From Oracle Solaris 10 to Oracle Solaris 11.1 covers topics for transitioning from Oracle Solaris 10 to Oracle Solaris 11.1, as well cumulative feature changes that were introduced in Oracle Solaris 11 11/11.

Note – This Oracle Solaris release supports systems that use the SPARC and x86 families of processor architectures. The supported systems appear in the *Oracle Solaris OS: Hardware Compatibility Lists*. This document cites any implementation differences between the platform types.

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

Who Should Use This Book

This book is intended for anyone responsible for administering one or more systems running the Oracle Solaris 11 release. To use this book, you should have 1–2 years of UNIX system administration experience. Attending UNIX system administration training courses might be helpful.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Typographic Conventions

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

TABLE P–1 Typographic Conventions

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

Shell Prompts in Command Examples

The following table shows UNIX system prompts and superuser prompts for shells that are included in the Oracle Solaris OS. In command examples, the shell prompt indicates whether the command should be executed by a regular user or a user with privileges.

TABLE P–2 Shell Prompts

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

General Conventions

Be aware of the following conventions used in this book.

- When following steps or using examples, be sure to type double-quotes ("), left single-quotes (‘), and right single-quotes (’) exactly as shown.
- The key referred to as Return is labeled Enter on some keyboards.
- The `root` path usually includes the `/usr/sbin`, `/usr/bin`, and `/etc` directories, so the steps in this book show the commands in these directories without absolute path names. Steps that use commands in other, less common, directories show the absolute paths in the examples.

◆ ◆ ◆ CHAPTER 1

Transitioning From Oracle Solaris 10 to an Oracle Solaris 11 Release (Overview)

This chapter provides overview information about transitioning from Oracle Solaris 10 to an Oracle Solaris 11 release.

The following topics are covered:

Welcome to Oracle Solaris 11.1

The Oracle Solaris 11.1 operating system (OS) is the first update of the latest major Oracle Solaris release, Oracle Solaris 11 11/11. An operating system for the enterprise environment, Oracle Solaris 11.1 is an integral part of Oracle's combined hardware and software portfolio. If you are moving from Oracle Solaris 10 to an Oracle Solaris 11 release, you might have some questions. The purpose of this guide is to provide answers to some of those questions.

Note – This book contains cumulative information for anyone who is transitioning from Oracle Solaris 10 to an Oracle Solaris 11 release. Important information about feature differences between Oracle Solaris 11 and Oracle Solaris 11.1 is also provided, when appropriate. For specific details about transitioning from Oracle Solaris 10 to the Oracle Solaris 11 11/11 release, see *Transitioning From Oracle Solaris 10 to Oracle Solaris 11*. For more information about a particular feature, refer to the product documentation.

Most Oracle Solaris 10 applications are known to work on Oracle Solaris 11. You can run supported applications *as is*. Or, you can run applications that rely on features that are excluded from Oracle Solaris 11 in an Oracle Solaris 10 virtual environment. See Chapter 10, "Managing Oracle Solaris Releases in a Virtual Environment." To determine the readiness of Oracle Solaris 10 applications to run on an Oracle Solaris 11 release, use the Oracle Solaris 11 compatibility checking tool, which is available at:

`http://www.oracle.com/technetwork/indexes/samplecode/solaris-sample-522122.html`

See also the following documentation:

`http://www.oracle.com/technetwork/articles/systems-hardware-architecture/o10-015-s11-isv-adoption-198348.pdf`

Note that this guide does not provide information about every new feature in Oracle Solaris 11, nor does it mention every feature that is excluded from Oracle Solaris 11.

- For information about new features, go to `http://www.oracle.com/technetwork/server-storage/solaris11/documentation/solaris11-1-whatsnew-1732377.pdf`.
- For information about excluded features, go to `http://www.oracle.com/technetwork/systems/end-of-notices/index.html`.
- For information about upgrading from Oracle Solaris 11 to Oracle Solaris 11.1, see Updating to Oracle Solaris 11.2.
- For information about Oracle's Sun hardware platforms and the corresponding Oracle Solaris operating system requirements, go to `http://www.oracle.com/technetwork/systems/software-stacks/stacks/index.html`.

Oracle Solaris 10 Features Compared to Oracle Solaris 11 Features

The following table compares Oracle Solaris 10 features to Oracle Solaris 11 features.

Note – Features are listed in alphabetical order.

TABLE 1–1 Oracle Solaris 10 Features Compared to Oracle Solaris 11 Features

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Oracle Solaris 11.1
x86: boot loader (GRUB)	GRUB Legacy (0.97)	GRUB Legacy (0.97)	GRUB 2 "GRUB, Firmware, and Disk Labeling Changes" on page 122
boot loader (administration)	**SPARC:** `installboot` **x86:** `installgrub`	**SPARC:** `installboot` **x86:** `installgrub`	`bootadm install-bootloader` (SPARC and x86) "GRUB, Firmware, and Disk Labeling Changes" on page 122
Booting (from a root device)	From a ZFS, UFS, or Solaris Volume Manager root device	From an Oracle Solaris ZFS root file system	From an Oracle Solaris ZFS root file system "System Boot, Recovery, and Platform Changes" on page 121
Booting (from the network)	**SPARC:** From the `ok` PROM prompt: `boot net[:dhcp]` or `boot net[:rarp]` **x86:** Requires a DHCP server that supports a Preboot Execution Environment (PXE) boot from the network	**SPARC:** `boot net:dhcp` **x86:** Requires a DHCP server that supports a PXE boot from the network	**SPARC:** `boot net:dhcp` **x86:** UEFI firmware and BIOS firmware types are supported. The PXE boot process has changed for UEFI firmware. "Booting Systems With UEFI and BIOS Firmware From the Network" in *Booting and Shutting Down Oracle Solaris 11.1 Systems*

TABLE 1–1 Oracle Solaris 10 Features Compared to Oracle Solaris 11 Features *(Continued)*

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Oracle Solaris 11.1
Booting (recovery)	**SPARC:** ok `boot -F failsafe` **x86:** Select the failsafe boot entry in the GRUB menu at boot time	Failsafe mode no longer supported (SPARC and x86) ■ Boot from an alternate boot environment (BE) or backup BE ■ Boot in single-user mode or perform system recovery steps	Failsafe mode is not supported on SPARC or x86 platforms. ■ Boot from an alternate BE or a backup BE. ■ Boot in single-user mode or perform system recovery steps. "System Boot, Recovery, and Platform Changes" on page 121
Desktop environment	Common Desktop Environment (CDE) (default) and GNOME 2.6 (optional)	Oracle Solaris Desktop (GNOME 2.30)	Oracle Solaris Desktop (GNOME 2.30) Chapter 12, "Managing Desktop Features"
Disk labeling	UFS root disk is SMI (VTOC); UFS non-root disk is SMI or EFI ZFS root disk is SMI (VTOC); ZFS non-root disk is SMI or EFI (recommended)	ZFS root disk is SMI (VTOC); ZFS non-root disk is SMI or EFI (recommended)	**SPARC with GPT-aware firmware and x86:** ZFS root disk is EFI (GPT) **SPARC:** ZFS root disk is SMI (VTOC) **SPARC and x86:** ZFS non-root disk is SMI or EFI (recommended)
File systems (default)	ZFS, UFS, or Solaris Volume Manager root file systems	ZFS root file system (default)	ZFS root file system (default) Chapter 5, "Managing File Systems"
x86: Firmware support	BIOS	BIOS	UEFI and BIOS Chapter 3, "Managing Devices"
GRUB configuration file (default)	`menu.lst`	`menu.lst`	`grub.cfg` (Do *not* edit) "GRUB, Firmware, and Disk Labeling Changes" on page 122
GRUB configuration file (custom)	`menu.lst`	`menu.lst`	`custom.cfg`

TABLE 1–1 Oracle Solaris 10 Features Compared to Oracle Solaris 11 Features *(Continued)*

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Oracle Solaris 11.1
Installation (graphical user interface (GUI))	GUI installation program on DVD or CD	Live Media (x86 only)	Live Media (x86 only)
Installation (interactive text)	Interactive text installation and interactive text installer for ZFS root pools	Text installer (stand-alone and network installation)	Text installer (stand-alone and network installation)
Installation (automated)	JumpStart feature of Oracle Solaris 10	Automated Installer (AI) feature of Oracle Solaris 11	Automated Installer (AI) feature of Oracle Solaris 11 Oracle VM Manager Ops Center
Installation (automated client configuration)	JumpStart profile files	AI manifests	AI manifests
Installation (other)	Oracle Solaris Flash Archive installation	See "System Boot, Recovery, and Platform Changes" on page 121.	See "System Boot, Recovery, and Platform Changes" on page 121.
Java (default version)	Java 6	Java 6	Java 7 Chapter 12, "Managing Desktop Features"
Network configuration (fixed and reactive)	`ifconfig` Edit `/etc/hostname.*` `ndd` for configuring protocols	**Fixed:** `dladm` for datalinks and `ipadm` for IP configuration **Reactive:** `netcfg` and `netadm`	**Fixed:** `dladm` for datalinks, `ipadm` for IP configuration, and `netadm` for viewing information about NCPs, including the `DefaultFixed` NCP **Reactive (active and non-active NCPs):** `netcfg` and `netadm` **Reactive (applies to currently active NCP only):** `dladm` and `ipadm` for datalink and IP configuration Chapter 7, "Managing Network Configuration"

TABLE 1–1 Oracle Solaris 10 Features Compared to Oracle Solaris 11 Features *(Continued)*

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Oracle Solaris 11.1
Network configuration (DHCP)	Oracle Solaris DHCP and other naming services	Internet Systems Consortium (ISC) DHCP and legacy Sun DHCP	Internet Systems Consortium (ISC) DHCP and legacy Sun DHCP *Working With DHCP in Oracle Solaris 11.1*
Network configuration (IPMP)	Miscellaneous commands, for example: `ifconfig` and `plumb`/`umplumb`	`dladm` and `ipadm`	`dladm` and `ipadm` See Appendix A, "Comparison Map: ifconfig and ipadm Commands," in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*
Network configuration (TCP/IP properties or tunables)	`ndd`	`ipadm`	`ipadm` See Appendix B, "Comparison Map: ndd and ipadm Commands," in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*
Network configuration (wireless)	`wificonfig`	**Fixed:** `dladm` and `ipadm` **Reactive:** `netcfg` and `netadm` **From the desktop:** NWAM GUI	**Fixed:** `dladm` and `ipadm` **Reactive:** `netcfg`, and `netadm` **From the desktop:** Network administration GUI
Packaging (software management)	SVR4 package and patch commands	Image Packaging System (IPS) `pkg`(1) commands, Package Manager and Update Manager GUIs	Image Packaging System (IPS) `pkg`(1) commands, Package Manager and Update Manager GUIs Chapter 6, "Managing Software and Boot Environments"
Print service (default)	LP print service, `lp` print commands, Solaris Print Manager GUI	CUPS	CUPS "Printer Configuration and Management Changes" on page 129

TABLE 1–1 Oracle Solaris 10 Features Compared to Oracle Solaris 11 Features *(Continued)*

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Oracle Solaris 11.1
Security management	`root` as a user account	`root` as a role	`root` as a role Chapter 9, "Managing Security"
System clustering	Oracle Solaris Cluster 3.3	Oracle Solaris Cluster 4.0	Oracle Solaris Cluster 4.1
System configuration and reconfiguration	`sysidtool`, `sys-unconfig`, `sysidconfig`, and `sysidcfg`	`sysconfig`, System Configuration Interactive (SCI) tool, SC profiles	`sysconfig`, System Configuration Interactive (SCI) tool, SC profiles Chapter 8, "Managing System Configuration"
System configuration (naming services)	Configured in files within `/etc` and `/var`	Managed through the Service Management Facility (SMF) feature of Oracle Solaris	Managed through the Service Management Facility (SMF) feature of Oracle Solaris Chapter 8, "Managing System Configuration"
System configuration (hostname)	Edit `/etc/nodename`	`svccfg -s` sets the `config/nodename` property of the `svc:system/identity:node` service to the desired name.	Use the `hostname` command. "System Configuration Changes and Migration of System Configuration to SMF" on page 113
System management (centralized)	Oracle Enterprise Manager Ops Center 11*g*	Oracle Enterprise Manager Ops Center 12*c*	Oracle Enterprise Manager Ops Center 12*c* (12.1.2.0.0)
System registration	Auto Registration feature Oracle Configuration Manager, starting with Oracle Solaris 10 1/13	Oracle Configuration Manager	Oracle Configuration Manager and the Oracle Auto Service Request utility
System upgrade and BE management	`lu` and SVR4 package commands	`pkg` commands, Package Manager, Update Manager `beadm` utility for managing boot environments	`pkg` commands, Package Manager, Update Manager `beadm` utility for managing boot environments Chapter 6, "Managing Software and Boot Environments"

TABLE 1–1 Oracle Solaris 10 Features Compared to Oracle Solaris 11 Features *(Continued)*

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Oracle Solaris 11.1
User account management	`useradd`, `usermod`, `userdel`, `groupadd`, `groupmod`, `groupdel`, `roleadd`, `rolemod`, and `roledel` Solaris Management Console GUI and equivalent command-line	`useradd`, `usermod`, `userdel`, `groupadd`, `groupmod`, `groupdel`, `roleadd`, `rolemod`, and `roledel`	`useradd`, `usermod`, `userdel`, `groupadd`, `groupmod`, `groupdel`, `roleadd`, `rolemod`, and `roledel` User Manager GUI "Commands and Tools for Managing User Accounts" on page 155
User environment management	Korn shell (`ksh`) `MANPATH` variable required	**Default shell:** `ksh93` **Default `ksh` path:** `/usr/bin/ksh`; `/bin/sh` is also `ksh93` **Default interactive shell:** `bash`; default `bash` path: `/usr/bin/bash` `MANPATH` variable no longer required	See Chapter 11, "Managing User Accounts and User Environments"
ZFS root pool disk (SPARC and x86)	Root pool disk requires SMI (VTOC) disk label and a slice 0	See "ZFS Root Pool Disk and Boot Administration" on page 52	See "ZFS Root Pool Disk and Boot Administration" on page 52

Removal of Legacy System Management Commands, Tools, Services, and Files

The following table lists (in alphabetical order) the commands, files, services, and tools that are deprecated or have been removed.

TABLE 1–2 Legacy System Management Commands, Files, Services, and Tools

Legacy Command, File, Service, or Tool	Replacement Command, Tool, Service, or File	For More Information
`bsmconv` and `bsmunconv`	`audit`	`audit(1M)`
`crypt` and `des`	`encrypt`	`encrypt(1)`
`/etc/defaultrouter`	`route`	`route(1M)`

TABLE 1–2 Legacy System Management Commands, Files, Services, and Tools *(Continued)*

Legacy Command, File, Service, or Tool	Replacement Command, Tool, Service, or File	For More Information
`graph` and `spline`	`gnuplot`	`gnuplot`(1) **Note –** Install the `image/gnuplot` package.
SPARC: `installboot` **x86:** `installgrub` The `installgrub` command is deprecated and should only be used to install the boot loader on systems that support GRUB 2.	**Oracle Solaris 11.1:** `bootadm install-bootloader` (SPARC and x86)	"ZFS Root Pool Disk and Boot Administration" on page 52
`localeadm`	Locale facet mechanism	"Internationalization and Localization Changes" on page 131
Print commands: `download`, `lpfilter`, `lpforms`, `lpget`, `lpset`, `lpsched`, `lpshut`, `lpsystem`, `lpusers`, `printmgr` (launches Solaris Print Manager), `print-service` , and `ppdmgr`	`cancel`, `cupsaccept`, `cupsreject`, `cupsdisable`, `cupsenable`, `lp`, `lpadmin`, `lpc`, `lpinfo`, `lpmove`, `lpoptions`, `lpq`, `lpr`, `lprm`, `lpstat`, and `system-config-printer` (launches CUPS Print Manager)	"Printer Configuration and Management Changes" on page 129
Print (LP) files and descriptions: ■ `~/.printers` ■ `/etc/printers.conf` ■ `/etc/lp/printers` ■ `/var/spool/lp` ■ `/var/lp/logs`	**CUPS print files and descriptions**: ■ `~/.cups/lpoptions` ■ `/etc/cups/printers.conf` ■ `/etc/cups` ■ `/var/spool/cups` ■ `/var/log/cups`	`lpoptions`(1)
Legacy SMF print services: ■ `svc:/application/print/ppd-cache-update:default` ■ `svc:/application/print/server:default` ■ `svc:/application/print/rfc1179:default` ■ `svc:/network/device-discovery/printers:snmp` ■ `svc:/application/print/ipp-listener:default` ■ `svc:/application/print/service-selector:default` **Replacement SMF print services:** ■ `svc:/application/cups/scheduler` ■ `svc:/application/cups/in-lpd`		"Printer Configuration and Management Changes" on page 129
`pmconfig` and `/etc/power.conf`	`poweradm`	`poweradm`(1M)
`rdist`	`rsync` or `scp`	`rsync`(1) and `scp`(1)
`rstart` and `rstartd`	`ssh`	`ssh`(1)

TABLE 1–2 Legacy System Management Commands, Files, Services, and Tools *(Continued)*

Legacy Command, File, Service, or Tool	Replacement Command, Tool, Service, or File	For More Information
`saf`, `sac`, `sacadm`, `nlsadmin`, `pmadm`, `ttyadm`, and`listen` `/usr/include/listen.h`, `getty`, `/usr/lib/saf/nlps_server`, `/var/saf`, `/etc/saf`, `ttymon` (`sac` and `getty` modes *only*), and `ports` (`sac` functionality)	`ttymon express` mode is still supported by the following SMF services: ■ `svc:/system/console-login:terma` ■ `svc:/system/console-login:termb`	"System Console, Terminal Services, and Power Management Changes" on page 118

Network SMF services:
 	
`svc:/network/physical:default`
 	
`svc:/network/physical:nwam`
 	
This SMF service is deprecated in Oracle Solaris 11. However, the service is still listed in the output of the `svcs -a` command.
	`svc:/network/physical:default`	Chapter 7, "Managing Network Configuration"

`smosservice` and `smdiskless`	No replacement available	
`sysidtool`, `sys-unconfig`, and `sysidcfg`	`sysconfig`, SCI tool, and SC configuration through profiles	"System Configuration Tools Changes" on page 120
User account management: Solaris Management Console GUI, `smc`, `smuser`, `smgroup`, and `passmgmt`	`useradd`, `usermod`, `userdel`, `groupadd`, `groupmod`, `groupdel`, `roleadd`, `rolemod`, `roledel` **Starting with Oracle Solaris 11.1:** User Manager GUI	"Managing User Accounts" on page 156
`vold` daemon	`volfs` and `rmvolmgr`	Chapter 3, "Managing Devices"

For more information about legacy commands that are no longer supported, see *Oracle Solaris 11.1 Release Notes*.

Transitioning Your Oracle Solaris 10 System to an Oracle Solaris 11 Release

When transitioning to Oracle Solaris 11, keep the following key points in mind:

- No upgrade methods or tools are available to transition from Oracle Solaris 10 to an Oracle Solaris 11 release. You cannot use an installer to upgrade from Oracle Solaris 10 to an Oracle Solaris 11 release. You must perform a fresh installation by using one of the installation options that are described in this chapter.

  However, you can migrate your Oracle Solaris 10 OS instances or zones and your data to an Oracle Solaris 11 system. For more information, see Table 1–3.
- The following Oracle Solaris 10 installation features are not available in an Oracle Solaris 11 release: the Oracle Solaris installation upgrade option, the Oracle Solaris Flash Archive installation method, JumpStart, and the Oracle Solaris Live Upgrade feature (`lu` suite of commands).

  The Automated Installer (AI) replaces JumpStart, and the `beadm` utility provides similar functionality to the `lu` commands. For more information, see "Migrating From JumpStart to AI" on page 35 and "Tools for Managing Boot Environments" on page 80.
- Oracle Solaris 11 introduces the Image Packaging System (IPS), which is a different mechanism than the legacy SVR4 package commands that are used in Oracle Solaris 10 and previous releases. See Chapter 6, "Managing Software and Boot Environments."

Table 1–3 describes the tools and features that are available for transitioning to an Oracle Solaris 11 release.

TABLE 1–3 Oracle Solaris 11 Transition Tools and Features

Tool or Feature	Description	For More Information
JumpStart Migration Utility (`js2ai`)	Used to convert Oracle Solaris 10 JumpStart rules, profiles, and `sysidcfg` file to a format that is compatible with AI manifest entries.	*Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.1 Automated Installer*
ZFS shadow migration feature	Used to migrate data from an existing file system to a new file system.	Chapter 4, "Managing Storage Features"
Oracle Solaris 11 support for Oracle Solaris 10 zones	Used to migrate your Oracle Solaris 10 application environments to an Oracle Solaris 11 system.	Chapter 10, "Managing Oracle Solaris Releases in a Virtual Environment"

TABLE 1–3 Oracle Solaris 11 Transition Tools and Features *(Continued)*

Tool or Feature	Description	For More Information
NFS file sharing and pool migration	Used to access shared files from an Oracle Solaris 10 system on an Oracle Solaris 11 system. Used to import a ZFS storage pool from an Oracle Solaris 10 system into an Oracle Solaris 11 system.	Chapter 5, "Managing File Systems"

Installation Features

The following installation methods are available:

- **x86: GUI installation with the Live Media** – The GUI installer can be used to install Oracle Solaris 11 on x86 platforms *only*. The GUI installer is capable of functioning with a minimum 1.5 GB of memory. The exact minimum requirement varies depending on system specifications. See "Installing Oracle Solaris by Using Installation Media" on page 34 for details.
- **Interactive text installation (From media or over the network)** – The text installer enables you to install Oracle Solaris on SPARC and x86 based systems from media or over a network.
- **Automated installation on single or multiple systems** – The Automated Installer (AI) installs Oracle Solaris 11 on a single or multiple client systems from an installation server on a network. Similar to JumpStart, AI provides a hands-free installation. You can also perform automated installations that boot from media. See "Installing Oracle Solaris by Using AI" on page 38.

  AI also supports the installation of zones. See "Oracle Solaris 11 Zone Features" on page 148.
- **Customized installation image creation with the Distribution Constructor** – The Distribution Constructor tool creates preconfigured installation images. See "Oracle Solaris Installation Methods" on page 32.

These installation tools and methods are no longer available:

- **Oracle Solaris Flash Archive Installation** – For information about recovering from a system failure, see "System Boot, Recovery, and Platform Changes" on page 121.
- **JumpStart feature of Oracle Solaris** – AI replaces JumpStart in this release. See "Installing Oracle Solaris by Using AI" on page 38.
- **Oracle Solaris Live Upgrade feature** – The suite of commands (`lu`) that are part of the Oracle Solaris Live Upgrade feature are also no longer supported. The `beadm` utility provides similar capability. See "Tools for Managing Boot Environments" on page 80.

See Chapter 2, "Transitioning to an Oracle Solaris 11 Installation Method."

Automated Installer Enhancements

The following automated installation enhancements are introduced in this release:

- `installadm` **command options** – The `installadm` command has three new options: `update-service`, `update-profile`, and `set-service`. These options enable you to maintain a set of installation services. The ability to specify a manifest location with a system boot argument has also been added in this release. See Part III, "Installing Using an Install Server," in *Installing Oracle Solaris 11.1 Systems*.
- **Installer support for connecting to Oracle support services** – The Oracle Configuration Manager and the Oracle Auto Services Request utility are enabled by default for the purpose of collecting system configuration information during an installation. Both services are enabled through two new Oracle Solaris 11.1 installation screens. See Appendix A, "Working With Oracle Configuration Manager," in *Installing Oracle Solaris 11.1 Systems*.
- **Interactive installation on iSCSI Targets** – The ability to install to iSCSI target logical unit numbers (LUNs) in included in the Oracle Solaris 11.1 interactive text and Live Media installers. You can choose between installing on local disks or connecting to a remote iSCSI disk by using DHCP auto-discovery or by manually specifying a target IP address, iSCSI target name and LUN, and an initiator name. This feature change enables installed OS images to be maintained in a central location. See "Installing With the GUI installer" in *Installing Oracle Solaris 11.1 Systems*.
- **Role-based access control (RBAC) profiles and authorizations for managing the Automated Install service** – Many of the commands that are used with an automated installation require increased privilege. Use one of the following methods to gain more privilege:
  - Use the `profiles` command to list the privileges that are assigned to you.
  - Use the `sudo` command with your user password to execute a privileged command. Use of the `sudo` command is dependent upon the security policy at your site.
  - Use the `roles` command to list the roles that are assigned to you. If you have the `root` role, you can use the `su` command to assume that role.

  See "Install Server Requirements" in *Installing Oracle Solaris 11.1 Systems*.

Software and Boot Environment Management Features

The Oracle Solaris 11 software is distributed in packages that are managed by the Image Packaging System (IPS). After you install the OS, you can access *package repositories* to install additional or updated software packages on your system. With IPS commands, you can list, search, install, update, and remove software packages.

Software management includes the following components:

- **IPS command-line utilities** – IPS includes `pkg` commands that install and manage packages from the command line. IPS commands also enable you to manage package publishers and copy or create package repositories.
- **IPS repositories** – An *IPS repository* is a location from which you can install software packages.
- **Boot environment management** – Boot environments (BEs) are bootable instances of an image. The `beadm` utility is used to create and manage boot environments.

Note – No upgrade path from Oracle Solaris 10 to Oracle Solaris 11 is available. You must perform a fresh installation, but first review the migration features in Table 1–3. You can use the `pkg update` command to update one or more packages from one version to a newer version.

See Chapter 6, "Managing Software and Boot Environments."

Network Administration Features

Network administration includes the following key features:

- **Bridging technologies** – Bridges are used to connect separate network segments that are paths between two nodes. When connected by a bridge, the attached network segments communicate as if they were a single network segment. Bridging is implemented at the datalink layer (L2) of the networking stack. Bridges use a packet-forwarding mechanism to connect subnetworks.

  Data center bridging (DCB) enables the same network fabric to be used for both Ethernet and storage traffic. For information about additional changes to bridging technologies in the current release, see "Network Configuration Feature Changes" on page 83.
- **Datalink and IP configuration migration to one SMF network configuration repository** – One SMF network configuration repository for both datalink and IP configuration is now used. Also, The `svc:/network/physical:default` SMF service now manages network configuration for both fixed and reactive network configuration.
- **Generic datalink name assignment** – Generic names are automatically assigned to datalinks using the `net0`, `net1`, `net`*N* naming convention, depending on the total number of network devices that are on the system. See "Displaying and Configuring Datalinks in Fixed Mode" on page 91.
- **Integrated Load Balancer (ILB)** – The ILB feature of Oracle Solaris provides Layer 3 and Layer 4 load-balancing capabilities for both SPARC and x86 based systems. ILB intercepts incoming requests from clients, decides which back-end server should handle the request based on load-balancing rules, and then forwards the request to the selected server. You can optionally configure an Oracle Solaris system as a load balancer. ILB performs optional

health checks and provides the data for the load-balancing algorithms, then verifies whether the selected server can handle the incoming request. See `ilbadm(1M)`.

- **IP network multipathing (IPMP) configuration changes** – Starting with Oracle Solaris 11, IPMP has a new conceptual model and different commands for managing IPMP configuration. See "Configuring IPMP in Oracle Solaris 11" on page 104.
- **Network observability** – In Oracle Solaris 10, the `ifconfig` and `netstat` commands are used to manage network observability. In Oracle Solaris 11, the `dlstat` and `flowstat` commands are used. For zones, you can use the `zonestat` command. See the `dlstat(1M)`, `flowstat(1M)`, and `zonestat(1)` man pages.
- **Profile-based network configuration** – Starting with Oracle Solaris 11, network configuration is profile-based. Two network configuration modes are used: fixed and reactive. The switch between network configuration modes no longer takes place at the service level, but at the profile level. The system defaults to fixed network configuration mode after a text installation or an installation with AI. See "How the Network Is Configured During an Installation" on page 86.
- **Virtual network interface cards (VNICs)** – VNICs are pseudo interfaces that you create on top of datalinks. Along with virtual switches, VNICs are the building blocks of a virtual network. You can create and modify VNICs in a system or in a zones environment. Starting with Oracle Solaris 11.1, the migration of VNICs is also possible. See "Building Virtual Networks" in *Using Virtual Networks in Oracle Solaris 11.1*.

See Chapter 7, "Managing Network Configuration."

System Configuration and SMF Features

The following system configuration and SMF features are supported:

- **Oracle Auto Service Request utility** – This feature can be used by customers who have a valid My Oracle Support account. See "System Registration and System Support Changes" on page 120.
- **SMF administrative layers** – Information for recording the source of properties, property groups, instances, and services has been added to the SMF repository. This information enables you to determine which settings are administrative customizations and which settings were delivered with Oracle Solaris by a manifest. See "SMF Administrative Changes" on page 116.
- **SMF manifest creation tool** – The `svcbundle` command can be used to generate SMF manifests, as well as profiles. The manifest is specified by using multiple `-s` options. To generate a manifest, you must specify the `service-name` and `start-method` command options. See `svcbundle(1M)`.

- **System Configuration Interactive (SCI) utility** – Uses SMF to centralize configuration information. The `sysconfig` utility replaces the `sys-unconfig` and `sysidtool` utilities that are used in Oracle Solaris 10. You can run the `sysconfig` utility interactively by using the SCI utility or in an automated manner by creating an SC configuration profile. See "System Configuration Tools Changes" on page 120.
- **System console and terminal device management** – The system console and locally connected terminal devices are now managed by SMF. The `sac` and `saf` programs for managing console services and terminals are no longer available.
- **System, network, and name services configuration migration to SMF** – Several aspects of system and network configuration, including configuration that was previously stored in various files within the `/etc` directory, is now stored in an SMF repository. Migrating configuration data to SMF service properties enables the delivery of a uniform, extensible architecture for system configuration that provides customers with a more complete capability for managing system configuration. See "System Configuration Changes and Migration of System Configuration to SMF" on page 113.
- **System registration** – Oracle Configuration Manager collects configuration information and then anonymously uploads it to the Oracle repository during the first reboot of a system after an installation. This information is analyzed and then used by Oracle to provide better service to customers. In Oracle Solaris 10, the Auto Registration feature performs a similar function. Starting with the Oracle Solaris 10 1/13 release, Oracle Configuration Manager replaces the Auto Registration feature. See "System Registration and System Support Changes" on page 120.

See Chapter 8, "Managing System Configuration."

Storage and File Systems Features

The following features are related to storage and file systems management:

- **Storage is simplified** – Oracle's Sun ZFS Storage Appliance provides a low-cost storage solution and simplified administration with a browser-based management and monitoring tool. The appliance can be used to share data between your Oracle Solaris 10 and Oracle Solaris 11 systems. As in Solaris 10 releases, data can be shared between your Oracle Solaris 10 and Oracle Solaris 11 systems by using the NFS protocol. In the Oracle Solaris 11 release, you can also share files between systems that are running Oracle Solaris and Windows by using the Server Message Block (SMB) protocol.
- **Improved device management** – New commands are available and existing commands have been updated to help you locate storage devices by their physical locations.
- **ZFS file system is the default file system** – ZFS fundamentally changes the way file systems are administered. ZFS includes features and benefits that are not found in any other file system that is available today.

The following features help you transition either your UFS file system or your ZFS storage pools to systems that are running Oracle Solaris 11:

- **Migrate your UFS data with ZFS shadow migration** – The ZFS shadow migration feature is used to migrate data from an existing file system to a new file system. You can either migrate a local file system to a new file system or migrate an NFS file system to a new local file system. For more information, see "Transitioning Your Oracle Solaris 10 System to an Oracle Solaris 11 Release" on page 23.
- **Migrate your Oracle Solaris 10 storage pools** – Storage devices that contain ZFS storage pools on your Oracle Solaris 10 systems can be exported, disconnected, if necessary, and imported into your Oracle Solaris 11 systems.
- **Migrate your UFS data** – You can remotely mount UFS file systems from an Oracle Solaris 10 system on an Oracle Solaris 11 system. In addition, a `ufsdump` of your UFS data can be restored into a ZFS file system by using the `ufsrestore` command.

See Chapter 4, "Managing Storage Features," and Chapter 5, "Managing File Systems."

Security Features

Oracle Solaris 11 introduces several new features and key enhancements in these areas:

- Auditing
- Containment security
- Cryptographic security
- Network security
- Rights management

For more information about these changes, see Chapter 9, "Managing Security."

Virtualization Features

Oracle Solaris 11 provides support for the following virtualization features:

- Oracle Solaris 10 zones
- Automated installation of non-global zones
- Zones monitoring
- NFS server support
- Network virtualization

For more information about these changes, see Chapter 10, "Managing Oracle Solaris Releases in a Virtual Environment."

User Account Management and User Environment Features

Oracle Solaris 11 introduces several important changes to how user accounts are set up and managed, as well as changes to the default user environment.

Some key changes in this release include the following:

- **Tools for creating and managing user accounts** – In Oracle Solaris 11, user accounts are managed solely by using command-line tools, such as the `useradd`, `usermod`, and `userdel` commands. The Solaris Management Console GUI and its associated command-line, for example, the `smc` and `smuser` commands, are no longer available. Starting with Oracle Solaris 11.1, the User Manager GUI provides capability that is similar to Solaris Management Console. The User Manager GUI can be used to create and manage users from the desktop.
- **Default user shell and path** – Oracle Solaris 11 provides more familiarity with Linux and Berkeley Software Distribution (BSD) operating systems. As a result, the default user shell and path have been changed, See "Default Login Shell and PATH Environment Variable" on page 161.
- **Administrative command locations** – Administrative commands have moved from `/sbin` to `/usr/sbin`.
- **Development tools locations** – Development tools have moved from `/usr/ccs/bin` to `/usr/bin`.

See Chapter 11, "Managing User Accounts and User Environments."

Desktop Features

Oracle Solaris 11 introduces several important changes to the desktop environment. The default desktop is now the Oracle Solaris Desktop, which includes GNOME 2.30 from the GNOME Foundation, the Firefox web browser, Thunderbird Email client, and the Lightning calendar manager from the Mozilla Foundation.

Note – The login manager has changed from CDE to the GNOME Desktop Manager (GDM). If you are transitioning from Oracle Solaris 10 to Oracle Solaris 11, and you previously customized your CDE login, review your display management configuration, as you might need to make some changes to the GDM configuration to ensure it works as expected. For more information, see "Troubleshooting Desktop Transition Issues" on page 168.

See Chapter 12, "Managing Desktop Features."

◆ ◆ ◆ CHAPTER 2

Transitioning to an Oracle Solaris 11 Installation Method

Oracle Solaris 11 introduces new installation features and methods for system administrators. This chapter provides conceptual information and some brief examples to acquaint you with these new methods, including information that applies specifically to Oracle Solaris 11.1. See the relevant Oracle Solaris 11 installation product documentation for detailed installation instructions for the Oracle Solaris 11 release that you are installing. This chapter also includes basic information about migrating from JumpStart to the Automated Installer (AI).

The following topics are covered:

For an end-to-end example of a basic AI installation of a SPARC based system, see Appendix A, "SPARC Automated Installation Scenario."

For information about upgrading your Oracle Solaris 11 system to Oracle Solaris 11.1, see Updating to Oracle Solaris 11.2.

For information about how the network is configured during an installation, see "How the Network Is Configured During an Installation" on page 86.

Oracle Solaris Installation Methods

Table 2–1 summarizes the installation methods that are available in this release. Except for AI, all of these installation methods are used to install single systems. AI can be used to install single systems or multiple systems over the network.

TABLE 2–1 Installation Methods That Are Supported in an Oracle Solaris 11 Release

Installation Method	Preparation?	Install Server?	Single or Multiple Systems
Live Media installation (**x86 only**)	No	No	Single
Text installation	No	No	Single
Text installation over the network	Yes	Yes, for retrieving the installation image from the server.	Single
Automated installations booting from media	Yes	Yes, for customized media preparation. No for installation.	Single
Automated installations of multiple clients	Yes	Yes	Single or multiple

Oracle Solaris Installation Requirements

Before installing an Oracle Solaris 11 release, refer to the following requirements.

ZFS Root Pool Installation Requirements

Oracle Solaris 11 is installed in a ZFS storage pool called the *root pool*. The root pool installation requirements are as follows:

- **Memory** – The minimum memory requirement is 1 GB. The Live Media ISO image, and both the GUI and text installers, are capable of functioning with a limited amount of memory. The exact requirement varies, depending on system specifications.

  If you want to install an Oracle Solaris 11 virtual image on Oracle VM VirtualBox, see the memory requirements listed here:

  `http://www.oracle.com/technetwork/server-storage/solaris11/downloads/virtual-machines-1355605.html`

- **Disk space** – At least 13 GB of disk space is recommended. The space is consumed as follows:

- **Swap area and dump device** – The default sizes of the swap and dump volumes created by the Oracle Solaris installation programs vary, based on the amount of memory that is on the system and other variables.

  After installation, you can adjust the sizes of your swap and dump volumes to the sizes of your choosing, as long as the new sizes support system operation. See "Managing Your ZFS Swap and Dump Devices" in *Oracle Solaris 11.1 Administration: ZFS File Systems*.
- **Boot environment (BE)** – A ZFS BE is approximately 6–8 GB in size, but can vary greatly depending on the size of the dump device. The size of the dump device is based on the size of the system's physical memory. In addition, consider that the size of a new BE increases when it is updated, depending on the amount of updates. You will need to monitor the disk space usage of all BEs on the system. All ZFS BEs in the same root pool use the same swap and dump devices.
- **Oracle Solaris OS components** – All subdirectories of the root file system that are part of the OS image, with the exception of `/var`, must be in the same dataset as the root file system. In addition, all Oracle Solaris OS components must reside within the root pool, with the exception of the swap and dump devices. For information about specific disk requirements, see Chapter 3, "Managing Devices."

- **x86 only: Support for running multiple operating systems** – You can partition the disk that will contain the OS prior to an installation or during an installation. See "Partitioning Your System" in *Installing Oracle Solaris 11.1 Systems*.

Oracle Solaris Preinstallation Tasks

Before installing an Oracle Solaris 11 release, review the following information:

- **x86: Prepare the boot environment (applies to x86 based systems that will run multiple operating systems)** – See "Preparing a Boot Environment for Installing Multiple Operating Systems" in *Installing Oracle Solaris 11.1 Systems*.
- **Ensure that you have the proper device drivers** – Before installing Oracle Solaris 11, determine whether the devices on your system are supported. You can use the Device Driver Utility to ensure that your system has the proper devices. The Device Driver Utility can be accessed through the text installer menu options. See "Ensuring That You Have the Proper Device Drivers" in *Installing Oracle Solaris 11.1 Systems*. See also the Hardware Compatibility Lists (HCL) at `http://www.oracle.com/webfolder/technetwork/hcl/index.html`.
- **x86: Configure the system's date and time (applies to x86 platforms that are installed with AI only)** – Oracle Solaris 11 keeps the Real Time Clock (RTC) in Coordinated Universal time (UTC) format. The behavior on x86 platforms is different than in Oracle Solaris 10. AI does not adjust the RTC date and time during an installation. To configure the date and time, see "Configuring Date and Time Before and After an Installation" on page 42.

Installing Oracle Solaris by Using Installation Media

Oracle Solaris can be installed by using any of the following installation methods:

- **x86: Live Media**

  The installer on the Live Media ISO image is for x86 platforms *only*. The Live Media installs a GUI desktop. Also, the Live Media requires more memory than the text installer. The exact memory requirements vary for each system. See "Oracle Solaris Installation Requirements" on page 32.

  If you are installing on x86 platforms that will run multiple operating systems, you can partition your disk during the installation process. See "Partitioning Your System" in *Installing Oracle Solaris 11.1 Systems*.

  The GUI installer cannot upgrade your operating system. The default GUI installer settings are described in "Default Settings With the GUI Installer" in *Installing Oracle Solaris 11.1 Systems*.

  To install the OS by using either the Live Media or the text installer, download the installation media from:

  ```
  http://www.oracle.com/
  technetwork/server-storage/solaris11/downloads/index.html
  ```

  You can copy the downloaded image to removable media, such as a USB drive or burn it to a DVD by using the `usbcopy` utility. To use the `usbcopy` utility, you must first install the `pkg:/install/distribution-constructor` package. See "How to Perform a GUI Installation" in *Installing Oracle Solaris 11.1 Systems*.

- **Interactive text installer**

  The text installation media contains a set of software that is more appropriate for a general-purpose server. The text installer can perform an installation on an existing Oracle Solaris x86 partition or on a SPARC slice. Or, the text installation can use the entire disk. If the whole disk option is chosen, a partition or slice is created to cover the targeted device. In either case, the installation overwrites everything on the targeted partition or slice. See "How to Perform a Text Installation" in *Installing Oracle Solaris 11.1 Systems*. If you use the text installer, you might have to install additional software packages afterwards. See "Adding Software After Text Installation" in *Installing Oracle Solaris 11.1 Systems*.

  If you are set up to perform an automated installation over the network, you can also perform an interactive text installation over the network. When using this method, you can only install a single system at a time. However, you can modify installation specifications by using the interactive selections. See "How to Perform a Text Installation Over the Network" in *Installing Oracle Solaris 11.1 Systems*.

- **Automated installations that boot from media**

  You can boot an AI image from media or a USB device to initiate a hands-free installation of just that system. An AI manifest provides installation instructions for the system. See "Creating a Custom AI Manifest" in *Installing Oracle Solaris 11.1 Systems*. The system must

have the minimum amount of required memory and adequate disk space. Also, the system must have network access so that software packages can be retrieved from an IPS repository on the Internet or on the local network. This step is required to complete the installation. See "Installing Using AI Media" in *Installing Oracle Solaris 11.1 Systems*.

For more information, see "Installing Oracle Solaris by Using AI" on page 38.

You can also create custom Live Media images, text installer images, and AI images. See *Creating a Custom Oracle Solaris 11.1 Installation Image*.

Note – After your system is installed, it cannot be updated by using a method that is similar to Solaris 10 upgrade methods. An Oracle Solaris 11 system is updated based on your desired maintenance schedule by using the `pkg` utility. See "Installing and Updating Packages" in *Oracle Solaris Administration: Common Tasks*. See also Updating to Oracle Solaris 11.2.

The following installation features are no longer supported:

- **Oracle Solaris Flash Archive Installation** – You can restore root pool snapshots to recover from a failed system or devices. See "System Boot, Recovery, and Platform Changes" on page 121.
- **JumpStart feature of Oracle Solaris** – The Automated Installer replaces this feature. See *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.1 Automated Installer*.

Oracle Solaris 11.1 Installation Media Paths

The media paths for the Oracle Solaris 11.1 installers are as follows:

x86 only: Live Media	`Oracle_Solaris-11_1-Live-X86`
SPARC: Interactive text installer	`Oracle_Solaris-11_1-Text-SPARC`
x86: Interactive text installer	`Oracle_Solaris-11_1-Text-X86`
SPARC: Automated Installer	`Oracle_Solaris-11_1-AI-SPARC`
x86: Automated Installer	`Oracle_Solaris-11_1-AI-X86`

Migrating From JumpStart to AI

The Automated Installer (AI) performs automated installations of networked systems. This installation method replaces the JumpStart installation method that is used in Oracle Solaris 10. For a detailed comparison of the two installation methods, see *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.1 Automated Installer*.

JumpStart to AI Migration Tasks

The `js2ai` utility can be used to assist you in migrating from JumpStart to AI. The utility is used to convert Oracle Solaris 10 JumpStart rules, profiles, and `sysidcfg` files to an AI manifest and system configuration files. See the `js2ai`(1M) man page for more details.

To use the `js2ai` utility, install the software package:

```
# pkg install install/js2ai
```

You can use the `js2ai` utility to perform most of the following tasks:

- **Replace JumpStart rules and profile files with AI criteria files and AI manifests**

  AI uses client criteria to specify which AI manifest files each client system should use to complete an installation. When adding an AI manifest to an AI install service, the criteria can be specified on the command line or in a file. AI uses client criteria to specify which client systems should use which AI manifest files to complete their installation. Use the `js2ai` utility with the `-r` option to convert both JumpStart rules and their associated profiles to AI criteria and manifests. Initially, you can use the `-S` option to skip validation:

  ```
  # /user/sbin/js2ai -rS [-d sysidcfg-dir] [-D destination-dir]
  ```

  See "Using js2ai To Convert JumpStart Rules and Profiles to AI Criteria and Manifests" in *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.1 Automated Installer*.

- **Convert JumpStart files to an AI configuration files**

  AI system configuration profiles are SMF XML profile files that specify system configuration. Use the `js2ai` utility with the `-s` option to convert any `sysidcfg` files that are associated with this JumpStart configuration to system configuration profile files. Initially, use the `-S` option to skip validation:

  ```
  # /user/sbin/js2ai -sS [-d sysidcfg-dir] [-D destination-dir]
  ```

  See "Using js2ai to Convert sysidcfg Files to System Configuration Profiles" in *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.1 Automated Installer*.

- **Set up an install server**

  You can use the same Oracle Solaris 11 server as both a JumpStart install server and an AI install server. However, JumpStart can only be used to install Oracle Solaris 10, not Oracle Solaris 11. See Chapter 4, "Installing Oracle Solaris 10 Using JumpStart on an Oracle Solaris 11 Server," in *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.1 Automated Installer*.

Note – Customers who have a valid My Oracle Support contract can now set up an Oracle Solaris 10 1/13 system as an AI install server by installing extra software packages. This offering enables an installation of the Oracle Solaris 11 11/11 release *only*. For more information, go to `https://support.oracle.com/`.

- **Dynamically derive an AI provisioning manifest**

  As an alternative to creating custom AI manifests, you can write a script that dynamically creates an AI manifest for each client at client installation time. This manifest is called a *derived manifest* because it is based on attributes of each client that is discovered at installation time. JumpStart begin scripts provide the ability to dynamically manipulate installation parameters that are passed to the installer. See "Creating an AI Manifest at Client Installation Time" in *Installing Oracle Solaris 11.1 Systems*

- **Access a software package repository for AI installations**

  After accessing the package repository (`http://pkg.oracle.com/solaris/release`) for AI installations, make a local copy of the package repository. See Chapter 2, "Copying IPS Package Repositories," in *Copying and Creating Oracle Solaris 11.1 Package Repositories*.

- **Provide system configuration instructions**

  You can use system configuration profiles to specify the configuration for the client system as a set of configuration parameters, in the form of a Service Management Facility (SMF) profile. This profile is applied during the first boot of the system after an AI installation. If you do not provide a system configuration profile for a particular client, the interactive configuration tool opens on that client during the installation. You can create a system configuration profile manually or by running the interactive configuration tool and saving the output to a file. For example, the following command creates a valid profile in `sc.xml` from responses that you provide interactively:

  ```
  # sysconfig create-profile -o sc.xml
  ```

  See Chapter 11, "Configuring the Client System," in *Installing Oracle Solaris 11.1 Systems*.

- **Create an SMF service that runs on first boot and executes a user-defined script**

  For any additional installation or configuration that cannot be accomplished by using an AI manifest or a system configuration profile, you can create a script that is executed at first boot by a run-once SMF service. See Chapter 13, "Running a Custom Script During First Boot," in *Installing Oracle Solaris 11.1 Systems*.

Installing Oracle Solaris by Using AI

The AI installation method can be used to perform a hands-free installation of Oracle Solaris.

Keep the following key points in mind:

- You can use AI to install single or multiple clients over the network.
- An AI server provides multi-platform installation support. However, you must create a separate install service for each client architecture (SPARC and x86) that you plan to install.
- Clients must be able to access an Oracle Solaris Image Packaging System (IPS) software package repository to retrieve the required software packages for the installation.
- The location of the IPS package repository, which is specified by a Universal Resource Identifier (URI), can be on the install server, on a server that is on the local network, or on the Internet. See "Configuring Publishers" in *Adding and Updating Oracle Solaris 11.1 Software Packages*.
- Clients can optionally be customized with specific installation parameters, for example, disk layout and software selection
- Clients can be optionally customized with specific system configuration parameters, for example, host name, network configuration, and user account information.
- Customizations can be made on a client-by-client basis and can also be scaled for large enterprise environments.

The AI process follows this general sequence:

1. The client system is booted from the network and retrieves its network configuration and the location of the install server from the DHCP server. SPARC clients can optionally get network configuration and location of the install server from by setting the `network-boot-arguments` variable in the Open Boot PROM (OBP).

   Note – The Reverse Address Resolution Protocol (RARP) does not work for booting and installing a system over the network with AI.

2. The install server provides a boot image to the client.
3. Characteristics of the client determine which installation instructions and which system configuration instructions are used to install the client.
4. Oracle Solaris 11 is installed on the client by pulling packages from the package repository that is specified by the installation instructions in the AI install service that you create.

AI Preinstallation Tasks

Prior to installing a system with AI, you must perform certain tasks. At minimum, you must set up an AI install server and create at least one install service. This scenario works well in situations where all of the clients are of the same architecture and will be installed with the same version of the Oracle Solaris OS. This type of installation uses the default AI manifest, which is not associated with any client criteria. When you create a new AI install service, */install-service-image-path*`/auto_install/manifest/default.xml` is the initial default AI manifest for that install service. In Oracle Solaris 11.1, the default AI manifest specifies the most recent version of the Oracle Solaris 11.1 release that is available from the IPS package repository (`http://pkg.oracle.com/solaris/release`).

AI uses DHCP to provide the IP address, subnet mask, router, name service server, and the location of the install server to the client machine to be installed. SPARC clients can optionally get their network configuration and install server location from the `network-boot-arguments` variable that is set in the OpenBoot PROM (OBP). Note that the DHCP server and AI install server can be the same system or two different systems. For more information about setting up an install server, see Chapter 8, "Setting Up an Install Server," in *Installing Oracle Solaris 11.1 Systems*.

For more information about the minimum set of tasks that must be completed to use AI, see "Minimum Requirements for AI Use" in *Installing Oracle Solaris 11.1 Systems*.

For a basic SPARC AI installation example that follows the minimum requirements for using AI, see Appendix A, "SPARC Automated Installation Scenario."

For additional information about customizing AI installations, provisioning client systems, and configuring client systems, refer to the following documentation:

- Chapter 9, "Customizing Installations," in *Installing Oracle Solaris 11.1 Systems*
- Chapter 10, "Provisioning the Client System," in *Installing Oracle Solaris 11.1 Systems*
- Chapter 11, "Configuring the Client System," in *Installing Oracle Solaris 11.1 Systems*

Setting Up an Install Client

When you initially set up your install server, you created at least one install service for each client architecture and for each version of Oracle Solaris that you plan to install. For each install service that you create for the different client architectures, you must also create customized installation instructions and system configuration instructions. Each client is then directed to the AI install server to access the information for the correct install service, as well as the AI manifest, and the system configuration profiles within that install service. If adequate system

configuration instructions are not provided prior to the installation, an interactive tool opens during the first boot after an installation, prompting you to provide the missing system configuration information.

Setting up an install client requires you to run the `installadm create-client` command on the install server, which associates a particular client with a particular install service. For example, you would set up a SPARC install client and associate the client with the MAC address `00:14:4f:a7:65:70` and the `solaris11_1-sparc` install service, as follows:

```
# installadm create-client -n solaris11_1-sparc -e 00:14:4f:a7:65:70
```

In this particular example, the DHCP server does not require configuration because the SPARC `wanboot-cgi` boot file has already been configured by using the `create-service` command. See "Creating an AI Install Service" in *Installing Oracle Solaris 11.1 Systems*.

Confirm that the client was added successfully by checking the `/etc/netboot` directory:

```
lrwxrwxrwx  1 root staff 33 2012-05-09 08:53 0100144FA76570 -> /etc/netboot/solaris11_1-sparc
```

EXAMPLE 2–1 Setting Up an x86 Installation Client

The following example associates the x86 client with the MAC address `0:e0:81:5d:bf:e0` and the `solaris11_1-i386` install service. The DHCP configuration that is shown in the output of this command must be manually added to the DHCP server. Otherwise, the client system cannot boot the `solaris11_1-i386` install service.

```
# installadm create-client -n solaris11_1-i386 -e 0:e0:81:5d:bf:e0
No local DHCP configuration found. If not already configured, the
following should be added to the DHCP configuration:
    Boot server IP      : 10.80.239.5
    Boot file(s)        :
        bios clients (arch 00:00):  0100E0815DBFE0.bios
        uefi clients (arch 00:07):  0100E0815DBFE0.uefi
```

In the following example, the `installadm create-client` command is used to set the default PXE boot files for an x86 client in the `/etc/inet/dhcpd4.conf` file of the ISC DHCP configuration for an Oracle Solaris 11.1 `i386` install service:

```
host 00E0815DBFE0 {
  hardware ethernet 00:E0:81:5D:BF:E0;
  if option arch = 00:00 {
    filename "0100E0815DBFE0.bios";
  } else if option arch = 00:07 {
    filename "0100E0815DBFE0.uefi";
  }
}
```

Booting the Client and Initiating an Oracle Solaris Installation

After performing the required prerequisite tasks for using AI, plus any optional customization tasks, you are ready to install the client system. The installation begins when you boot the client system over the network.

Boot a SPARC client, as follows:

1. Bring the system to the `ok` PROM prompt, then boot the system.

   ```
   ok boot net:dhcp - install
   ```

   Note – The syntax for booting a SPARC based system from the network has changed in Oracle Solaris 11.

   If you are *not* using DHCP, use this command:

   ```
   ok setenv network-boot-arguments host-ip=client-ip,
   router-ip=router-ip,subnet-mask=subnet-mask,hostname=hostname,
   file=wanboot-cgi-file
   ```

   When you use the `network-boot-arguments` variable, the SPARC client does *not* have DNS configuration information. Ensure that the AI manifest that is used with the client specifies an IP address instead of a host name for the location of the IPS package repository, and for any other URI in the manifest.

2. Boot the system.

   ```
   ok boot net - install
   ```

   See "Installing a SPARC Client" in *Installing Oracle Solaris 11.1 Systems* for a list of the events that occur during a SPARC client installation.

Perform a PXE boot of an x86 client, as follows:

1. Boot the client system.
2. When the client boots, instruct the firmware to boot from the network by typing the specific keystroke sequence when the firmware screen (BIOS or UEFI) is displayed.

   For information about UEFI firmware support on x86 platforms, see"Booting Systems With UEFI and BIOS Firmware From the Network" in *Booting and Shutting Down Oracle Solaris 11.1 Systems*.

3. When the GRUB menu is displayed, select the second entry (Automated Install), then press Return to install that image.

   ```
   Oracle Solaris 11.1 Text Installer and command line
   Oracle Solaris 11.1 Automated Install
   ```

See "Installing an x86 Client" in *Installing Oracle Solaris 11.1 Systems* for a list of the events that occur during an x86 client installation.

Information About Installing and Configuring Zones

Non-global zones are installed and configured on the first reboot after the global zone is installed. With AI, non-global zones can be installed on the system by using the configuration element that is defined in the AI manifest. During the first boot after the global zone installation, the zone's self-assembly SMF service (`svc:/system/zones-install:default`) configures and installs each non-global zone that is defined in the global zone AI manifest. If the zone is configured with the `auto-boot` property set to `true` (`autoboot=true`), the `system/zones-install` service boots the zone after it is installed. See Chapter 12, "Installing and Configuring Zones," in *Installing Oracle Solaris 11.1 Systems*.

Download Locations for AI Files

During an AI installation, several important AI files are downloaded to the following locations:

Installation log file:
`/system/volatile/install_log`

AI client manifest that is downloaded from the AI server:
`/system/volatile/ai.xml`

AI client derived manifest (if used):
`/system/volatile/manifest.xml`

System configuration profiles downloaded from the AI server during installation:
`/system/volatile/profile/*`

List of AI services:
`/system/volatile/service_list`

Additional Installation Tasks

You might need to perform the following additional tasks before or after an installation.

Configuring Date and Time Before and After an Installation

Oracle Solaris 11 keeps the Real Time Clock (RTC) in Coordinated Universal time (UTC) format. The behavior on x86 platforms is different in Oracle Solaris 11 than in Oracle Solaris 10.

The interactive installers enable you to configure the date and time during the installation. As part of that process, the RTC is updated with the time in UTC format. However, AI does *not* adjust the RTC date and time during an installation. To ensure that the time stamp of installed files are correct, configure the time in the BIOS in UTC format *before* beginning the installation. On x86 platforms, when using the `pkg update` command, the OS continues to keep time in RTC in the local time format. This method is used to avoid time inconsistencies between Oracle Solaris 11 BEs and BEs from previous releases.

Note – If you are running Oracle Solaris 11 as an Oracle VM VirtualBox guest, you need to check or uncheck the Hardware Clock in UTC time setting in the system preferences for the virtual machine.

▼ How to Switch From Local Time Format to UTC Format

1. **Set the time lag between the kernel and RTC to 0 (zero), as follows:**

   ```
   # rtc -z GMT
   ```

2. **If the date/time requires an adjustment, use the `date` command. See `date(1)`.**

▼ How to Switch From UTC Format to Local Time Format

Use the following procedure when the switch from UTC to local time is complete, and each time you reconfigure the time zone setting by using the `sysconfig` command.

1. **Run the `rtc` *timezone* command with the `-z` option.**

   ```
   # rtc -z timezone
   ```

   For example:

   ```
   # rtc -z US/Pacific
   ```

2. **If the date/time requires an adjustment, use the `date` command.**

Maintaining Local Time on a System Running Multiple Operating Systems That Keep RTC Time as Local Time

If you maintain and boot several operating systems on the same Oracle Solaris 11 system, and those operating systems keep RTC time as local time, there are several ways that these operating systems can coexist, from the RTC time point of view:

- Switch from local time to UTC format in the OS that keeps RTC time in local time format.

  For example, if you are dual-booting Windows 7, set the registry key as follows:

  ```
  [HKEY_LOCAL_MACHINESYSTEM\CurrentControlSet\Control\TimeZoneInformation] \
  "RealTimeIsUniversal"=dword:00000001
  ```

- Switch from the UTC format to local time on a freshly installed Oracle Solaris 11 system.
- Enable the Network Time Protocol (NTP) in operating systems that assume that the RTC format is running in local time. In this case, the time is synchronized automatically.

x86: Adding Custom Entries to the GRUB Menu After an Installation

In Oracle Solaris 11, the GRUB Legacy boot loader uses the `menu.lst` file to maintain both Oracle Solaris and custom menu entries, for example a Linux menu entry. After installing Oracle Solaris, any custom menu entries that were not preserved during the installation can be manually added to the GRUB menu by editing the `menu.lst` file.

Starting with Oracle Solaris 11.1, GRUB (GRUB 2) uses a different boot loader and a different configuration file, `grub.cfg`. This file contains most of the GRUB configuration, including all of the Oracle Solaris menu entries. The file does not contain any custom menu entries. Unlike the `menu.lst` file, the `grub.cfg` file is managed *solely* by using the `bootadm` command. Do *not* directly edit this file. GRUB 2 includes an additional configuration (`custom.cfg`) that can be used to add custom menu entries to the GRUB menu after an installation. If you want to add custom boot entries to the GRUB configuration, you must first create a `custom.cfg` file and it must reside in the same location as the `grub.cfg` and `menu.conf` files, */pool-name*`/boot/grub/`.

During the boot process, GRUB checks for the existence of a `custom.cfg` file in the top-level dataset of the root pool, in the `boot/grub` subdirectory. If the file exists, GRUB sources the file and processes any commands that are in the file, as if the contents were textually inserted in the main `grub.cfg` file.

On a system with 64-bit UEFI firmware, entries in the `custom.cfg` file might appear as follows:

```
menuentry "Windows (64-bit UEFI)" {
   insmod part_gpt
   insmod fat
   insmod search_fs_uuid
   insmod chain
   search --fs-uuid --no-floppy --set=root cafe-f4ee
   chainloader /efi/Microsoft/Boot/bootmgfw.efi
}
```

On a system with BIOS firmware, entries in this file might appear as follows:

```
menuentry "Windows" {
   insmod chain
   set root=(hd0,msdos1)
   chainloader --force +1
}
```

See "Customizing the GRUB Configuration" in *Booting and Shutting Down Oracle Solaris 11.1 Systems*.

Troubleshooting an Oracle Solaris Installation

Refer to the following troubleshooting information for issues that you might encounter during or after installing an Oracle Solaris 11 release:

- "What to Do If Your System Boots in Console Mode" in *Installing Oracle Solaris 11.1 Systems*
- Chapter 15, "Troubleshooting Automated Installations," in *Installing Oracle Solaris 11.1 Systems*

If you encounter login or password issues after installing an Oracle Solaris 11 release, see "Booting for System Recovery" on page 123.

Monitoring the Live Media Startup Process

Switching to the text boot screen is useful if you suspect that the system startup process is not proceeding normally. The text screen might contain informational messages or a request for user input. Switching to the text boot screen has no impact on the boot sequence, other than how the information is displayed on the screen. Initialization of the operating system continues and completes as normal.

To switch to a text boot, press any key a few seconds after the GUI boot screen appears and the progress animation begins. Note that after switching from the GUI boot to a text boot, you cannot switch back to the GUI boot screen.

◆ ◆ ◆ CHAPTER 3

Managing Devices

This chapter provides information about managing devices in Oracle Solaris 11 releases.

The following topics are covered:

Device Identity and Configuration Changes

As in Oracle Solaris 10 releases, all supported devices that are connected to the system when it is installed should be accessible after installation. Devices can be configured with the `cfgadm` command and most devices are hot-pluggable, which means you can add and remove devices while the system is booted.

The new `hotplug` command is available for PCI Express (PCIe) and PCI SHPC (Standard Hot Plug Controller) devices that provides offline and online capabilities as well as enable and disable operations. You would still use the `cfgadm` command to manage hot pluggable USB and SCSI devices as in previous Oracle Solaris releases.

For more information, see Chapter 4, "Dynamically Configuring Devices (Tasks)," in *Oracle Solaris 11.1 Administration: Devices and File Systems*.

In this release, you can more easily identify devices by using the `croinfo` command to identify physical device location information.

You can use the following commands to display information by chassis, receptacle, and occupant values for the devices on your system:

- `diskinfo` – Display general information about physical disk locations

- `format` – Display physical disk location information for disks when reviewing partition tables or relabeling. For example, the following `format` output identifies the two internal disks on this system, under `/dev/chassis/SYS/HDO` and `/dev/chassis/SYS/HD1`.

  ```
  # format
  Searching for disks...done

  AVAILABLE DISK SELECTIONS:
         0. c1t0d0 <FUJITSU-MAY2073RCSUN72G-0401 cyl 8921 alt 2 hd 255 sec 63>
            /pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@0,0
            /dev/chassis/SYS/HD0/disk
         1. c1t1d0 <FUJITSU-MAY2073RCSUN72G-0401-68.37GB>
            /pci@0,0/pci1022,7450@2/pci1000,3060@3/sd@1,0
            /dev/chassis/SYS/HD1/disk
  ```

  The above output identifies two internal system disks but disks from a storage array are generally identified by their storage array name.

- `prtconf -l` – Display system configuration information that includes physical disk location information
- `zpool status -l` – Display physical disk location information for pool devices

In addition, you can use the `fmadm add-alias` command to include a disk alias name that helps you identify the physical location of disks in your environment. For example:

```
# fmadm add-alias SUN-Storage-J4200.0912QAJ001 J4200@RACK10:U26-27
# fmadm add-alias SUN-Storage-J4200.0905QAJ00E J4200@RACK10:U24-25
```

Use the `diskinfo` command to determine where a disk is located:

```
% diskinfo -c c0t24d0
D:devchassis-path                                 t:occupant-type  c:occupant-compdev
------------------------------------------------  ---------------  ------------------
/dev/chassis/J4200@RACK10:U26-27/SCSI_Device__9/disk  disk             c0t24d0
```

In this example, the `/dev/chassis` disk name includes an alias name that helps you locate the device in your environment.

The following `diskinfo` example shows how to display a specific disk's physical location.

```
$ diskinfo -c c0t24d0 -o cp
c:occupant-compdev  p:occupant-paths
------------------  ---------------------------------------------------------
c0t24d0             /devices/pci@0,600000/pci@0/pci@9/LSILogic,sas@0/sd@18,0
```

Note – The `diskinfo` command require that chassis support SES diagnostic page 0xa (Additional Element Status) and must set the Element Index Present (EIP) bit to 1. Enclosures that do not meet this criteria will not be fully enumerated, and thus, will not be properly represented.

Changes to Device Driver Customization

In Oracle Solaris 11, driver customizations are made in the `/etc/driver/drv` directory rather than in the `/kernel directory` as in previous releases. This improvement means that your driver customizations are not overwritten when the system is upgraded. The files in the `/etc/driver/drv` directory are preserved during the upgrade. Customizing a driver configuration usually means that a per-device parameter or global property that impacts all devices is added or modified.

For more information, see "How to Customize a Driver Configuration" in *Oracle Solaris 11.1 Administration: Devices and File Systems*.

Preparing Disks for ZFS Storage Pools

Creating ZFS storage pools in Oracle Solaris 11 is similar to creating pools in Oracle Solaris 10. The following sections provide summary information about preparing disks for a ZFS root pool and non-root pools.

Review the following general pool device configuration recommendations:

- Create non-root pools by using whole disks, which are easier to manage than disk slices. For example, you can easily create a mirrored storage pool with four devices as follows:

  ```
  # zpool create tank mirror c0t1d0 c0t2d0 mirror c1t1d0 c1t2d0
  ```

- When ZFS storage pools are created with whole disks, the disks are labeled with an EFI label rather than an SMI label. You can identify an EFI label by the lack of cylinder information in the disk label as displayed in the format utility. For example:

  ```
  partition> print
  Current partition table (original):
  Total disk sectors available: 286478269 + 16384 (reserved sectors)

  Part      Tag    Flag     First Sector         Size         Last Sector
    0        usr    wm               256      136.60GB          286478302
    1 unassigned    wm                 0           0               0
    2 unassigned    wm                 0           0               0
    3 unassigned    wm                 0           0               0
    4 unassigned    wm                 0           0               0
    5 unassigned    wm                 0           0               0
    6 unassigned    wm                 0           0               0
    8   reserved    wm         286478303        8.00MB          286494686
  ```

- We recommend that you create non-root pools with whole disks.

Oracle Solaris releases support advanced format disks in addition to traditional 512n disks. For more information, see "Advanced Format Disk Support " in *Oracle Solaris 11.1 Administration: Devices and File Systems*.

ZFS Root Pool Installation Improvements

Review the following installation improvements for root pools:

- **Disk label improvements** – If the disk label or labels that are intended to contain the OS are unknown, the disks will be automatically relabeled with the appropriate disk label.

  In Oracle Solaris 11.1, SPARC based systems with GPT enabled firmware and most x86 based systems are installed with an EFI (GPT) label on the root pool disk or disks.

  In addition, the AI installer has improved the `whole_disk` keyword syntax so that if `whole_disk` is set to `true`, the disk's contents are replaced, even if it has existing partitions or slices.

- **AI installation of a mirrored root pool** – Oracle Solaris 10 installation features allow you to create a mirrored root pool during installation.

  You can use AI manifest keyword syntax to create a mirrored root pool during an Oracle Solaris 11 automatic installation. For example, the following Oracle Solaris 11.1 syntax creates a mirrored root pool using whole disks:

```
<!DOCTYPE auto_install SYSTEM "file:///usr/share/install/ai.dtd.1">
.
.
.
   <target>
        <disk whole_disk="true" in_zpool="rpool" in_vdev="mirrored">
        <disk_name name="c1t0d0" name_type="ctd"/>
        </disk>
        <disk whole_disk="true" in_zpool="rpool" in_vdev="mirrored">
        <disk_name name="c2t0d0" name_type="ctd"/>
        </disk>
      <logical>
        <zpool name="rpool" is_root="true">
        <vdev name="mirrored" redundancy="mirror"/>
         <!--
            Subsequent <filesystem> entries instruct an installer to create
            following ZFS datasets:

                <root_pool>/export         (mounted on /export)
                <root_pool>/export/home    (mounted on /export/home)
.
.
.       </zpool>
      </logical>
     </target>
.
.
.
```

ZFS Root Pool Device Requirements

In general, root pool devices are relabeled and the root pool is created when the system is installed.

- **Oracle Solaris 11:** An SMI (VTOC) label is applied automatically to the root pool disk or disks during installation on both SPARC and x86 based systems.

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: none requested
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       ONLINE       0     0     0
          c7t0d0s0  ONLINE       0     0     0
```

- **Oracle Solaris 11.1:** An EFI label is applied automatically to the root pool disk or disks during installation on SPARC based systems with GPT enabled firmware and most x86 based systems. Otherwise, a VTOC disk label is installed on the root pool disk, as shown in the following example:

```
# zpool status rpool
  pool: rpool
 state: ONLINE
  scan: none requested
config:

        NAME      STATE     READ WRITE CKSUM
        rpool     ONLINE       0     0     0
          c7t0d0  ONLINE       0     0     0
```

  When you attach a disk to create a mirrored root pool, use the whole disk syntax.

```
# zpool attach rpool c7t0d0 c7t2d0
Make sure to wait until resilver is done before rebooting.
```

  The pool remains in a `DEGRADED` state until the new disk is resilvered.

```
# zpool status rpool
  pool: rpool
 state: DEGRADED
status: One or more devices is currently being resilvered.  The pool will
        continue to function in a degraded state.
action: Wait for the resilver to complete.
        Run 'zpool status -v' to see device specific details.
  scan: resilver in progress since Thu Jan 24 08:15:13 2013
    224M scanned out of 22.0G at 6.59M/s, 0h56m to go
    221M resilvered, 0.99% done
config:

        NAME        STATE     READ WRITE CKSUM
        rpool       DEGRADED     0     0     0
          mirror-0  DEGRADED     0     0     0
            c7t0d0  ONLINE       0     0     0
            c7t2d0  DEGRADED     0     0     0  (resilvering)
```

- The pool must exist either on a disk slice or on disk slices that are mirrored. If you attempt to use an unsupported pool configuration during an `beadm` operation, you will see a message similar to the following:

```
ERROR: ZFS pool name does not support boot environments
```

- On an x86 based system, the disk must contain an Oracle Solaris `fdisk` partition. An Oracle Solaris `fdisk` partition is created automatically when the x86 based system is installed. For more information about `fdisk` partitions, see "Guidelines for Creating an fdisk Partition" in *Oracle Solaris 11.1 Administration: Devices and File Systems*.

For more general information about creating ZFS root pools, see "ZFS Storage Pool Creation Practices" on page 56.

ZFS Root Pool Disk and Boot Administration

ZFS root pool disk and boot administration summary is as follows:

- **Oracle Solaris 10 and Oracle Solaris 11**
  - **SPARC:** (OBP) PROM needs root pool disk with an SMI (VTOC) label.
  - **SPARC:** If replacing a root pool disk with `zpool replace`, apply boot blocks manually.

```
# installboot -F zfs /usr/platform/‘uname -i‘/lib/fs/zfs/bootblk /dev/rdsk/c1t0d0s0
```

  - **SPARC and x86:** Attaching a root pool disk with `zpool attach` to create a mirrored root pool requires the slice syntax.

```
# zpool attach rpool c0t5000CCA03C5A5314d0s0 c0t5000CCA03C5A5340d0s0
```

    If you attempt to attach a disk with an EFI label to a root pool disk that requires an SMI (VTOC) label, you will need to relabel it manually before it can be attached.

```
# format -L vtoc -d c1t0d0
Searching for disks...done
selecting c1t0d0
[disk formatted]
c1t0d0 is labeled with VTOC successfully.
```

    Be very careful that you are relabeling the correct disk because this command does no error checking. If you force an SMI (VTOC) label on a disk that is intended for the root pool, the default partition table is applied. This means that the default `s0` slice size might be too small. For more information about changing partition or slice sizes, see "How to Label a Disk" in *Oracle Solaris 11.1 Administration: Devices and File Systems*.
  - **x86:** GRUB legacy and root pool disk needs an SMI (VTOC) label.
  - **x86:** If replacing a root pool disk with `zpool replace`, apply the boot blocks manually.

```
# installgrub /boot/grub/stage1 /boot/grub/stage2 /dev/rdsk/c1t0d0s0
```

  - **x86:** Root pool disk must be less than 2 TBs.
- **Oracle Solaris 11.1**
  - **SPARC:** OBP requires a root pool disk with an SMI (VTOC) label.
  - **SPARC:** If replacing a root pool disk with `zpool replace`, apply boot blocks manually.

```
# bootadm install-bootloader
```

- **SPARC:** Attaching a root pool disk with `zpool attach` to create a mirrored root pool requires the slice syntax.

  ```
  # zpool attach rpool c0t5000CCA03C5A5314d0s0 c0t5000CCA03C5A5340d0s0
  ```

- **x86:** GRUB 2 and root pool disk has an EFI label in most cases.
- **x86:** If replacing a root pool disk with `zpool replace`, apply the boot blocks manually.

  ```
  # bootadm install-bootloader
  ```

- **x86:** Attaching a root pool disk with `zpool attach` to create a mirrored root pool requires the whole disk syntax.

  ```
  # zpool attach rpool c0t5000CCA03C5A5314d0 c0t5000CCA03C5A5340d0
  ```

- **Current Oracle Solaris 10 and 11 Releases**

  Using the `zpool attach` command applies the boot blocks automatically.

Swap and Dump Device Configuration Changes

In Oracle Solaris 10 releases, a UFS root environment provides one disk slice for both swap and dump devices. After an Oracle Solaris 11 system is installed, two separate volumes are created as a swap device and a dump device.

```
# dumpadm
      Dump content: kernel pages
       Dump device: /dev/zvol/dsk/rpool/dump (dedicated)
Savecore directory: /var/crash
  Savecore enabled: yes
   Save compressed: on

# swap -l
swapfile             dev    swaplo   blocks     free
/dev/zvol/dsk/rpool/swap 182,2         8  4061176  4061176
```

Display information about the swap and dump volume names and sizes. For example:

```
# zfs list -t volume -r rpool
NAME         USED  AVAIL  REFER  MOUNTPOINT
rpool/dump  4.13G  51.6G  4.00G  -
rpool/swap  4.13G  51.6G  4.00G  -
```

You can also display swap space sizes in human-readable format. For example:

```
# swap -sh
total: 1.4G allocated + 227M reserved = 1.6G used, 432G available
# swap -lh
swapfile                   dev    swaplo   blocks     free
/dev/zvol/dsk/rpool/swap 285,2        8K     4.0G     4.0G
```

The management of ZFS swap and dump volumes differs from the management of a single slice for a UFS swap and dump device in the following ways:

- You cannot use a single volume for both swap and dump devices in a ZFS root environment
- You cannot use a file as swap device in a ZFS root environment
- The system requires that the dump device is approximately 1/2 to 3/4 the size of physical memory. If the dump device is too small, you will see an error similar to the following:

```
# dumpadm -d /dev/zvol/dsk/rpool/dump
dumpadm: dump device /dev/zvol/dsk/rpool/dump is too small to hold a system dump
dump size 36255432704 bytes, device size 34359738368 bytes
```

You can easily increase the size of the dump device by increasing the volume's `volsize` property, but it might take some time to reinitialize the volume. For example:

```
# zfs get volsize rpool/dump
NAME        PROPERTY  VALUE  SOURCE
rpool/dump  volsize   1.94G  local
# zfs set volsize=3g rpool/dump
# zfs get volsize rpool/dump
NAME        PROPERTY  VALUE  SOURCE
rpool/dump  volsize   3G     local
```

Changing the size of the swap volume is difficult if the swap device is in use. Consider creating a second swap volume and adding it as a swap device. For example:

```
# zfs create -V 3G rpool/swap2
# swap -a /dev/zvol/dsk/rpool/swap2
# swap -l
swapfile             dev    swaplo   blocks     free
/dev/zvol/dsk/rpool/swap 182,2         8  4061176  4061176
/dev/zvol/dsk/rpool/swap2 182,4         8  6291448  6291448
```

Then, add an entry for the new swap device in the `/etc/vfstab` file. For example:

```
/dev/zvol/dsk/rpool/swap2    -        -       swap    -       no      -
```

◆ ◆ ◆ CHAPTER 4

Managing Storage Features

This chapter describes storage management changes in Oracle Solaris 11 releases.

The following topics are covered:

Comparison of Solaris Volume Manager Configurations to ZFS Configurations

In Oracle Solaris 10 releases, you could create redundant volumes for UFS file systems by using Solaris Volume Manager. Solaris Volume Manager is a traditional volume management product with a layer of volume management and a layer of file system management.

ZFS, available in Oracle Solaris 10 and Oracle Solaris 11 releases, eliminates volume management altogether. Instead of creating virtualized volumes, ZFS aggregates devices into a storage pool. The storage pool describes the physical characteristics of the storage (device layout, data redundancy, and so on) and acts as an arbitrary data store from which file systems can be created. File systems are no longer constrained to individual devices, allowing them to share disk space with all file systems in the pool.

In Oracle Solaris 11, you can easily create a redundant ZFS storage pool in one command. ZFS provides two types of redundant configurations, mirrored pools and RAID-Z pools. RAID-Z configurations have similar features to RAID-5.

ZFS dynamically stripes data across all non-redundant, mirrored, and RAID-Z configurations.

- Solaris Volume Manager RAID-0 (stripe and concatenation) is not available in ZFS RAID-Z configurations

- Solaris Volume Manager RAID-1 (mirror) is available as a ZFS mirrored configuration. For example:

```
# zpool create tank mirror c1t0d0 c2t0d0 mirror c1t1d0 c2t1d0
```

- Solaris Volume Manager RAID-5 (distributed parity) is available as a ZFS RAID-Z (`raidz1`) configuration. For example:

```
# zpool create rzpool raidz1 c1t0d0 c2t0d0 c1t1d0 c2t1d0
```

- Solaris Volume Manager does not offer a RAID-6 but ZFS provides both RAIDZ-2 and RAIDZ-3 parity configurations, which means a RAIDZ-2 configuration can withstand the failure of two disks and a RAIDZ-3 configuration can withstand the failure of 3 disks. For example:

```
# zpool create rzpool raidz2 c0t1d0 c1t1d0 c4t1d0 c5t1d0 c6t1d0 c7t1d0
raidz2 c0t2d0 c1t2d0 c4t2d0 c5t2d0 c6t2d0 c7t2d0
```

Recommended ZFS Storage Pool Practices

ZFS uses a pool storage model where storage devices are aggregated into a storage pool. File systems within the storage pool use all the storage in the pool.

The following sections provide recommended practices for creating, monitoring, and troubleshooting ZFS storage pools.

ZFS Storage Pool Creation Practices

- **Specific root pool device and boot disk requirements**
  - "ZFS Root Pool Device Requirements" on page 50
  - "ZFS Root Pool Disk and Boot Administration" on page 52
- **General root pool creation practices**
  - The root pool must be created as a mirrored configuration or as a single-disk configuration. Neither a RAID-Z nor a striped configuration is supported. You cannot add additional disks to create multiple mirrored top-level virtual devices by using the `zpool add` command, but you can expand a mirrored virtual device by using the `zpool attach` command.
  - The root pool cannot have a separate log device.
  - Pool properties can be set during an AI installation by using the `pool_options` keyword syntax, but the `gzip` compression algorithm is not supported on root pools.
  - Do not rename the root pool after it is created by an initial installation. Renaming the root pool might cause an unbootable system.
  - Do not create a root pool on a USB stick for a production system because root pool disks are critical for continuous operation, particularly in an enterprise environment. Consider using a system's internal disks for the root pool, or at least use, the same quality

disks that you would use for your non-root data. In addition, a USB stick might not be large enough to support a dump volume size that is equivalent to at least 1/2 the size of physical memory.

- Consider keeping root pool components separate from non-root pool data.

- **Non-root pool creation practices** – Create non-root pools with whole disks by using the `d*` identifier. Do not use the `p*` identifier.
  - ZFS works best without any additional volume management software.
  - For better performance, use individual disks or at least LUNs made up of just a few disks. By providing ZFS with more visibility into the LUN setup, ZFS is able to make better I/O scheduling decisions.
  - **Mirrored storage pools** – Consume more disk space but generally perform better with small random reads. For example:

    ```
    # zpool create tank mirror c1d0 c2d0 mirror c3d0 c4d0
    ```

    Mirrored storage pools are also more flexible in that you can detach, attach, and replace existing devices in the pool.
  - **RAID-Z storage pools** – Can be created with 3 parity strategies, where parity equals 1 (`raidz`), 2 (`raidz2`), or 3 (`raidz3`).
    - A RAID-Z configuration maximizes disk space and generally performs well when data is written and read in large chunks (128K or more). Create a single-parity RAIDZ (`raidz`) configuration at 3 disks (2+1).
    - A RAIDZ-2 configuration offers better data availability, and performs similarly to RAID-Z. RAIDZ-2 has significantly better mean time to data loss (MTTDL) than either RAID-Z or 2-way mirrors. Create a double-parity RAID-Z (`raidz2`) configuration at 6 disks (4+2).
    - A RAIDZ-3 configuration maximizes disk space and offers excellent availability because it can withstand 3 disk failures. Create a triple-parity RAID-Z (raidz3) configuration at 8 disks (5+3).

- **Non-redundant pools** – If you create a non-redundant pool, you will see a message similar to the following:

  ```
  # zpool create pond c8t2d0 c8t3d0
  'pond' successfully created, but with no redundancy; failure of one
  device will cause loss of the pool
  ```

  Creating a pool with no redundancy is not recommended because a device failure could mean that the data is unrecoverable. Consider creating a ZFS storage pool with redundancy. For example:

  ```
  # zpool create pond mirror c8t2d0 c8t3d0
  ```

ZFS Storage Pool Monitoring Practices

- Make sure that pool capacity is below 90% for best performance. Monitor pool and file system space to make sure that they are not full. Consider using ZFS quotas and reservations to make sure file system space does not exceed 90% pool capacity.
- Run `zpool scrub` on a regular basis to identify data integrity problems:
  - If you have consumer-quality drives, consider a weekly scrubbing schedule.
  - If you have datacenter-quality drives, consider a monthly scrubbing schedule.
  - You should also run a scrub prior to replacing devices to ensure that all devices are currently operational.
- Use `zpool status` on a weekly basis to monitor pool and pool device status. Also use `fmdump` or `fmdump -eV` to see if any device faults or errors have occurred.

ZFS Storage Pool Troubleshooting Practices

Troubleshooting pool problems in Oracle Solaris 11 is similar to diagnosing problems in the Oracle Solaris 10 release, but review the following new diagnostic descriptions and features:

- Failed devices – Review the `zpool status -l` output to identify the physical location of the failed device and replace it. For information about replacing a failed disk, see "Replacing or Repairing a Damaged Device" in *Oracle Solaris 11.1 Administration: ZFS File Systems*.
- Failed device notification – The `smtp-notify` service can be configured to send electronic mail notifications in response to various fault management events, such as when a hardware component has been diagnosed as faulty. For more information, see the notification parameters section of `smf(5)`.

  By default, some notifications are set up automatically to be sent to the root user. If you add an alias for your user account as root in the `/etc/aliases` file, you will receive electronic mail notifications, similar to the following:

```
-------- Original Message --------
Subject: Fault Management Event: tardis:SMF-8000-YX
Date: Wed, 21 Sep 2011 11:11:27 GMT
From: No Access User <noaccess@tardis.drwho.COM>
Reply-To: root@tardis.drwho.COM
To: root@tardis.drwho.COM

SUNW-MSG-ID: ZFS-8000-D3, TYPE: Fault, VER: 1, SEVERITY: Major
EVENT-TIME: Wed Sep 21 11:11:27 GMT 2011
PLATFORM: Sun-Fire-X4140, CSN: 0904QAD02C, HOSTNAME: tardis
SOURCE: zfs-diagnosis, REV: 1.0
EVENT-ID: d9e3469f-8d84-4a03-b8a3-d0beb178c017
DESC: A ZFS device failed.  Refer to http://sun.com/msg/ZFS-8000-D3
for more information.
AUTO-RESPONSE: No automated response will occur.
IMPACT: Fault tolerance of the pool may be compromised.
REC-ACTION: Run 'zpool status -x' and replace the bad device.
```

- Moving devices – Devices that are part of a ZFS storage pool contain a device ID if the device driver creates or fabricates device IDs. Like all file systems, ZFS has a very close relationship with its underlying devices so if you attempt to upgrade a system's firmware, move a pool device to a different controller, or change a device's cabling, you might consider exporting the pool first. If the device ID does not follow the device change and this can happen with non-Oracle hardware, then the pool and pool data might become unavailable. In general, Oracle's Sun hardware can recover if a device is changed under a live pool because our drivers fully support device IDs, but you might consider exporting the pool before making any hardware changes.

For a full description of troubleshooting pool problems, see Chapter 10, "Oracle Solaris ZFS Troubleshooting and Pool Recovery," in *Oracle Solaris 11.1 Administration: ZFS File Systems*.

COMSTAR Replaces iSCSI Target Daemon

The Oracle Solaris 10 release uses the iSCSI target daemon and the `iscsitadm` command and the ZFS `shareiscsi` property to configure iSCSI LUNs.

In the Oracle Solaris 11 release, the COMSTAR (Common Multiprotocol SCSI Target) features provide the following components:

- Support for different types of SCSI targets, not just the iSCSI protocol
- ZFS volumes are used as backing store devices for SCSI targets by using one or more of COMSTAR's supported protocols.

Although the iSCSI target in COMSTAR is a functional replacement for the iSCSI target daemon, no upgrade or update path exist to convert your iSCSI LUNs to COMSTAR LUNs.

- Both the iSCSI target daemon and the `shareiscsi`property are not available in Oracle Solaris 11. The following commands are used to manage iSCSI targets and LUNs.
  - The `itadm` command manages SCSI targets.
  - The `srptadm` command manages SCSI RDMA Protocol (SRP) target ports.
  - The `stmfadm` command manages SCSI LUNs. Rather than setting a special iSCSI property on the ZFS volume, create the volume and use `stmfadm` to create the LUN.

For information about COMSTAR, see Chapter 11, "Configuring Storage Devices With COMSTAR (Tasks)," in *Oracle Solaris 11.1 Administration: Devices and File Systems*.

◆ ◆ ◆ CHAPTER 5

Managing File Systems

This chapter provides information about managing file systems in Oracle Solaris 11 releases.

The following topics are covered:

Oracle Solaris 11 File System Changes

The file systems that are available in the Oracle Solaris 11 release are very similar to the Oracle Solaris 10 file systems.

- Disk-based file systems – HSFS, PCFS, UDFS, UFS, and ZFS
- Network-Based file systems – NFS and SMB
- Virtual file systems – CTFS, FIFOFS, MNTFS, NAMEFS, OBJFS, SHAREFS, SPECFS, and SWAPFS
- Temporary file system (TMPFS)
- Loopback file system (LOFS)
- Process file system (PROCFS)

The general file system differences are as follows:

- CacheFS is not available in the Oracle Solaris 11 release.
- ZFS is the default root file system.
- UFS is a supported legacy file system, but it is not supported as a bootable root file system.
- The legacy Solaris Volume Manager product is supported, but you cannot boot from a Solaris Volume Manager root device.

- ZFS uses a separate ZFS volume for swap and dump devices. UFS can use a single slice for both swap and dump devices.

Root File System Requirements and Changes

The root file system hierarchy is mostly identical to systems running Solaris 10 that have a ZFS root file system. A ZFS root pool contains a ZFS file system with separate directories of system-related components, such as `etc`, `usr`, and `var`, that must be available for the system to function correctly.

- After a system is installed, the root of the Solaris file system is mounted, which means files and directories are accessible.
- All subdirectories of the root file system that are part of the Oracle Solaris OS, with the exception of `/var`, must be contained in the same file system as the root file system.
- During an Oracle Solaris 11 installation, a separate `/var` file system is created automatically for a global zone and a non-global zone.
- During an Oracle Solaris 11.1 installation, a `rpool/VARSHARE` file system is mounted at `/var/share`. The purpose of this file system is to share file systems across boot environments so that the amount of space that is needed in the `/var` directory for all BEs is reduced.

  ```
  # ls /var/share
  audit  cores  crash  mail
  ```

  Symbolic links are automatically created from `/var` to the `/var/share` components listed above for compatibility purposes. This file system generally requires no administration except to ensure that `/var` components do not fill the root file system. If an Oracle Solaris 11 system is updated to Oracle Solaris 11.1, it might take some time to migrate data from the original `/var` directory to the `/var/share` directory.
- In addition, all Oracle Solaris OS components must reside in the root pool, with the exception of the swap and dump devices.
- A default swap device and dump device are automatically created as ZFS volumes in the root pool when a system is installed. You cannot use the same volume for both swap and dump devices. In addition, you cannot use swap files in a ZFS root environment. For more information, see "Swap and Dump Device Configuration Changes" on page 53.

Mounting File System Changes

Review the following considerations when mounting file systems on systems running Oracle Solaris 11.

- Similar to Oracle Solaris 10 releases, a ZFS file system is mounted automatically when it is created. No need exists to edit the `/etc/vfstab` to mount local ZFS file systems.

- If you want to create and mount a local legacy UFS file system to be mounted at boot time, you will need to add an entry to the `/etc/vfstab` file as in previous Solaris releases.
- If you want to mount a remote file system at boot time, you will need to add an entry to the `/etc/vfstab` file and start the following service:

  ```
  # svcadm enable svc:/network/nfs/client:default
  ```

  Otherwise, the file system is not mounted at boot time.

Managing ZFS File System Changes

The following ZFS file system features, not available in the Oracle Solaris 10 release, are available in Oracle Solaris 11:

- ZFS file system encryption – You can encrypt a ZFS file system when it is created. For more information, see Chapter 9, "Managing Security."
- ZFS file system deduplication – For important information about determining whether your system environment can support ZFS data deduplication, see "ZFS Data Deduplication Requirements" on page 68.
- ZFS file system sharing syntax changes – Includes both NFS and SMB file system sharing changes. For more information, see"ZFS File System Sharing Changes" on page 65.
- ZFS man page change – The `zfs.1m` manual page has been revised so that core ZFS file system features remain in the `zfs.1m` page, but delegated administration, encryption, and share syntax and examples are covered in the following pages:
  - `zfs_allow`(1M)
  - `zfs_encrypt`(1M)
  - `zfs_share`(1M)

Displaying ZFS File System Information

After the system is installed, review your ZFS storage pool and ZFS file system information.

Display ZFS storage pool information by using the `zpool status` command.

Display ZFS file system information by using the `zfs list` command. For example:

For a description of the root pool components, see "Reviewing the Initial ZFS BE After an Installation" on page 80.

Resolving ZFS File System Space Reporting Issues

The `zpool list` and `zfs list` commands are better than the previous `df` and `du` commands for determining your available pool and file system space. With the legacy commands, you cannot

easily discern between pool and file system space, nor do the legacy commands account for space that is consumed by descendent file systems or snapshots.

For example, the following root pool (`rpool`) has 5.46 GB allocated and 68.5 GB free.

```
# zpool list rpool
NAME   SIZE  ALLOC   FREE  CAP  DEDUP  HEALTH  ALTROOT
rpool   74G  5.46G  68.5G   7%  1.00x  ONLINE  -
```

If you compare the pool space accounting with the file system space accounting by reviewing the USED columns of your individual file systems, you can see that the pool space is accounted for. For example:

```
# zfs list -r rpool
NAME                      USED  AVAIL  REFER  MOUNTPOINT
rpool                    5.41G  67.4G  74.5K  /rpool
rpool/ROOT               3.37G  67.4G    31K  legacy
rpool/ROOT/solaris       3.37G  67.4G  3.07G  /
rpool/ROOT/solaris/var    302M  67.4G   214M  /var
rpool/dump               1.01G  67.5G  1000M  -
rpool/export             97.5K  67.4G    32K  /rpool/export
rpool/export/home        65.5K  67.4G    32K  /rpool/export/home
rpool/export/home/admin  33.5K  67.4G  33.5K  /rpool/export/home/admin
rpool/swap               1.03G  67.5G  1.00G  -
```

Resolving ZFS Storage Pool Space Reporting Issues

The SIZE value that is reported by the `zpool list` command is generally the amount of physical disk space in the pool, but varies depending on the pool's redundancy level. See the examples below. The `zfs list` command lists the usable space that is available to file systems, which is disk space minus ZFS pool redundancy metadata overhead, if any.

- **Non-redundant storage pool** – Created with one 136-GB disk, the `zpool list` command reports SIZE and initial FREE values as 136 GB. The initial AVAIL space reported by the `zfs list` command is 134 GB, due to a small amount of pool metadata overhead. For example:

  ```
  # zpool create tank c0t6d0
  # zpool list tank
  NAME   SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
  tank   136G  95.5K   136G     0%  1.00x  ONLINE  -
  # zfs list tank
  NAME   USED  AVAIL  REFER  MOUNTPOINT
  tank    72K   134G    21K  /tank
  ```

- **Mirrored storage pool**– Created with two 136-GB disks, `zpool list` command reports SIZE as 136 GB and initial FREE value as 136 GB. This reporting is referred to as the *deflated* space value. The initial AVAIL space reported by the `zfs list` command is 134 GB, due to a small amount of pool metadata overhead. For example:

  ```
  # zpool create tank mirror c0t6d0 c0t7d0
  # zpool list tank
  NAME   SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
  tank   136G  95.5K   136G     0%  1.00x  ONLINE  -
  ```

```
# zfs list tank
NAME   USED  AVAIL  REFER  MOUNTPOINT
tank    72K   134G    21K  /tank
```

- **RAID-Z storage pool** – Created with three 136-GB disks, the `zpool list` commands reports SIZE as 408 GB and initial FREE value as 408 GB. This reporting is referred to as the *inflated* disk space value, which includes redundancy overhead, such as parity information. The initial AVAIL space reported by the `zfs list` command is 133 GB, due to the pool redundancy overhead. The following example creates a RAIDZ-2 pool.

```
# zpool create tank raidz2 c0t6d0 c0t7d0 c0t8d0
# zpool list tank
NAME   SIZE  ALLOC   FREE    CAP  DEDUP  HEALTH  ALTROOT
tank   408G   286K   408G     0%  1.00x  ONLINE  -
# zfs list tank
NAME   USED  AVAIL  REFER  MOUNTPOINT
tank  73.2K   133G  20.9K  /tank
```

Making ZFS File Systems Available

Making ZFS file systems available is similar to Oracle Solaris 10 releases in the following ways:

- A ZFS file system is mounted automatically when it is created and then remounted automatically when the system is booted.
- You do not have to modify the `/etc/vfstab` file to mount a ZFS file system, unless you create a legacy mount for ZFS file system. Mounting a ZFS file system automatically is recommended over using a legacy mount.
- You do not have to modify the `/etc/dfs/dfstab` file to share file systems. For more information about sharing ZFS file systems, see "ZFS File System Sharing Changes" on page 65.
- Similar to a UFS root, the swap device must have an entry in the `/etc/vfstab` file.
- File systems can be shared between Oracle Solaris 10 and Oracle Solaris 11 systems by using NFS sharing.
- File systems can be shared between Oracle Solaris 11 systems by using NFS or SMB sharing.
- ZFS storage pools can be exported from an Oracle Solaris 10 system and then imported on an Oracle Solaris 11 system.

ZFS File System Sharing Changes

In Oracle Solaris 10, you could set the `sharenfs` or `sharesmb` property to create and publish a ZFS file system share, or you could use the legacy `share` command.

In Oracle Solaris 11, you create a ZFS file system share and then publish the share as follows:

- Create an NFS or SMB share of a ZFS file system by using the `zfs set share` command.

```
# zfs create rpool/fs1
# zfs set share=name=fs1,path=/rpool/fs1,prot=nfs rpool/fs1
name=fs1,path=/rpool/fs1,prot=nfs
```

- Publish the NFS or SMB share by setting the `sharenfs` or `sharesmb` property to `on`.

```
# zfs set sharenfs=on rpool/fs1
# cat /etc/dfs/sharetab
/rpool/fs1      fs1     nfs     sec=sys,rw
```

The primary new sharing differences are as follows:

- Sharing a file system is a two-step process: creating a share by using the `zfs set share` command, then publishing the share by setting the `sharenfs` or `sharesmb` property.
- The `zfs set share` command replaces the `sharemgr` interface for sharing ZFS file systems.
- The `sharemgr` interface is no longer available. The legacy `share` command and the `sharenfs` property are still available. See the examples below.
- The `/etc/dfs/dfstab` file still exists but modifications are ignored. SMF manages ZFS or UFS share information so that file systems are shared automatically when the system is rebooted, similar to the way ZFS mount and share information is managed.
- If you unpublish a share, you can republish it by using the `share` command or by using the `share -a` command to republish all shares.
- Descendent file systems do not inherit share properties. If a descendent file system is created with an inherited `sharenfs` property set to on, then a share is created for the new descendent file system.

In Oracle Solaris 11.1, sharing ZFS file systems has improved with the following primary enhancements:

- The share syntax is simplified. You can share a file system by setting the new `share.nfs` or `share.smb` property.

```
# zfs set share.nfs=on tank/home
```

- Better inheritance of share properties to descendent file systems. In preceding example, where the `share.nfs` property is set on the `tank/home` file system, the `share.nfs` property value is inherited to any descendent file systems.

```
# zfs create tank/home/userA
# zfs create tank/home/userB
```

- You can also specify additional property values or modify existing property values on existing file system shares.

```
# zfs set share.nfs.nosuid=on tank/home/userA
```

These file sharing improvements are associated with pool version 34. For more information, see "Sharing and Unsharing ZFS File Systems" in *Oracle Solaris 11.1 Administration: ZFS File Systems*.

Legacy ZFS Sharing Syntax

Legacy sharing syntax is still supported without having to modify the `/etc/dfs/dfstab` file. Legacy shares are managed by an SMF service.

1. Use the `share` command to share a file system.

   For example, to share a ZFS file system:

   ```
   # share -F nfs /tank/zfsfs
   # cat /etc/dfs/sharetab
   /tank/zfsfs        -       nfs     rw
   ```

   The above syntax is identical to sharing a UFS file system:

   ```
   # share -F nfs /ufsfs
   # cat /etc/dfs/sharetab
   /ufsfs  -       nfs     rw
   /tank/zfsfs     -       nfs     rw
   ```

2. You can create a file system with the `sharenfs` property enabled, as in previous releases. The Oracle Solaris 11 behavior is that a default share is created for the file system.

   ```
   # zfs create -o sharenfs=on rpool/data
   # cat /etc/dfs/sharetab
   /rpool/data     rpool_data      nfs     sec=sys,rw
   ```

The above file system shares are published immediately.

ZFS Sharing Migration/Transition Issues

Review the share transition issues in this section.

- **Upgrading your system** – ZFS shares will be incorrect if you boot back to an older BE due to property changes in this release. Non-ZFS shares are unaffected. If you plan to boot back to an older BE, save a copy of the existing share configuration prior to the `pkg update` operation to be able to restore the share configuration on the ZFS datasets.
  - In the older BE, use the `sharemgr show -vp` command to list all shares and their configuration.
  - Use the `zfs get sharenfs` *filesystem* command and the `zfs sharesmb` *filesystem* commands to get the values of the sharing properties.
  - If you boot back to an older BE, reset the `sharenfs` and `sharesmb` properties to their original values.
- **Legacy unsharing behavior** – Using the `unshare -a` command or `unshareall` command unpublishes a share, but does not update the SMF shares repository. If you try to re-share the existing share, the shares repository is checked for conflicts, and an error is displayed.

ZFS Data Deduplication Requirements

In Oracle Solaris 11, you can use the deduplication (dedup) property to remove redundant data from your ZFS file systems. If a file system has the dedup property enabled, duplicate data blocks are removed synchronously. The result is that only unique data is stored, and common components are shared between files. For example:

```
# zfs set dedup=on tank/home
```

Do not enable the dedup property on file systems that reside on production systems until you perform the following steps to determine if your system can support data deduplication.

1. Determine if your data would benefit from deduplication space savings. If your data is not dedup-able, there is no point in enabling dedup. Running the following command is very memory intensive:

```
# zdb -S tank
Simulated DDT histogram:
bucket              allocated                       referenced
______   ______________________________   ______________________________
refcnt   blocks   LSIZE   PSIZE   DSIZE   blocks   LSIZE   PSIZE   DSIZE
------   ------   -----   -----   -----   ------   -----   -----   -----
     1    2.27M    239G    188G    194G    2.27M    239G    188G    194G
     2     327K   34.3G   27.8G   28.1G     698K   73.3G   59.2G   59.9G
     4    30.1K   2.91G   2.10G   2.11G     152K   14.9G   10.6G   10.6G
     8    7.73K    691M    529M    529M    74.5K   6.25G   4.79G   4.80G
    16      673   43.7M   25.8M   25.9M    13.1K    822M    492M    494M
    32      197   12.3M   7.02M   7.03M    7.66K    480M    269M    270M
    64       47   1.27M    626K    626K    3.86K    103M   51.2M   51.2M
   128       22    908K    250K    251K    3.71K    150M   40.3M   40.3M
   256        7    302K     48K   53.7K    2.27K   88.6M   17.3M   19.5M
   512        4    131K   7.50K   7.75K    2.74K    102M   5.62M   5.79M
    2K        1      2K      2K      2K    3.23K   6.47M   6.47M   6.47M
    8K        1    128K      5K      5K    13.9K   1.74G   69.5M   69.5M
 Total    2.63M    277G    218G    225G    3.22M    337G    263G    270G

dedup = 1.20, compress = 1.28, copies = 1.03, dedup * compress / copies = 1.50
```

   If the estimated dedup ratio is greater than 2, then you might see dedup space savings.

   In this example, the dedup ratio (dedup = 1.20) is less than 2, so enabling dedup is not recommended.

2. Make sure your system has enough memory to support dedup.
   - Each in-core dedup table entry is approximately 320 bytes.
   - Multiply the number of allocated blocks times 320. For example:

     ```
     in-core DDT size = 2.63M x 320 = 841.60M
     ```

3. Dedup performance is best when the deduplication table fits into memory. If the dedup table has to be written to disk, then performance will decrease. If you enable deduplication on your file systems without sufficient memory resources, system performance might

degrade during file system related operations. For example, removing a large dedup-enabled file system without sufficient memory resources might impact system performance. .

Considering ZFS Backup Features

- There are no `ufsdump` and `ufsrestore` command equivalents – You can use a combination of features to provide file system backup features.
- Create ZFS snapshots of important file systems and clone file systems that you can then modify as needed
- Send and receive ZFS snapshots to a remote system
- Save ZFS data with archive utilities such as `tar`, `cpio`, and `pax` or enterprise backup products.

Migrating File System Data to ZFS File Systems

Consider the following recommended data migration practices, if you are migrating data to systems running the Oracle Solaris 11 release.

Recommended Data Migration Practices

- Do not mix UFS directories and ZFS file systems in the same file system hierarchy because this model is confusing to administer and maintain.
- Do not mix NFS legacy shared ZFS file systems and ZFS NFS shared file systems because this model is difficult to maintain. Consider using only ZFS NFS shared file systems.
- Use the shadow migration feature to migrate existing UFS data over NFS to ZFS file systems.

Migrating Data With ZFS Shadow Migration

ZFS shadow migration is a tool you can use to migrate data from an existing file system to a new file system. A *shadow* file system is created that pulls data from the original source as necessary.

You can use the shadow migration feature to migrate file systems as follows:

- A local or remote ZFS file system to a target ZFS file system
- A local or remote UFS file system to a target ZFS file system

Shadow migration is a process that pulls the data to be migrated:

- Create an empty ZFS file system.
- Set the `shadow` property on an empty ZFS file system, which is the target (or shadow) file system, to point to the file system to be migrated. For example:

```
# zfs create -o shadow=nfs://system/export/home/ufsdata users/home/shadow2
```

- Data from the file system to be migrated is copied over to the shadow file system. For step-by-step instructions, see "Migrating ZFS File Systems" in *Oracle Solaris 11.1 Administration: ZFS File Systems*.

Review the following considerations when migrating file systems:

- The file system to be migrated must be set to read-only. If the file system is not set to read-only, in progress changes might not be migrated.
- The target file system must be completely empty.
- If the system is rebooted during a migration, the migration continues after the reboot.
- Access to directory content that is not completely migrated or access to file content that is not completely migrated is blocked until the entire content is migrated.
- If you want the UID, GID, and ACL information to be migrated to the shadow file system during an NFS migration, make sure that the name service information is accessible between the local and remote systems. You might consider copying a subset of the file system data to be migrated for a test to see that all the ACL information is migrated properly before completing a large migration of data over NFS.
- Migrating file system data over NFS can be slow, depending on your network bandwidth.
- Monitor file system data migration with the `shadowstat` command. See "Migrating ZFS File Systems" in *Oracle Solaris 11.1 Administration: ZFS File Systems*.

Migrating UFS Data to a ZFS File System (`ufsdump` and `ufsrestore`)

You can also use `ufsrestore` to restore a previous `ufsdump` dump. For example:

```
# mount -F nfs rsystem:/export/ufsdata /tank/legacyufs
# ls /tank/legacyufs
ufsdump-a
# zfs create tank/newzfs
# cd /tank/newzfs
# ufsrestore rvf /tank/legacyufs/ufsdump-a
```

If the original UFS file system data includes POSIX-draft ACLs, they are translated into to NFSv4 ACLs. See the Chapter 7, "Using ACLs and Attributes to Protect Oracle Solaris ZFS Files," in *Oracle Solaris 11.1 Administration: ZFS File Systems*.

◆ ◆ ◆ CHAPTER 6

Managing Software and Boot Environments

This chapter provides information about how to manage software and boot environments (BEs) in an Oracle Solaris 11 release.

The following topics are covered:

Oracle Solaris 11 Package Changes

The Image Packaging System (IPS) is a framework that provides the capability for software lifecycle management, which includes installation, upgrade, and the removal of packages. IPS uses packaging mechanisms that are significantly different than the legacy SVR4 packaging mechanism that is used in Oracle Solaris 10. An IPS package is a collection of directories, files, links, drivers, dependencies, groups, users, and license information in a defined format. This collection represents the installable objects of a package. Packages have attributes, such as package name and description. IPS packages are stored in IPS package repositories that are populated by IPS publishers. See Chapter 1, "Introduction to the Image Packaging System," in *Adding and Updating Oracle Solaris 11.1 Software Packages*.

The following IPS components, as well as the utility for managing boot environments, are described in this chapter:

- **IPS command-line utilities** – IPS includes a suite of `pkg` commands that enable you to list, search, install, update, and remove software packages. See `pkg`(1). IPS commands also enable you to manage package publishers and copy or create package repositories. See "Updating the Software on Your Oracle Solaris 11 System" on page 77.
- **IPS publishers and repositories** – A *publisher* identifies a person or an organization that provides one or more packages. A *repository* is a location from which you can install packages. See `http://pkg.oracle.com/solaris/release/`.

  If you have a system with zones that does not have direct access to a public IPS repository, see "Proxy Configuration on a System That Has Installed Zones" in *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.
- **Boot environment management** – An image is a location where IPS packages are installed and where other IPS operations are performed. Boot environments, also called a BEs, are bootable instances of an image. The `beadm` utility is used to create and manage boot environments, along with any other software packages that are installed in that image. Multiple BEs can be maintained on a single system, and each BE can have a different software version installed. A new BE can also be created automatically, as a result of a package operation. See "Tools for Managing Boot Environments" on page 80.

Oracle Solaris 10 SVR4 and IPS Package Comparison

Review the following information about software packaging in Oracle Solaris 11:

- The `SUNW` prefix for package names is no longer used. With the introduction of IPS, all software packages are renamed. A set of mappings has been added to the former SVR4 package database for compatibility. The mappings ensure package dependencies are met for administrators who want to install a legacy SVR4 package.
- Certain SVR4 package commands, such as `pkgadd`, are retained for administering legacy SVR4 packages, but the primary package installation and update interface is now the `pkg`(1) set of commands. If you previously used the `pkgadd` command to install a particular package, you can check whether that package is available as an IPS package. The IPS package name will most likely be different.

  Locate an SVR4 package as follows:

```
$ pkg info -g http://pkg.oracle.com/solaris/release/ SUNWcsl
          Name: SUNWcsl
       Summary:
         State: Not installed (Renamed)
    Renamed to: system/library@0.5.11-0.133
                consolidation/osnet/osnet-incorporation
     Publisher: solaris
       Version: 0.5.11
 Build Release: 5.11
```

```
        Branch: 0.133
Packaging Date: Wed Oct 27 18:35:58 2010
          Size: 0.00 B
          FMRI: pkg://solaris/SUNWcsl@0.5.11,5.11-0.133:20101027T183558Z
```

This output shows that the SVR4 `SUNWcsl` package has been renamed (Rename) is now the IPS `system/library` package. If the package you want is not installed, use the `pkg install` command to install the package:

```
$ pkg install system/library
```

- If an SVR4 package is available as an IPS package, install the IPS package and not the SVR4 package. Installing the IPS package provides many advantages, including that only versions compatible with the rest of the image can be installed and dependencies are automatically checked and updated. See *Adding and Updating Oracle Solaris 11.1 Software Packages*.
- Certain SVR4 package commands, for example, `patchadd`, are no longer available. Instead, use the IPS `pkg update` command. When you use this command, any package dependencies are automatically resolved.
- IPS packages have FMRIs, similar to SMF service names. Package names are also hierarchical instead of abbreviated. As described previously, the core system library package in Oracle Solaris 10 is `SUNWcsl` but the IPS name is `system/library`. The FMRI format of `system/library` is similar to:

  ```
  pkg://solaris/system/library@0.5.11,5.11-0.175.1.0.0.24.2:20120919T185104Z
  ```

  See "Fault Management Resource Identifiers" in *Adding and Updating Oracle Solaris 11.1 Software Packages*.

Note – Due to organizational restructuring of the files that are delivered with each package, there is not an exact one-to-one mapping of Oracle Solaris 10 package names to Oracle Solaris 11 package names.

- Oracle Solaris packages are not split into development, documentation, and runtime components. For example, in Oracle Solaris 10, the standard X11 library (`libX11`) runtime is in the `SUNWxwplt` package, while the headers for the same package are in `SUNWxwinc`, and the documentation is in the `SUNWxwpmn` package. In Oracle Solaris 11, all of these components are located in the in the `pkg:/x11/library/libx11` package. If you want to minimize the system, you can choose to exclude certain components by using the `pkg facet` command.

  Remove the man pages as follows:

  ```
  # pkg change-facet facet.doc.man=false
  ```

  Remove the header files as follows:

  ```
  # pkg change-facet facet.devel=false
  ```

Note – These are global settings that remove all of the man pages for all packages and all of the header files for all packages.

See "Controlling Installation of Optional Components" in *Adding and Updating Oracle Solaris 11.1 Software Packages*.

- SVR4 packaging and patch tools are still supported in Oracle Solaris 10 Containers. These Oracle Solaris 10 branded, non-global zones run on Oracle Solaris 11, utilizing the zones and branded zones technology. See "Oracle Solaris 11 Zone Features" on page 148.
- For SVR4 package to IPS package conversion information, see "Converting SVR4 Packages To IPS Packages" in *Packaging and Delivering Software With the Image Packaging System in Oracle Solaris 11.1*.

The following table compares SVR4 package and patch commands with IPS package commands.

TABLE 6–1 SVR4 and IPS Package Command Equivalents

SVR4 Package Command	IPS Package Command Equivalent
`pkgadd`	`pkg install`
`patchadd`	`pkg update`
`pkgrm`	`pkg uninstall`
`pkgadm addcert`, `pkgadm removecert`	`pkg set-publisher -k`, `-c`, `--approve-ca-cert`, `--revoke-ca-cert`, `unset-ca-cert`
`pkginfo`, `pkgchk -l`	`pkg info`, `pkg list`, `pkg contents`, `pkg search`
`pkgchk`	`pkg verify`, `pkg fix`, `pkg revert`

IPS Installation Package Groups

Oracle Solaris 10 installation methods provide software package clusters that install a group of packages based on the system's purpose, such as minimal network, desktop, developer, and all for servers.

Oracle Solaris 11 provides three group packages that install different sets of packages appropriate for a larger server, a smaller server or non-global zone, or a graphical desktop environment.

The following table describes the group packages that are installed on the system, depending upon the default installation method that is used.

TABLE 6–2 Oracle Solaris 11 Group Packages Installed by Default

Group Name/Summary	Description	Default Installation Method	Distribution Constructor ISO Boot Image
`group/system/solaris-desktop` Oracle Solaris Desktop	Provides the GNOME desktop environment and other GUI tools such as web browsers and mail. It also includes drivers for graphics and audio devices.	Live Media	Live Media
`group/system/solaris-large-server` Oracle Solaris Large Server	Provides common network services for an enterprise server. This group package also contains hardware drivers that are required for servers, such as InfiniBand drivers.	Text installer from media and default Automated Installer	Text installer
`group/system/solaris-small-server` Oracle Solaris Small Server	Provides a command-line environment and is also an smaller set of packages to be installed on a server.	Non-global Zones	

Display package group information as follows:

```
# pkg info -r *group*
```

Display the contents of these package groups:

```
# pkg contents -o fmri -r -t depend pkg-grouping
```

Determine which package group is currently installed on your system:

```
# pkg list group/system/\*
```

IPS also includes other meta and group packages that can be installed on your system to provide a trusted desktop or multi-user desktop.

If you want to install most packages, similar to installing the Solaris 10 `SUNWCall` package cluster, consider installing the `group/system/solaris-large-server` package group. See "Listing All Installable Packages in a Group Package" in *Adding and Updating Oracle Solaris 11.1 Software Packages*.

Displaying Information About Software Packages

To display information about software packages, refer to the following examples. No special privileges are required to display information about packages.

List the packages that are currently installed on your system:

```
$ pkg list | more
```

Determine whether a specific package is installed in the current image and whether an update is available.

```
$ pkg list amp
pkg list: no packages matching 'amp' installed
```

Display more information about a package that is not installed. Use the `-r` option to query the package repository, as follows:

```
$ pkg info -r amp
          Name: amp
       Summary:
         State: Not installed (Renamed)
    Renamed to: web/amp@0.5.11-0.133
                consolidation/sfw/sfw-incorporation
     Publisher: solaris
       Version: 0.5.11
 Build Release: 5.11
        Branch: 0.133
Packaging Date: Wed Oct 27 18:31:05 2010
          Size: 0.00 B
          FMRI: pkg://solaris/amp@0.5.11,5.11-0.133:20101027T183105Z

          Name: group/feature/amp
       Summary: AMP (Apache, MySQL, PHP) Deployment Kit for Oracle Solaris
   Description: Provides a set of components for deployment of an AMP (Apache,
                MySQL, PHP) stack on Oracle Solaris
      Category: Meta Packages/Group Packages (org.opensolaris.category.2008)
                Web Services/Application and Web Servers (org.opensolaris.category.2008)
         State: Not installed
     Publisher: solaris
       Version: 0.5.11
 Build Release: 5.11
        Branch: 0.175.1.0.0.24.0
Packaging Date: Tue Sep 04 18:03:28 2012
          Size: 5.46 kB
          FMRI: pkg://solaris/group/feature/amp@0.5.11,5.11-0.175.1.0.0.24.0:20120904T180328Z

          Name: web/amp
       Summary:
         State: Not installed (Renamed)
    Renamed to: group/feature/amp@0.5.11-0.174.0.0.0.0.0
                consolidation/ips/ips-incorporation
     Publisher: solaris
       Version: 0.5.11
 Build Release: 5.11
```

```
        Branch: 0.174.0.0.0.0.0
Packaging Date: Wed Sep 21 19:15:02 2011
          Size: 5.45 kB
          FMRI: pkg://solaris/web/amp@0.5.11,5.11-0.174.0.0.0.0.0:20110921T191502Z
```

If you know the name of the tool that you want to install, but not the name of the package, use the `search` subcommand in one of the following ways:

```
$ pkg search /usr/bin/emacs
INDEX      ACTION VALUE         PACKAGE
path       file   usr/bin/emacs pkg:/editor/gnu-emacs@23.4-0.175.1.0.0.24.0
$ pkg search file::emacs
INDEX      ACTION VALUE               PACKAGE
basename   file   usr/share/info/emacs pkg:/editor/gnu-emacs@23.4-0.175.1.0.0.24.0
basename   file   usr/bin/emacs        pkg:/editor/gnu-emacs@23.4-0.175.1.0.0.24.0
```

Updating the Software on Your Oracle Solaris 11 System

With IPS, you can update all of the packages on your system that have available updates, or you can update individual packages that are not constrained by the system. If a package is constrained, an appropriate message indicating why it has been constrained is provided. Package constraints generally represent a dependency or a versioning issue. For most package update operations, either a clone BE is created or a backup BE is created before applying the software updates to the clone BE so that you can boot back to the previous BE. Some `pkg update` operations, such as updating a non-global zone or updating a specific package, might not generate a clone BE or a backup BE.

The following options are available:

- **Adding software packages after an installation** – The Live Media contains a set of software that is appropriate for a desktop or laptop computer. The text installation media contains a smaller set of software that is more appropriate for a general-purpose server system. The text installer does not install the GNOME desktop. To add packages, including the Oracle Solaris Desktop (GNOME 2.30), after a text installation, see "Adding Software After Text Installation" in *Installing Oracle Solaris 11.1 Systems*.
- **Updating all of the packages on your installed system** – To update all of the packages on your system that have available updates, use the `pkg update` command, as follows:

  ```
  # pkg update
  ```

  Running this command updates packages that you might not otherwise consider updating, for example, kernel components and other low-level system packages.

  Depending on your package repository or publisher status, your system could be updated automatically from Solaris 11 to Solaris 11.1. If you want to update your system image, but you do not want to update to another release, see "Updating an Image" in *Adding and Updating Oracle Solaris 11.1 Software Packages*.

See an example of using this command to update a BE in "Managing Boot Environments" on page 80.

Display those packages on a system that require an update without actually installing the packages.

```
# pkg update -nv --accept
```

- **Adding or updating individual packages** – To add individual software packages, use the `pkg install` command. Any dependent packages are also updated at the same time.

  Install an individual package as follows:

  ```
  # pkg install communication/im/pidgin
  ```

  Update an individual package as follows:

  ```
  # pkg update system/management/ocm
  ```

- **Install package updates that deliver fixes** – A `pkg update` operation might include bug fixes, so the operation is similar to applying a specific patch or patches in previous Oracle Solaris releases.

Installing Maintenance Updates on an Oracle Solaris 11 System

Oracle customers with an active Oracle support plan have access to the `support` package repository so that you can routinely update your Oracle Solaris 11 systems. Updates to the `support` repository are called Support Repository Updates (SRUs) and occur on a regular basis. See "How to Configure the Oracle Solaris `support` Repository" on page 79.

If you need to access an IPS repository on a system that has Oracle Solaris Zones installed by using `https_proxy` and `http_proxy`, see "Proxy Configuration on a System That Has Installed Zones" in *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.

- **SRUs** – Updates from the Oracle Solaris 11 `support` repository are available as support repository updates (SRUs). SRUs take the place of maintenance updates or patch bundles that are available for Oracle Solaris 10 releases.
- **Future Oracle Solaris 11 releases** – Future Oracle Solaris 11 releases are made available in the `support` repository or a `release` repository that provides the currently available OS.

The following summary provides information about selecting the update method that works best for your environment. For more information about the best way to update your system images, see "Updating an Image" in *Adding and Updating Oracle Solaris 11.1 Software Packages*.

- **Desktop systems or laptop computers** – In a desktop environment, you can identify what updates are available by using the following command:

  ```
  # pkg update -nw --accept
  ```

- **Development systems** – You can use the `pkg update` operation to apply a specific fix or apply an SRU to these systems to assess the impact of your applications that are currently under development. A new BE is created when an SRU is applied, and you can fall back to the original BE, if necessary.
- **Production systems** – In a large enterprise environment, SRUs can be applied to a non-production system to assess the impact of the OS changes to the currently running production environment. If the non-production system remains stable after the SRU installation and assessment is complete, the SRU can be applied to a new BE on the production system, and you can fall back to the original BE, if necessary.

How to Configure the Oracle Solaris support Repository

Use the following steps to configure the support repository, if you want to apply support updates.

1. **Log into the following site.**

   `http://pkg-register.oracle.com/`

2. **Download the SSL key and certificate for the Oracle Solaris 11 release.**

   Consider creating a directory inside `/var/pkg` to store the key and certificate.

   ```
   # mkdir -m 0755 -p /var/pkg/ssl
   # cp -i Oracle_Solaris_11_Support.key.pem /var/pkg/ssl
   # cp -i Oracle_Solaris_11_Support.certificate.pem /var/pkg/ssl
   ```

3. **Copy the key and certificate from the directory that you downloaded the key and certificate into this directory.**

   The key files are kept by *reference*, so if the files become inaccessible to the packaging system, you will encounter errors.

4. **Set the publisher to the `support` repository.**

   ```
   # pkg set-publisher \
               -k /var/pkg/ssl/Oracle_Solaris_11_Support.key.pem \
               -c /var/pkg/ssl/Oracle_Solaris_11_Support.certificate.pem \
               -O https://pkg.oracle.com/solaris/support solaris
   ```

5. **Install the updated packages from the `support` repository, if desired.**

   ```
   # pkg update
   ```

   As mentioned previously, this operation updates the packages on the system with the latest package versions either by creating a new BE or by creating a backup BE.

Managing Boot Environments

Previously, you could perform a live upgrade or use the `patchadd` command to update your BE. In Oracle Solaris 11, the `pkg update` command is used to update a BE, or you can use the `beadm` command set to create, display, and remove BEs.

Tools for Managing Boot Environments

In Oracle Solaris 11, the `beadm` utility replaces the `lu` set of commands for managing ZFS BEs. In most cases, the `pkg update` command creates and updates a clone BE, so you boot back to the previous BE, if necessary.

TABLE 6–3 Comparison of Boot Environment Command Syntax

Oracle Solaris 10 Syntax	Oracle Solaris 11 Syntax	Description
`lucreate -n` *newBE*	`beadm create` *newBE*	Create a new BE
`lustatus`	`beadm list`	Display BE information
`luactivate` *newBE*	`beadm activate` *newBE*	Activate a BE
`ludelete` *BE*	`beadm destroy` *BE*	Destroy an inactive BE
`luupgrade` or `patchadd`	`pkg update`	Upgrade or update a BE

See *Creating and Administering Oracle Solaris 11.1 Boot Environments* and `beadm`(1M).

The system performs the following actions in most cases:

1. Creates a clone of the current BE that is a bootable image.
2. Updates the packages in the clone BE, but does not update any packages in the current BE.
3. Sets the new BE as the default boot choice the next time the system is booted. The current BE remains as an alternate boot choice.

Use the `beadm` command to create, rename, mount, unmount, activate, or destroy BEs.

Reviewing the Initial ZFS BE After an Installation

After a system is installed, the following root pool file systems and components are available:

```
# zfs list -r rpool
NAME                      USED  AVAIL  REFER  MOUNTPOINT
rpool                    13.0G   121G  4.58M  /rpool
rpool/ROOT               6.81G   121G    31K  legacy
rpool/ROOT/solaris       6.81G   121G  4.07G  /
rpool/ROOT/solaris/var    364M   121G   207M  /var
```

```
rpool/VARSHARE              50K   121G    50K  /var/share
rpool/dump                4.13G   121G  4.00G  -
rpool/export                63K   121G    32K  /export
rpool/export/home           31K   121G    31K  /export/home
rpool/swap                2.06G   121G  2.00G  -
```

- `rpool` – Is the root pool and a mount point containing boot-related components.
- `rpool/ROOT` – Is a special component that is not accessible and requires no administration.
- `rpool/ROOT/solaris` – Is the actual root ZFS BE, which is accessible from the `/` directory.
- `rpool/ROOT/solaris/var` – Is the separate `var` file system.
- `rpool/VARSHARE` – Is a special component for the `/var/shared` file system that is new in Oracle Solaris 11.1 For more information, see "Root File System Requirements and Changes" on page 62.
- `rpool/dump` – Is the dump volume.
- `rpool/swap` – Is the swap volume.
- `rpool/export/home` – Is a default mount point for home directories. In an enterprise environment with many users, you might consider moving `export/home` to another pool.

How to Update Your ZFS Boot Environment

To update a ZFS boot environment, use the `pkg update` command. A clone or backup BE is created and automatically activated in most cases. The `pkg update` command displays whether a backup BE or a new BE is created.

Caution – If you update your BE and also upgrade your root pool version, if one is available for the recent update, you will not be able to boot back to a previous BE, if previous BE is in a lower pool version. Make sure you have tested all the features and you are satisfied with the current release update before upgrading your pool version.

For information about upgrading your pool version, see "Upgrading ZFS Storage Pools" in *Oracle Solaris 11.1 Administration: ZFS File Systems*.

1 Display your existing BE information.

```
# beadm list
BE        Active Mountpoint Space Policy Created
--        ------ ---------- ----- ------ -------
solaris   NR     /          9.71G static 2013-01-04 12:35
```

In the above output, `NR` means the BE is active now and will be the active BE on reboot.

2 Update your BE.

```
# pkg update
                 Packages to remove:   117
               Packages to install:   186
```

```
              Packages to update:   315
          Create boot environment:   Yes
DOWNLOAD                                  PKGS       FILES    XFER (MB)
Completed                                618/618 29855/29855  600.7/600.7
.
.
.
```

If your existing BE name is `solaris`, a new BE, `solaris-1`, is created and automatically activated after the `pkg update` operation is complete.

3 Reboot the system to complete the BE activation. Then, confirm your BE status.

```
# init 6
.
.
.
# beadm list
BE        Active Mountpoint Space  Policy Created
--        ------ ---------- -----  ------ -------
solaris   -      -          18.19M static 2013-01-04 12:35
solaris-1 NR     /          9.82G  static 2013-01-31 13:03
```

4 If an error occurs when booting the new BE, activate and boot to the previous BE.

```
# beadm activate solaris
# init 6
```

If the activated BE does not boot, see "How to Boot From a Backup BE for Recovery Purposes" on page 124.

◆ ◆ ◆ CHAPTER 7

Managing Network Configuration

Network configuration in Oracle Solaris 11 works differently than in Oracle Solaris 10. This chapter provides basic information about how to configure your network in an Oracle Solaris 11 release.

The following topics are covered:

Network Configuration Feature Changes

The following features are new or have changed in Oracle Solaris 11.1:

- **One SMF network configuration repository for both datalink and IP configuration** – Oracle Solaris 11 uses one SMF network configuration repository for both datalink and IP configuration. As a result, the commands that are used to manage network configuration have also changed.
- **Networking command changes** – You can now use the `dladm` and `ipadm` commands to manage a reactive network configuration profile (NCP). To use fixed networking commands on a reactive NCP, it *must* be currently active. Otherwise, use the `netcfg` command to modify the NCP. See "Network Configuration Commands" on page 88.

Note – When the `DefaultFixed` NCP is active, you must use fixed networking commands to manage network configuration.

- `DefaultFixed` **Location** – In addition to the `Automatic` and `NoNet` system-defined Locations, the `DefaultFixed` Location profile has been added. This Location tracks changes that are made to naming services, etc., while the profile is active. Whenever the `DefaultFixed` NCP is active, the `DefaultFixed` Location is also active.
- **Link aggregations that span multiple switches** – Link aggregation support now includes a solution that enables the spanning of aggregations across multiple switches, independent of the switch vendor. See Chapter 2, "Using Link Aggregations," in *Managing Oracle Solaris 11.1 Network Performance*.
- **Multihoming policy changes** – The *multihoming model* controls the system policy for accepting and transmitting IP packets when multiple IP interfaces are simultaneously enabled. For example, if a system is configured with more than one IP interface, there might be multiple equivalent routes to a given destination. Similarly, a packet destined for an IP address that is hosted on another IP interface might arrive on a different IP interface. The system's behavior in these situations is determined by the selected *multihoming policy*. Oracle Solaris 11 supports three multihoming properties. The following properties are equivalent to the `ndd` multihoming policies used in Oracle Solaris 10:

`strong`	Equivalent to the `ip_strict_dst_multihoming = 1` setting through `ndd`, with an additional requirement that packets originated from the host are only sent out on interfaces where the IP source address of the outgoing packet is an address that is configured on the outgoing interface.
`weak`	Equivalent to the `ip_strict_dst_multihoming = 0` setting through `ndd`.
`src-priority`	Equivalent to the weak end-system model in receive behavior, for example, the packet is accepted on any interface, as long as the IP destination of the packet is configured on one of the host's interfaces.

  See `ipadm`(1M) for more details.
- **Advanced networking features**
  - **Edge virtual bridging (EVB)** – EVB is a technology that enables a host to exchange virtual link information with an external switch. EVB features enable you to advertise more information about virtual link configurations on the network than just the bandwidth share or priority definitions for physical links that data center bridging (DCB) features provide. See Chapter 9, "Edge Virtual Bridging in Oracle Solaris," in *Managing Oracle Solaris 11.1 Network Performance*.

- **Exclusive IP zones by default** – Exclusive IP zones enable you to assign a separate IP stack, per zone. Each zone has the flexibility to configure IP within that stack, completely separate of other zones. See Part II, "Oracle Solaris Zones," in *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.
- **VNIC migration** – Associations between Physical NICs (PNICs) and VNICs can now be migrated without disrupting network connectivity. Use the `dladm modify-vnic` command to migrate one or more VNICs from one underlying datalink to another underlying datalink, without having to delete and reconfigure the VNICs. The underlying link can be a physical link, a link aggregation, or an etherstub. For example:

  ```
  # dladm modify-vnic -l net1 -L ether0
  ```

  `-l` Refers to the destination datalink to which the VNICs are migrated.

  `-L` Refers to the original datalink over which the VNICs are configured. The `-L` option is restricted to global modification only.

  See "Migrating VNICs" in *Using Virtual Networks in Oracle Solaris 11.1*.

How the Network Is Configured in Oracle Solaris

Oracle Solaris 11 uses profile-based network configuration, which is comprised of two network configuration modes: fixed (manual) and reactive (automatic). The way you manage network configuration depends on the configuration mode that you are using and the profiles that are currently active on the system. After an installation, two system-defined network configuration profiles (NCPs) are present on the system: `DefaultFixed` and `Automatic`. Three system-defined Location profiles are present on the system after an installation: `Automatic`, `NoNet`, and `DefaultFixed` (new in Oracle Solaris 11.1). Additional reactive profiles can be created after an installation.

Both the text and AI installation methods default to fixed network configuration. For fixed network configuration, the `dladm` and `ipadm` commands are used. If the `Automatic` NCP or another reactive NCP is active after the installation, the `netcfg` and `netadm` commands are used to manage network configuration. Starting with Oracle Solaris 11.1, you can also use the `dladm` and `ipadm` commands to manage a reactive NCP, but the NCP must be currently active on the system.

Note the following additional information about profile-based network configuration:

- **Profile types and network configuration** – The two main profile types are the network configuration profile (NCP) and the Location profile. An NCP specifies the configuration of network datalinks and IP interfaces and addresses. The Location profile manages system-wide network configuration, for example naming services and IPfilter settings. Exactly one NCP and one Location profile *must* be active on the system at all times. If the `DefaultFixed` NCP is active, the system-defined `DefaultFixed` Location profile is also active. If any other reactive NCP is active, the Location that is activated is determined by rules and criteria that are specified in each of the reactive Locations. For information about other network profile types, see "Network Profiles and Types" in *Connecting Systems Using Reactive Network Configuration in Oracle Solaris 11.1*.
- **Using the `Automatic` NCP** – The `Automatic` NCP is a system-defined profile that manages datalink and IP configuration, based on the current network environment. This NCP is automatically updated whenever your network environment changes, for example, when network devices are added or removed from the system. You cannot delete the `Automatic` NCP. You can modify this NCP by using the `dladm` and `ipadm` commands, but any modifications should be made *with care*.

  Rather than directly modifying the `Automatic` NCP, the preferred method is to clone this NCP, and then apply your changes to the copy. Because the system does not change the configuration for any user-defined NCPs, including copies of the `Automatic` NCP, any changes that you make are preserved. See Example 7–8.
- **How system-defined Locations work** – These profiles include the `Automatic`, `NoNet`, and `DefaultFixed` Locations. The `DefaultFixed` Location (new in Oracle Solaris 11.1) tracks changes that are made to naming services, etc. For example, the system updates the `DefaultFixed` Location to preserve changes that are made to any relevant SMF services (while the Location is active). When the `DefaultFixed` NCP is active on the system, so is the `DefaultFixed` Location. System-defined Location profiles can be modified by using the `netcfg` command, but only after these profiles have been activated on a system for the first time. For more information, see Chapter 1, "Reactive Network Configuration (Overview)," in *Connecting Systems Using Reactive Network Configuration in Oracle Solaris 11.1*.

How the Network Is Configured During an Installation

During an installation, the network is configured, as follows:

- For a GUI installation, the `Automatic` NCP is activated, and the network is automatically configured, based on current network conditions.
- For a text installation, you must choose Automatic, Manual, or None.
  - If you choose Automatic, the `Automatic` NCP is activated, and the network is automatically configured upon reboot.

- If you choose Manual, the `DefaultFixed` NCP is activated, and you are presented with a series of installation screens that enable you to manually configure network settings.
- If you choose None, the `DefaultFixed` NCP is activated, but you do not provide network parameters during the installation. Thus, after a reboot, no network interface is plumbed or configured. Only the loopback IPv4 and IPv6 interfaces (`lo0`) are activated. You can create persistet network configuration by using `dladm` and `ipadm` after the installation. See "Managing Network Configuration in Fixed Mode" on page 89.

- For an installation with AI, the network is configured according to the profile that you set up before the installation. If you did not specify any network settings prior to installing Oracle Solaris, the interactive `sysconfig` tool runs during the installation, enabling you to set network parameters for the system. See "Installing Oracle Solaris by Using AI" on page 38.

Note – Several aspects of network configuration have changed in Oracle Solaris 11, including where certain network configuration information is stored. For example, a system's default route is no longer stored in the `/etc/defaultrouter` file because this file is deprecated in Oracle Solaris 11. Do not check this file after an installation to determine a system's default route. Instead, use the `route -p show` command or the `netstat -nr` command. For more information, see "Creating Persistent Routes (Fixed and Reactive)" on page 103.

EXAMPLE 7–1 Verifying the Active NCP on a System

After an installation, use the `netadm list` command to determine which NCP is active (online). In the following example, the output of the `netadm list` command shows that the Automatic NCP is currently active:

```
$ netadm list
TYPE        PROFILE        STATE
ncp         Automatic      online
ncu:phys    net0           online
ncu:ip      net0           online
ncu:phys    net1           offline
ncu:ip      net1           offline
ncu:phys    net2           offline
ncu:ip      net2           offline
ncu:phys    net3           offline
ncu:ip      net3           offline
loc         Automatic      offline
loc         NoNet          offline
loc         myloc          online
loc         myncp          disabled
```

In the previous output, a user-defined Location named `myloc` is also online. This Location defines system-wide network settings for this particular configuration. When using reactive network configuration, exactly one NCP (either the Automatic NCP or another reactive NCP) and one Location must be active on the system at all times.

EXAMPLE 7–1 Verifying the Active NCP on a System *(Continued)*

The output of the `netadm list` command in the following example shows that the `DefaultFixed` NCP is active, which means you must configure the network manually by using the `dladm` and `ipadm` commands. Note that whenever the `DefaultFixed` NCP is online, the `DefaultFixed` Location is also online:

```
# netadm list
TYPE        PROFILE        STATE
ncp         Automatic      disabled
ncp         DefaultFixed   online
loc         Automatic      offline
loc         NoNet          offline
loc         DefaultFixed   online
```

EXAMPLE 7–2 Switching the Default NCP

Switching network configuration modes requires you to enable the appropriate NCP for that configuration mode. The following example shows how to switch from reactive mode to fixed mode by enabling the `DefaultFixed` NCP:

```
$ netadm enable -p ncp DefaultFixed
```

Switch to the `Automatic` NCP, as follows:

```
$ netadm enable -p ncp Automatic
```

The process of switching network configuration modes can take a few minutes. During this time, any messages about various network services that are displayed can be safely ignored.

Network Configuration Commands

The following commands are used to manage network configuration:

- `dladm` – Configures datalinks. The command creates persistent configuration that is applied to the currently active profile on the system (fixed and reactive).
- `ipadm` – Configures IP interfaces and addresses. The command creates persistent configuration that is applied to the currently active profile on the system (fixed and reactive).
- `netcfg` – Administers reactive network configuration on the system for both active and non-active NCPs.
- `netadm` – Displays information about the system's network profiles; enables and disables NCPs and Locations.

Starting with Oracle Solaris 11.1, you can use fixed networking commands to manage reactive NCPs, provided that the NCP is currently active. This change applies generally to all fixed networking commands. You can still use the `netcfg` and `netadm` commands to manage any reactive NCP (active and non-active).

Note the following additional information about using networking commands in this release:

- The `dladm` and `ipadm` commands are used to configure datalinks and IP interfaces, respectively, for the currently active NCP (fixed and reactive).
- The `netcfg` command is used to configure other properties of reactive NCPs (active and non-active).
- You cannot use the `netcfg` and `netadm` commands to administer the `DefaultFixed` NCP, which is the system's only fixed NCP. However, you can view the properties and the status (online or offline) of this NCP by using these commands.
- Configure and view properties that refer to the default route, as follows:
  - For both fixed and reactive NCPs, you can use the `route -p add` command to create a static route (default or otherwise) that is applied to the currently active NCP. This command directly sets the default route in the system's routing table. See "Creating Persistent Routes (Fixed and Reactive)" on page 103.
  - For reactive NCPs *only*, you can use the `netcfg` command to create a single, per-interface default route. To view the default route for the NCP, use the `netcfg` command.
  - To view the currently active routes on a system for any NCP, use the `netstat -rn` command.

See "Network Configuration Tools" in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*.

Managing Network Configuration in Fixed Mode

If you are managing the network in fixed mode, the active NCP is `DefaultFixed`. This profile is system-defined and is the only fixed profile on the system. Oracle Solaris does not support the use of multiple fixed profiles. The properties of the `DefaultFixed` NCP reflect the persistent configuration for the system while this NCP is active.

Note – Starting with Oracle Solaris 11.1, you can use fixed networking commands to configure reactive profiles that are currently active.

Using fixed network configuration enables you to have full control of all network configuration information. If the `DefaultFixed` NCP is active, you make explicit changes to network

configuration by using the `dladm` and `ipadm` commands. Conversely, with reactive network configuration, the network is automatically configured, as a direct result of changes in the current network conditions. If you are using reactive networking, the `netcfg` command is used to create and manage reactive profiles that specify network configuration parameters. See "Managing Network Configuration in Reactive Mode" on page 97.

When configuring the network in fixed mode, note the following additional information:

- Persistent network configuration is now managed through SMF, not by editing the following files:
  - `/etc/defaultdomain`
  - `/etc/dhcp.*`
  - `/etc/hostname.*`
  - `/etc/hostname.ip*.tun*`
  - `/etc/nodename`
  - `/etc/nsswitch.conf`

  Note – The `/etc/nsswitch.conf` file is still referenced in this release, but you do not directly edit the file to make configuration changes. See "Configuring Naming Services in Fixed Mode" on page 93.

  For more information about setting a system's host name, see "System Configuration Changes and Migration of System Configuration to SMF" on page 113.
- During an installation, the system undergoes a one time upgrade to convert any existing `/etc` network configuration files to their respective `ipadm` and `dladm` configurations. The `dladm` command is used to configure datalinks. The `ipadm` command is used to configure IP interfaces. The `ipadm` command provides almost equivalent functionality to the `ifconfig` command. The `ipadm` command also replaces the `ndd` command. To compare `ifconfig` and `ndd` command options with the `ipadm` command, see Appendix A, "Comparison Map: ifconfig and ipadm Commands," in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1* and Appendix B, "Comparison Map: ndd and ipadm Commands," in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*.
- Network virtualization features are also configured and managed by using the `dladm` and `ipadm` commands. Objects that are in the link layer (Layer 2) of the network stack, for example, virtual local area networks (VLANs), tunnels, link aggregations, and the newer virtual NICs (VNICs), are configured by using the `dladm` command. Interfaces that are on the IP layer (Layer 3) are configured by using the `ipadm` command. See Chapter 2, "Creating and Administering Virtual Networks in Oracle Solaris," in *Using Virtual Networks in Oracle Solaris 11.1* and Chapter 6, "Configuring IP Tunnels," in *Configuring and Administering Oracle Solaris 11.1 Networks*.

For additional information about setting network properties, see Chapter 5, "Internet Protocol Suite Tunable Parameters," in *Oracle Solaris 11.1 Tunable Parameters Reference Manual*.

Displaying and Configuring Datalinks in Fixed Mode

When you perform a fresh installation, all datalinks are automatically assigned generic names by using the `net0`, `net1`, and `net`*N* naming convention, depending on the total number of network devices on a system. After the installation, you can use different datalink names. See Chapter 3, "Working With Datalinks," in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*.

Note – During an upgrade, link names that were used previously are retained.

Display information about the datalinks on a system, as follows:

```
# dladm show-phys
LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
net2              Ethernet            up         10000  full      hxge0
net3              Ethernet            up         10000  full      hxge1
net4              Ethernet            up         10     full      usbecm0
net0              Ethernet            up         1000   full      igb0
net1              Ethernet            up         1000   full      igb1
net9              Ethernet            unknown    0      half      e1000g0
net5              Ethernet            unknown    0      half      e1000g1
net10             Ethernet            unknown    0      half      e1000g2
net11             Ethernet            unknown    0      half      e1000g3
```

Note – In Oracle Solaris 10, the `/etc/path_to_inst` file can be used to store information about physical and virtual network devices. In Oracle Solaris 11, this file does not contain link names for physical network interfaces. To display this information, use the `dladm show-phys` command, as shown in the previous example.

Display a datalink name, its device name, and its location in this manner:

```
# dladm show-phys -L net0
LINK              DEVICE       LOC
net0              e1000g0      IOBD
```

Rename a datalink, as follows:

If an IP interface is configured over the datalink, first remove that interface:

```
# ipadm delete-ip interface
```

Then, change the link's current name:

```
# dladm rename-link old-linkname new-linkname
```

where *old-linkname* refers to the current name of the datalink and *new-linkname* refers to any name that you want to assign to the datalink. For more information, see Chapter 3, "Working With Datalinks," in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1* and Chapter 1, "Overview of Fixed Network Configuration," in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*.

EXAMPLE 7–3 Displaying a System's MAC Addresses

Display the MAC addresses of the physical links in a system, as follows:

```
# dladm show-phys -m
```

This command is similar to using the `ifconfig` command.

Display the MAC addresses of all of the links in a system, physical and non-physical, as follows:

```
# dladm show-linkprop -p mac-address
```

Configuring IP Interfaces and Addresses in Fixed Mode

The `ipadm` command is used to manually configure IP interfaces and addresses. For example, a static IPv4 interface is configured as follows:

```
# ipadm create-ip net0
# ipadm create-addr -T static -a local=10.9.8.7/24 net0
net0/v4
```

The `-T` option can be used to specify three address types: `static`, `dhcp`, and `addrconf` (for auto-configured IPv6 addresses). In this example, the system is configured with a static IPv4 address. You can use the same syntax to specify a static IPv6 address. However, static IPv6 addresses require that a link-local IPv6 address be configured prior to creating any static IPv6 addresses. This configuration is accomplished by creating an IPv6 `addrconf` address before creating the static IPv6 address:

```
# ipadm create-ip net0
# ipadm create-addr -T addrconf net0
net0/v6
# ipadm create-addr -T static -a local=ec0:a:99:18:209:3dff:fe00:4b8c/64 net0
net0/v6a
```

Configure an interface with DHCP, as follows:

```
# ipadm create-ip net0
# ipadm create-addr -T dhcp net0
net0/v6a
```

Use the `addrconf` argument with the `-T` option to specify an automatically generated IPv6 address:

```
# ipadm create-ip net0
# ipadm create-addr -T addrconf net0
net0/v6
```

If you wanted to change the IP address that was provided for the `net0` interface in the previous example, you would need to first remove the interface and then re-add it. For example:

```
# ipadm delete-addr net0/v4
# ipadm create-addr -T static -a local=10.7.8.9/24 net0
net0/v4
```

See also Chapter 2, "Configuring a System for the Network," in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1* and `ipadm`(1M).

Configuring Naming Services in Fixed Mode

The SMF repository is the primary repository for all naming services configuration. The previous behavior, where you modified a configuration file to configure naming services no longer works. These services must be enabled or refreshed for any changes to take affect.

Note – If no network configuration exists, naming services default to `files only` behavior, rather than `nis files`. The `svc:/system/name-service/cache` SMF service should be enabled at all times.

The following table describes the naming service configuration that has migrated to SMF.

TABLE 7–1 SMF Service to Legacy File Mapping

SMF Service	Files	Description
`svc:/system/name-service/switch:default`	`/etc/nsswitch.conf`	Naming service switch configuration (used by the `nscd` command)
`svc:/system/name-service/cache:default`	`/etc/nscd.conf`	Naming service cache (`nscd`)
`svc:/network/dns/client:default`	`/etc/resolv.conf`	DNS naming service

TABLE 7–1 SMF Service to Legacy File Mapping *(Continued)*

SMF Service	Files	Description
svc:/network/nis/domain:default	/etc/defaultdomain /var/yp/binding/$DOMAIN/*	Shared NIS domain configuration (used by all NIS services). Also historical shared use by LDAP naming services **Note –** Must be enabled when using nis/client or ldap/client
svc:/network/nis/client:default	Not applicable	NIS client naming service (ypbind and related files)
svc:/network/ldap/client:default	/var/ldap/*	LDAP client naming service (ldap_cachemgr and related files)
svc:/network/nis/server:default	Not applicable	NIS server naming service (ypserv)
svc:/network/nis/passwd:default	Not applicable	NIS server passwd service (rpc.yppasswdd)
svc:/network/nis/xfr:default	Not applicable	NIS server transfer naming service (ypxfrd)
svc:/network/nis/update:default	Not applicable	NIS server update naming service (rpc.ypupdated)
svc:/system/name-service/upgrade:default	Not applicable	Naming legacy file to SMF upgrade service

EXAMPLE 7–4 Configuring Naming Services by Using SMF

The following example shows how to configure DNS by using SMF commands.

```
# svccfg
svc:> select dns/client
svc:/network/dns/client> setprop config/search = astring: \
("us.company.com" "eu.company.com" "companya.com" "companyb.com" "company.com" )
svc:/network/dns/client> setprop config/nameserver = net_address: \
( 10.2.201.12 10.2.201.30 )
svc:/network/dns/client> select dns/client:default
svc:/network/dns/client:default> refresh
svc:/network/dns/client:default> validate
svc:/network/dns/client:default> select name-service/switch
svc:/system/name-service/switch> setprop config/host = astring: "files dns"
svc:/system/name-service/switch> select system/name-service/switch:default
svc:/system/name-service/switch:default> refresh
svc:/system/name-service/switch:default> validate
svc:/system/name-service/switch:default>
```

EXAMPLE 7–4 Configuring Naming Services by Using SMF *(Continued)*

```
# svcadm enable dns/client
# svcadm refresh name-service/switch
# grep host /etc/nsswitch.conf
hosts:  files dns
# cat /etc/resolv.conf
#
# copyright (c) 2011, Oracle and/or its affiliates. All rights reserved.
#

#
# _AUTOGENERATED_FROM_SMF_V1_
#
# WARNING: THIS FILE GENERATED FROM SMF DATA.
#   DO NOT EDIT THIS FILE.  EDITS WILL BE LOST.
# See resolv.conf(4) for details.

search    us.company.com eu.company.com companya.com companyb.com company.com
nameserver    10.2.201.12
nameserver    10.2.201.30
.
.
.
```

EXAMPLE 7–5 Setting Multiple NIS Servers by Using SMF

The following example shows how to set multiple NIS servers.

```
# svccfg -s nis/domain setprop config/ypservers = host: "(1.2.3.4 5.6.7.8)"
```

Note that there is space between `1.2.3.4` and `5.6.7.8`.

EXAMPLE 7–6 Setting Multiple DNS Options by Using SMF

The following example shows how to set multiple `/etc/resolv.conf` options.

```
# svccg
svc:> select /network/dns/client
svc:/network/dns/client> setprop config/options = "ndots:2 retrans:3 retry:1"
svc:/network/dns/client> listprop config/options
config/options astring     ndots:2 retrans:3 retry:1

# svcadm refresh dns/client
# grep options /etc/resolv.conf
options ndots:2 retrans:3 retry:1
svc:/network/dns/client> exit
```

`resolv.conf` Error-Checking Capabilities

Prior to the naming services migration to SMF, errors in the `resolv.conf` file configuration were processed silently and went undetected without producing any warnings. As a result, the `resolv.conf` file did not behave according to how it was configured. In Oracle Solaris 11, some

basic error checking is performed through the use of SMF templates so that error conditions are properly reported. Note that other SMF services also all have some rudimentary error-checking capabilities. However, the `resolv.conf` error reporting is the most prominent, due to the absence of error reporting in `libresolv2`. See `resolv.conf(4)`.

Temporarily Resetting SMF Naming Services

Reset the configuration properties of an SMF naming service back to the `files only` mode, as follows:

```
# /usr/sbin/nscfg unconfig name-service/switch
# svcadm refresh name-service/switch
```

Note – Refresh the `name-service` switch SMF service for the changes to take affect.

The `nscfg unconfig` command resets the SMF configuration *only*. The `sysconfig` command executes the appropriate SMF services, as well as resets SMF and `on disk` legacy files and services to their original state.

Importing Naming Services Configuration

The `nscfg` command transfers legacy file configuration for the `name-service` switch components into the SMF repository. The command imports the legacy file, converting and pushing the configuration to SMF. For example:

```
# /usr/sbin/nscfg import -f FMRI
```

The command that is used in the following example is the simplest way to populate the DNS configuration with information from the `resolv.conf` file. In this example, the `nscfg` command reads the information in the `/etc/resolv.conf` file, converts it, then stores the information in the `svc:/network/dns/client` SMF service.

```
# /usr/sbin/nscfg import -f dns/client
```

If your system is running in `files only` mode, and no naming services have been configured or enabled, use the `nscfg` command to manually configure the system, as shown here:

```
# vi /etc/resolv.conf
# /usr/sbin/nscfg import -f dns/client
# cp /etc/nsswitch.dns /etc/nsswitch.conf
# /usr/sbin/nscfg import -f name-service/switch
# svcadm enable dns/client
# svcadm refresh name-service/switch
```

For more information, see `nscfg(1M)`.

▼ How to Use a Legacy `nsswitch.conf` File

When you change a system's naming service, you need to modify the name service switch information accordingly.

1. **Become an administrator.**
2. **Copy the `nsswitch.conf` file to the new system.**
3. **Load the information from the file into the SMF repository.**

   ```
   # nscfg import -f svc:/system/name-service/switch:default
   ```

4. **Refresh the name service switch SMF service.**

   ```
   # svcadm refresh name-service/switch
   ```

Configuring LDAP in Fixed Mode

The easiest way to set up LDAP is to enable the `DefaultFixed` NCP and perform fixed network configuration. Then, if you want to use an LDAP proxy or LDAP self modes and some form of security credentials, run the `ldapclient` command to complete the LDAP setup. See `ldapclient(1M)`.

Managing Network Configuration in Reactive Mode

Reactive network configuration handles network connectivity and network configuration based on current network conditions through the use of several different types of profiles. Individual profiles contain properties that determine how the network is configured. These profiles are then activated and deactivated by the system, or by you. Assuming your site has a DHCP server that can provide IP addresses and name service information, reactive network configuration provides out-of-box functionality for automatic network configuration of a system that does not require manual configuration. See Chapter 1, "Reactive Network Configuration (Overview)," in *Connecting Systems Using Reactive Network Configuration in Oracle Solaris 11.1*.

When you are using reactive network configuration, the system automatically detects changes in network conditions and adjusts the network configuration accordingly, based on the new network environment. So, in situations where cables are regularly plugged or unplugged, cards are added or removed, etc., the system restores network connectivity without any user intervention. One disadvantage of using reactive network configuration is that you have less control over network configuration in the event that network conditions change.

For reactive network configuration, the system provides the `Automatic` NCP and the `Automatic` Location. These two profiles perform basic configuration of wired and wireless

networking. The only time you are required to interact with reactive networking is if you are prompted by the system for more information, for example, to provide a security key or password for a wireless network.

You can optionally create user-defined reactive NCPs and Locations that are configured with properties that you specify. Use the `netcfg` command, either in command-line mode or interactively to create reactive NCPs, Locations, and other profile types.

Part of the process of creating a reactive NCP is to configure the individual components that are contained within the NCP. These individual configuration objects are called Network Configuration Units (NCUs), and each NCU represents a physical link or an interface with properties that define the configuration for that particular link or interface, as shown in the output of the following output:

```
netcfg> select ncp myncp
netcfg:ncp:myncp> select ncu ip nge0
netcfg:ncp:myncp:ncu:nge0> list
ncu:nge0
    type                interface
    class               ip
    parent              "myncp"
    enabled             true
    ip-version          ipv4,ipv6
    ipv4-addrsrc        dhcp
    ipv6-addrsrc        dhcp,autoconf
```

The following examples show how to create and modify NCPs by using the `netcfg` command interactively. See Chapter 2, "Creating and Configuring Reactive Network Profiles (Tasks)," in *Connecting Systems Using Reactive Network Configuration in Oracle Solaris 11.1* for detailed instructions.

EXAMPLE 7–7 Creating a New Reactive NCP

In the following example, a new NCP named `myncp` and two NCUs (one link and one interface) are created.

```
$ netcfg
netcfg> create ncp myncp
netcfg:ncp:myncp> create ncu phys net0
Created ncu 'net0', Walking properties ...
activation-mode (manual) [manual|prioritized]>
mac-address>
autopush>
mtu>
netcfg:ncp:myncp:ncu:net0> end
Committed changes
netcfg:ncp:myncp> create ncu ip net0
Created ncu 'net0'. Walking properties ...
ip-version (ipv4,ipv6) [ipv4|ipv6]> ipv4
ipv4-addrsrc (dhcp) [dhcp|static]> dhcp
ipv4-default-route>
netcfg:ncp:myncp:ncu:net0> verify
```

EXAMPLE 7–7 Creating a New Reactive NCP *(Continued)*

```
All properties verified
netcfg:ncp:myncp:ncu:net0> end
Committed changes
netcfg:ncp:myncp> list
ncp:myncp
        management-type     reactive
NCUs:
        phys    net0
        ip      net0
netcfg:ncp:myncp> list ncu phys net0
ncu:net0
        type                    link
        class                   phys
        parent                  "myncp"
        activation-mode         manual
        enabled                 true
netcfg:ncp:myncp> list ncu ip net0
ncu:net0
        type                    interface
        class                   ip
        parent                  "myncp"
        enabled                 true
        ip-version              ipv4
        ipv4-addrsrc            dhcp
        netcfg:ncp:myncp> exit
```

In this example, because the `ipv4` value is selected, no prompt is displayed for the `ipv6-addrsrc` property, as this property is unused. Likewise, for the `phys` NCU, the default value (manual activation) for the `priority-group` property is accepted, so no other conditionally related properties are applied.

EXAMPLE 7–8 Creating a New Reactive NCP by Cloning the Automatic NCP

You can optionally create a new reactive NCP by cloning the `Automatic` NCP, then modifying its properties to set new network configuration parameters. This method is preferred to modifying the original system-defined `Automatic` NCP, because that NCP is subject to change if network conditions change. In the following example, a new NCP named `newncp` is created by cloning the system-defined `Automatic` NCP:

```
netcfg> list
NCPs:
    Automatic
    DefaultFixed
    bs
Locations:
    Automatic
    NoNet
    DefaultFixed
netcfg> create -t Automatic ncp newncp
netcfg:ncp:newncp> list
ncp:newncp
    management-type     reactive
NCUs:
```

EXAMPLE 7–8 Creating a New Reactive NCP by Cloning the Automatic NCP *(Continued)*

```
    phys    net1
    phys    net0
    ip    net1
    ip    net0
netcfg:ncp:newncp> destroy ncu ip net1
Destroyed ncu 'net1'
netcfg:ncp:newncp>  list
ncp:newncp
    management-type     reactive
NCUs:
    phys    net1
    phys    net0
    ip    net0
netcfg:ncp:newncp> exit
```

EXAMPLE 7–9 Creating an NCU for an Existing Reactive NCP

You can configure network settings for a reactive NCP when you create the profile, or you can modify an existing NCP by using the `netcfg select` command, as shown in the following example where an NCU is created for an existing NCP. The difference between the following example and Example 7–7 is that the `select` subcommand is used instead of the `create` subcommand. In the following example, an IP NCU for an existing NCP is created interactively.

```
$ netcfg
netcfg> select ncp myncp
netcfg:ncp:myncp> list
ncp:myncp
        management-type    reactive
NCUs:
        phys   net0
netcfg:ncp:myncp> create ncu ip net0
Created ncu 'net0'. Walking properties ...
ip-version (ipv4,ipv6) [ipv4|ipv6]> ipv4
ipv4-addrsrc (dhcp) [dhcp|static]> dhcp
ipv4-default-route>
netcfg:ncp:myncp:ncu:net0> end
Committed changes
netcfg:ncp:myncp> list
ncp:myncp
        management-type    reactive
NCUs:
        phys    net0
        ip      net0
netcfg:ncp:myncp> list ncu phys net0
ncu:net0
        type                    link
        class                   phys
        parent                  "myncp"
        activation-mode         manual
        enabled                 true
netcfg:ncp:myncp> list ncu ip net0
NCU:net0
        type                    interface
        class                   ip
```

EXAMPLE 7–9 Creating an NCU for an Existing Reactive NCP *(Continued)*

```
        parent                  "myncp"
        enabled                 true
        ip-version              ipv4
        ipv4-addrsrc            dhcp
        netcfg:ncp:myncp> exit
```

EXAMPLE 7–10 Configuring a Static IP Address for an existing NCP

In the following example, a static IP address is configured for an existing NCP.

```
netcfg> select ncp myncp
netcfg:ncp:myncp:ncu:nge0> list
ncu:nge0
    type                interface
    class               ip
    parent              "myncp"
    enabled             true
    ip-version          ipv4,ipv6
    ipv4-addrsrc        dhcp
    ipv6-addrsrc        dhcp,autoconf
netcfg:ncp:myncp:ncu:nge0> set ipv4-addrsrc=static
netcfg:ncp:myncp:ncu:nge0> set ipv4-addr=1.2.3.4/24
netcfg:ncp:myncp:ncu:nge0> set ipv4-default-route=1.2.3.1
netcfg:ncp:myncp:ncu:nge0> end
Committed changes
netcfg:ncp:myncp>
```

EXAMPLE 7–11 Enabling an NCP

In the following example, an NCP named `myncp` is enabled.

```
$ netadm enable -p ncp myncp
Enabling ncp 'myncp'
```

Configuring Naming Services in Reactive Mode

System-wide network configuration is managed in the Location profile. There are system-defined and user-defined Locations. Properties of user-defined Locations are configured by using the `netcfg` command. See Chapter 2, "Creating and Configuring Reactive Network Profiles (Tasks)," in *Connecting Systems Using Reactive Network Configuration in Oracle Solaris 11.1* for step-by-step instructions.

The following system-defined Locations are used for specific conditions and then automatically activated when those conditions are met:

- `DefaultFixed` – Is activated when the `DefaultFixed` NCP is active.

  You cannot manually enable the `DefaultFixed` Location or switch the active Location when the `DefaultFixed` NCP is active because fixed network configuration is being used. However, if a reactive NCP (`Automatic` or any user-defined NCP) is active, you can use the `netadm` command to manually enable any different Location, which can be either a system-defined Location (`Automatic` or `NoNet`), or any *manually enabled* user-defined Location.
- `Automatic` – Is activated when any reactive NCP is active, at least one IP address is "up", and there is no other user-defined Location with activation rules that make it a better match.

  The `Automatic` Location configures DNS through DHCP *only*.
- `NoNet` - Is activated when a reactive NCP is active and there are no "up" IP addresses.

Note – Before configuring naming service properties in a Location, you need to update the file that is to be referenced by the `nameservices-config-file` property of the specified Location. This file can be stored anywhere on the system. However, do not use the `/etc/nsswitch.conf` file name, as this file is overwritten.

Create a new user-defined Location profile and then configure NIS, as follows:

```
$ netcfg
netcfg> create loc officeloc
Created loc 'officeloc'.  Walking properties ...
activation-mode (manual) [manual|conditional-any|conditional-all]> conditional-all
conditions> advertised-domain contains oracle.com
nameservices (dns) [dns|files|nis|ldap]> nis
nameservices-config-file ("/etc/nsswitch.dns")> /etc/nsswitch.nis
nis-nameservice-configsrc [manual|dhcp]> dhcp
nfsv4-domain>
ipfilter-config-file>
ipfilter-v6-config-file>
ipnat-config-file>
ippool-config-file>
ike-config-file>
ipsecpolicy-config-file>
netcfg:loc:officeloc> end
Committed changes
netcfg> exit
```

In the following example NIS is configured for an existing Location.

```
$ netcfg> select loc origloc
netcfg:loc:origloc> set nameservices=nis
netcfg:loc:origloc> set nis-nameservice-configsrc=manual
netcfg:loc:origloc> set nis-nameservice-servers="1.2.3.38,1.3.3.36"
netcfg:loc:origloc> set default-domain="org.company.com"
```

```
netcfg:loc:origloc> set nameservices-config-file="/etc/nsswitch.nis"
netcfg:loc:origloc> end
Committed changes
netcfg> exit
```

Configuring LDAP in Reactive Mode

Reactive network configuration mode provides limited support for LDAP. Only LDAP anonymous mode works when in reactive mode. If you want to use an LDAP proxy or LDAP self modes and some form of security credentials you must first enable the `DefaultFixed` profile and manually configure your network. For instructions, see Chapter 12, "Setting Up LDAP Clients (Tasks)," in *Working With Naming and Directory Services in Oracle Solaris 11.1*.

Creating Persistent Routes (Fixed and Reactive)

The `/etc/defaultrouter` file is deprecated in Oracle Solaris 11. You can no longer manage routes (default or otherwise) by using this file. Also, after an installation, you cannot determine a system's default route by checking this file. Instead, choose from the following methods to determine the system's default route.

You can configure route information for a system in the following ways:

- For any currently active NCP (fixed or reactive), use the `route` command with the `-p` option to persistently add a route:

  ```
  # route -p add default ip-address
  ```

  Because this command applies the specified route to the currently active NCP, the default route is removed and potentially replaced, if the active NCP changes.

  Note – This behavior is true for all types of network configuration, not just the default route settings.

  For routes that are created by using this method, use the `route -p show` command to display all of the static routes that are associated with the currently active NCP:

  ```
  # route -p show
  ```

- Display the currently active routes on a system (applies to both types of NCPs) by using the `netstat` command:

  ```
  # netstat -rn
  ```

- Create a single, per-interface default route for any reactive NCP (active or non-active) by using the `netcfg` command. See Example 7–9.

  Display the default route for the NCP, as follows:

```
# netcfg "select ncp MY-STATIC; select ncu ip e1000g0; get ipv4-default-route"
        ipv4-default-route      "10.80.226.1"
```

Default routes that are created with the `netcfg` command can also be viewed by using the `netstat -rn` command, but only when the relevant NCP is active. Routes that are created in this way cannot be viewed by using the `route -p show` command.

For more information, see the `netstat(1M)` and `route(1M)` man pages.

Configuring IPMP in Oracle Solaris 11

IPMP works differently in Oracle Solaris 11 than in Oracle Solaris 10. One significant change is that IP interfaces are now grouped into a *virtual* IP interface, for example, `ipmp0`. The virtual IP interface serves all of the data IP addresses, while test addresses that are used for probe-based failure detection are assigned to an underlying interface such as `net0`. For more information, see "How IPMP Works" in *Managing Oracle Solaris 11.1 Network Performance*.

Oracle Solaris 11 also uses different commands for managing IPMP configuration. As a result, some configuration tasks are also performed differently. Refer to the following general work flow when transitioning from your existing IPMP configuration to the new IPMP model:

1. Make sure you are using fixed network configuration and that the `DefaultFixed` NCP is enabled on your system prior to configuring IPMP. See "How to Change the Active NCP On the System" in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*.
2. Ensure that MAC addresses on SPARC based systems are unique. See "How to Ensure That the MAC Address of Each Interface Is Unique" in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*.
3. Use the `dladm` command to configure datalinks. To use the same physical network devices within your IPMP configuration, you will need to first identify the datalinks that are associated with each device instance:

   ```
   # dladm show-phys
   LINK              MEDIA               STATE      SPEED  DUPLEX    DEVICE
   net1              Ethernet            unknown    0      unknown   bge1
   net0              Ethernet            up         1000   full      bge0
   net2              Ethernet            unknown    1000   full      e1000g0
   net3              Ethernet            unknown    1000   full      e1000g1
   ```

   If you previously used `e1000g0` and `e1000g1` for your IPMP configuration, you now use `net2` and `net3`. Note that datalinks can be based not only on physical links but also on aggregations, VLANs, VNICs, and so on. For more information, see "Displaying a System's Datalinks (dladm show-link)" in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*.
4. Use the `ipadm` to perform the following tasks:
   - Configure the network layer
   - Create IP interfaces

- Add IP interfaces to the IPMP group
- Add data IP addresses to the IPMP group

For detailed instructions, see "Configuring IPMP Groups" in *Managing Oracle Solaris 11.1 Network Performance*.

To learn more about how Oracle Solaris 11 network configuration commands map to Oracle Solaris 10 network configuration commands, see Appendix A, "Comparison Map: ifconfig and ipadm Commands," in *Connecting Systems Using Fixed Network Configuration in Oracle Solaris 11.1*.

Managing Network Configuration From the Desktop

You can manage network configuration from the desktop by using the Network Administration GUI (formerly NWAM). The tool is similar to using the `netcfg` and `netadm` commands. With the GUI, you can connect to a wired or wireless network, configure a new wired or wireless connection, create Location profiles, and activate or deactivate profiles. Managing reactive network configuration from the desktop works best for users of laptop computers, and in situations where network conditions change often, for example when switching from a home office to the wireless network at work, or when traveling.

Note – If the `DefaultFixed` NCP is currently active, you can view information about your network configuration and you can switch to another NCP, but to configure network settings for this NCP, you must use the `dladm` and `ipadm` commands.

Follow these general guidelines and best practices for managing network configuration from the desktop:

- When managing network configuration from the desktop, the simplest solution is to enable the system-generated `Automatic` NCP. See Example 7–2. At home, you can use this NCP to connect to your wireless network.
- If you decide you want to use a wired connection, plug in the Ethernet cable. Do not switch the default `Automatic` NCP. The network connection will automatically adapt from a wireless network connection to a wired network connection, without having to make any other changes to your existing network configuration.
- At the office, the same rules apply. If no Ethernet cable is plugged into the network, and the `Automatic` NCP is enabled, reactive networking is used and a wireless network connection is automatically established.
- If you switch to the `DefaultFixed` NCP, you must then configure the various network components manually by using the `dladm` and `ipadm` commands.
- Keep in mind that for both the home and office scenario, you must initially choose a wireless network and save it to your list of favorite wireless networks, if you have not already done so.

Choose a wireless network by using the network administration GUI or by running the `netadm select-wifi` command. For example:

```
$ netadm select-wifi net1
1: ESSID home BSSID 0:b:e:85:26:c0
2: ESSID neighbor1 BSSID 0:b:e:49:2f:80
3: ESSID testing BSSID 0:40:96:29:e9:d8
4: Other
Choose WLAN to connect to [1-4]: 1
```

- To view the status of your current network connection, hover your mouse over the Network Status notification icon located on the desktop or just click the icon. The Network Status notification icon also includes contextual menu for creating and managing network configuration with the GUI.

  If the Network Status notification icon is not visible in the desktop, start it by choosing System → Administration → Network. To start the GUI from the command line, run the `nwam-manager` command. See the `nwam-manager`(1M) man page in the JDS/GNOME man page collection for more information.
- IP-related configuration is managed in the Network Profile section of the Network Preferences dialog box. The Network Preferences icon is displayed in the upper right-hand corner of the desktop. Access the Network Preferences dialog box by clicking the Network Status notification icon located on the desktop or by choosing the Network Preferences option from the Network Status notification icon's contextual menu.

See Chapter 4, "Using the Network Administration Graphical User Interface," in *Connecting Systems Using Reactive Network Configuration in Oracle Solaris 11.1* or the online help for detailed instructions.

Network Configuration and Administration Commands (Quick Reference)

The following table describes the commands that are used to manage network configuration in both fixed and reactive mode.

Note – Starting with Oracle Solaris 11.1, you can use fixed networking commands to manage reactive NCPs, provided that the NCP is currently *active*. The command usage in the following table reflects this change.

TABLE 7–2 Commands That Are Used to Configure and Administer the Network

Configuration/Administration Task	Commands to Use in Reactive Mode	Commands to Use in Fixed Mode
Switch network configuration modes (enable or disable an NCP or a Location profile).	Enable the `Automatic` NCP: `netadm enable -p ncp Automatic` Enable any reactive NCP: `netadm enable -p ncp` *ncp-name* Enable a Location: `netadm enable -p loc` *loc-name*	Enable the `DefaultFixed` NCP: `netadm enable -p ncp DefaultFixed`
List the status of all network profiles on a system.	`netadm list`	`netadm list`
Configure link properties.	`netcfg "create ncp` *ncp-name*; `create ncu phys` *ncu-name*; set *property=value*"	`dladm set-linkprop -p` *property=value link*
Configure IP interfaces.	`netcfg "create ncp` *ncp-name*; `create ncu ip` *ncu-name*; set *property=value*"	`ipadm create-ip` *interface*
Configure IP addresses.	Static IP: `netcfg "select ncp` *ncp-name*; `select ncu ip` *ncu-name*; `set ipv4–addrsrc=static; set ipv4–addr=1.1.1.1/24"` DHCP: `netcfg "create ncp` *ncp-name*; `create ncu ip` *ncu-name*; `set ipv4–addrsrc=dhcp"`	IPv4 or IPv6 static address: `ipadm create-addr -T static -a` *IP-address address-object* IPv4 DHCP address: `ipadm create-addr -T dhcp` *address-object* Auto-generated IPv6 address based on a system's MAC address: `ipadm create-addr -T addrconf` *address-object*
Set the `netmask` property.	■ For the `Automatic` NCP: The DHCP server assigns the `netmask` property. ■ For other reactive NCPs: This property is set by using the `netcfg` command and is assigned as part of the static IP address. To assign the property, append the end of the IP address with *address/prefixlen* (`192.168.1.1/24`). View the `netmask` property: `ipadm show-addr`	This property is set by using the `ipadm` command, as part of the static IP address assignment. To assign the property, append the end of the IP address with *address/prefixlen* (`192.168.1.1/24`). See "How to Configure an IP Interface" in *Oracle Solaris Administration: Network Interfaces and Network Virtualization*. View the `netmask` property: `ipadm show-addr`

TABLE 7–2 Commands That Are Used to Configure and Administer the Network *(Continued)*

Configuration/Administration Task	Commands to Use in Reactive Mode	Commands to Use in Fixed Mode
Modify existing network configuration.	Configure link properties: `netcfg "select ncp` *ncp-name*; `select ncu phys` *ncu-name*; set *property=value*" Configure an IP interface: `netcfg "select ncp` *ncp-name*; `select ncu ip` *ncu-name*; set *property=value*"	`dladm set-linkprop -p` *property=value link* `ipadm set-prop [-t] -p prop=`*value*`[,...]` *protocol* `ipadm set-addrprop [-t] –p prop=`*value*`[,...]` *addrobj* `ipadm set-ifprop -p` *property=value interface* `ipadm set-prop -p property=`*value* `-m` *protocol interface* `ipadm set-addrprop -p property=`*value addrobj*
Configure or modify naming services (NIS and DNS).	Configure DNS from DHCP: `netcfg "create loc` *loc-name* ; `set dns-nameservice-configsrc=dhcp"` Manually configure DNS: `netcfg "create loc` *loc-name*; `set dns-nameservice-configsr=static; set dns-nameservice-servers=1.1.1.1; set dns-nameservice-search=foo.com"` For an existing Location: `netcfg "select..."`	Set parameters for naming services: `svccfg` and `svcadm`
Configure LDAP.	Only LDAP anonymous mode works when in reactive mode. To use an LDAP proxy or LDAP self-modes, enable the `DefaultFixed` NCP.	`ldapclient` or SMF commands to select LDAP.
Configure default route.	For any reactive NCP: `netcfg "select ncp` *ncp-name*; `select ncu ip` *ncu-name*; `set ipv4–default-route=1.1.1.1"` Persistently set a default route: `route -p add default` *routerIP-address*	Persistently set a default route: `route -p add default` *routerIP-address* Set any persistent route: `route -p add -net` *nIP-address* `-gateway` *gIP-address*

TABLE 7–2 Commands That Are Used to Configure and Administer the Network *(Continued)*

Configuration/Administration Task	Commands to Use in Reactive Mode	Commands to Use in Fixed Mode
Display default route.	`netstat -rn` displays all of the active routes currently in use by the kernel, regardless of how the route was configured For any reactive NCP configured with a single, per-interface default route: `netcfg "select ncp` *ncp-name*; `select ncu ip` *ncu-name*; `get ipv4–default-route"` `route -p show` displays all of the static routes that are associated with the currently active NCP, if it was added with the `route -p add` command	`netstat -rn` displays all of the active routes currently in use by the kernel, regardless of how the route was configured `route -p show` displays all of the static routes that are associated with the currently active NCP, if it was added with the `route -p add` command
Configure host name (nodename).	When the `Automatic` NCP is enabled, the SMF service property is only set if the DHCP server does not provide a value for the nodename/hostname option (DHCP standard option code 12). See `nodename(4)`	**Oracle Solaris 11:** `svccfg -s` sets the `config/nodename` property of the `svc:system/identity:node` SMF service to the desired name. **Oracle Solaris 11.1:** Use the `hostname` command. See `hostname(1)`.
Import naming service configuration.	Configured in the Locations profile.	`/usr/sbin/nscfg import -f` *FMRI* `nsccfg` exports existing legacy files into the SMF repository.
Unconfigure and reconfigure a system (including all network configuration).	Unconfigure an Oracle Solaris instance: `sysconfig unconfigure system` Reconfigure an Oracle Solaris instance: `sysconfig configure system`	

◆ ◆ ◆ CHAPTER 8

Managing System Configuration

This chapter provides information about the system configuration features and tools that are supported in Oracle Solaris 11 releases.

The following topics are covered:

Comparing Oracle Solaris 10 System Configuration to Oracle Solaris 11 System Configuration

TABLE 8–1 Oracle Solaris 10 and Oracle Solaris 11 System Configuration Comparison

System Configuration Feature, Tool, or Function	Oracle Solaris 10	Oracle Solaris 11	Oracle Solaris 11.1
System configuration (network, and naming service configuration)	Configured in various files within the `/etc` directory	Configured through properties of the appropriate SMF service	Configured through properties of the appropriate SMF service. See "Configuring Naming Services in Fixed Mode" on page 93

TABLE 8–1 Oracle Solaris 10 and Oracle Solaris 11 System Configuration Comparison *(Continued)*

System Configuration Feature, Tool, or Function	Oracle Solaris 10	Oracle Solaris 11	Oracle Solaris 11.1
System console service (serial port monitor) configuration	`getty`, `pmadm`, `ttyadm`, `ttymon`	Configured through properties of the appropriate SMF service	Configured through properties of the appropriate SMF service See "System Console and Terminal Services Changes" on page 118
System configuration (nodename/hostname)	Edit `/etc/nodename`	Configured through properties of the appropriate SMF service See "System Configuration Changes and Migration of System Configuration to SMF" on page 113	Use the `hostname` command. See `hostname(1)`
System logging	`syslog`	`syslog`	`syslog` (default) and `rsyslog` See "System Configuration Changes and Migration of System Configuration to SMF" on page 113
Power management	Edit `/etc/power.conf` file or use the `pmconfig` command	`poweradm`	`poweradm` See "Power Management Configuration Changes" on page 119
System unconfiguration and reconfiguration	By using the `sysidtool`. `sys-unconfig`, `sysidconfig`, and `sysidcfg` commands	`sysconfig` or the SCI tool	`sysconfig` or the SCI tool See "System Configuration Tools Changes" on page 120
System registration	Auto Registration feature Starting with Oracle Solaris 10 1/13: Oracle Configuration Manager	Oracle Configuration Manager	Oracle Configuration Manager See "System Registration and System Support Changes" on page 120

TABLE 8–1 Oracle Solaris 10 and Oracle Solaris 11 System Configuration Comparison *(Continued)*

System Configuration Feature, Tool, or Function	Oracle Solaris 10	Oracle Solaris 11	Oracle Solaris 11.1
System recovery	Flash archive features	Use backup boot environments (BEs) and system recovery procedures	Use backup boot environments (BEs) and system recovery procedures "System Boot, Recovery, and Platform Changes" on page 121
Printer configuration and administration	LP print commands, Solaris Print Manager	CUPS command-line, CUPS Print Manager, and CUPS web browser interface	CUPS command-line, CUPS Print Manager, and CUPS web browser interface See "Printer Configuration and Management Changes" on page 129
Locale and time zone configuration	Edit `/etc/default/init`	Configured through properties of the appropriate SMF service	Configured through properties of the appropriate SMF service See "Locale and Time Zone Configuration Changes" on page 133

System Configuration Changes and Migration of System Configuration to SMF

In Oracle Solaris 11, certain aspects of system configuration have migrated to SMF. For information about naming services that have migrated to SMF, see Table 7–1.

The following key changes are introduced in this release:

- **DNS server setup** – The process for setting up a DNS server has changed. For detailed instructions, see "Administering DNS (Tasks)" in *Working With Naming and Directory Services in Oracle Solaris 11.1*.
- `/etc/default/init` **file is now read-only** – Locale and time zone configuration have migrated to SMF. All changes to environment variables should be managed through the new `svc:/system/environment:init` SMF service.

  To use the `svc:/system/environment:init` SMF service, make sure the `skip_init_upgrade` property is set to `true`:

```
# svccfg -s svc:/system/environment:init setprop \
upgrade/skip_init_upgrade=true
# svcadm refresh svc:/system/environment:init
```

For more information, see "Internationalization and Localization Changes" on page 131.

- `/etc/dfs/dfstab` **configuration** – Publishing and unpublishing a file system share is now done with the `zfs` command. See Chapter 5, "Managing File Systems."
- `/etc/hostname.`<if>, `/etc/dhcp.`<if>, and `/etc/hostname.ip*.tun*` **configuration** – Persistent network configuration by editing these files is no longer necessary. The `ipadm` and `dladm` commands are used to manage this type of network configuration. See "Managing Network Configuration in Fixed Mode" on page 89.
- **Mapping a system's host name** – Depending on the Oracle Solaris 11 release you are running, a system's host name is mapped as follows during an installation:
  - **Oracle Solaris 11:** In Oracle Solaris 10, during an installation, the `/etc/hosts` file is updated to map the system's host name to one of the its non-loopback IP addresses. In Oracle Solaris 11, the host name is mapped to a system's IPv4 and IPv6 addresses. For example:

    ```
    ::1 foobar localhost
    127.0.0.1 foobar loghost localhost
    ```

    If you prefer the previous behavior, where `hostname` maps to the IP address of a non-loopback interface, you must manually modify the `/etc/hosts` file to include this type of mapping, as shown in the following example:

    ```
    ::1 localhost
    127.0.0.1 loghost localhost
    129.148.174.232 foobar
    ```

  - **Oracle Solaris 11.1:** The host name is mapped to the primary interface at installation time. The `system/identity:node` SMF service includes a property that enables an administrator to disable the feature.
- **Power management configuration** – Power management is no longer configured by editing the `/etc/power.conf` file and by using the `pmconfig` command. Instead, the `poweradm` command is used. See "Power Management Configuration Changes" on page 119.
- **Configuring a system's identity** – Depending on the Oracle Solaris 11 release that you are running, configure a system's identity as follows:
  - **Oracle Solaris 11:** Configure a system's identify (nodename/hostname) by setting the `config/nodename` service property of the `svc:/system/identity:node` SMF service, as shown in this example:

```
# svccfg -s svc:/system/identity:node setprop config/nodename = astring: nodename
# svcadm refresh svc:/system/identity:node
# svcadm restart svc:/system/identity:node
```

Note – If the system is configured to use DHCP, which is always the case if the `Automatic` NCP is enabled, the SMF service property can only be set if the DHCP server does not provide a value for the nodename/hostname option (DHCP standard option code 12). See `nodename(4)`.

- **Oracle Solaris 11.1:** Use the `hostname` command to permanently set the system's host name. Initially, the `hostname` value is stored in `config/nodename`, but this value is overridden if the system is configured by DHCP, in which case, DHCP provides the `hostname` value. If the `hostname` command is used, then the `hostname` value is whatever is specified in `config/nodename`. If you set a system's identity by using the `hostname` command, this setting cannot be overridden by DHCP until you execute the `hostname` command with the `-D` option. The corresponding SMF properties and the associated SMF service are also automatically updated when you use the `hostname` command. See `hostname(1)`.

- **System console and terminal services configuration** – The `sac` command and the Service Access Facility (SAF) program are no longer supported. The system console and locally connected terminal devices are represented as instances of the SMF `console-login` service, `svc:/system/console`. See "System Console, Terminal Services, and Power Management Changes" on page 118.
- **System logging services** – New in Oracle Solaris 11.1, `rsyslog` is a reliable and extended `syslog` daemon with a modular design implementation that supports several features, for example, filtering, TCP, encryption, high-precision timestamps, as well as output control.

  The status of the `system-log` services can be displayed by running the following command:

```
# svcs -a | grep system-log
disabled       Nov_21   svc:/system/system-log:rsyslog
online         Nov_30   svc:/system/system-log:default
```

Note – The `syslog` SMF service, `svc:/system/system-log:default`, continues to be the default logging service for in Oracle Solaris 11.

- **Time zone configuration** – In Oracle Solaris 10, the time zone is configured by editing the `/etc/TIMEZONE` (`/etc/default/init`) file. In Oracle Solaris 11, the `svc:/system/timezone:default` SMF service enables you set a system's time zone. See "Locale and Time Zone Configuration Changes" on page 133.

SMF Administrative Changes

Information for recording the source of properties, property groups, instances, and services has been added to the SMF repository. This information enables users to determine which settings are administrative customizations and those settings that are delivered with Oracle Solaris by a manifest.

The different settings by administrator, profile, or manifest are captured in *layers*. Use the `svccfg listprop` command with the new `-l` option to explore the values that are in each of the layers. The `svccfg -s` *service:instance* `listprop -l all` command lists all of the property groups and property values for the selected *service:instance*, with all of the layers that are available for each property group and the property value that is set. For example:

```
root@system1# svccfg -s mysvc:default listprop -l all
start                           method      manifest
start/exec                      astring     manifest            /var/tmp/testing/blah.ksh
start/timeout_seconds           count       manifest            600
start/type                      astring     manifest            method
stop                            method      manifest
stop/exec                       astring     manifest            /var/tmp/testing/blah.ksh
stop/timeout_seconds            count       manifest            600
stop/type                       astring     manifest            method
startd                          framework   manifest
startd/duration                 astring     manifest            transient
ifoo                            framework   site-profile
ifoo                            framework   manifest
ifoo/ibar                       astring     admin               adminv
ifoo/ibar                       astring     manifest            imanifest_v
ifoo/ibar                       astring     site-profile        iprofile_v
general                         framework   site-profile
general                         framework   manifest
general/complete                astring     manifest
general/enabled                 boolean     site-profile        true
general/enabled                 boolean     manifest            true
```

In this example, the property group `ifoo` shows the type of information that is listed when the new `-l` option is used.

By comparison, running the same command without the new `-l` options lists the information, as follows:

```
# svccfg -s mysvc:default listprop
start                           method
start/exec                      astring     /var/tmp/testing/blah.ksh
start/timeout_seconds           count       600
start/type                      astring     method
stop                            method
stop/exec                       astring     /var/tmp/testing/blah.ksh
stop/timeout_seconds            count       600
stop/type                       astring     method
startd                          framework
startd/duration                 astring     transient
ifoo                            framework
```

```
ifoo/ibar                       astring     adminv
general                         framework
general/complete                astring
general/enabled                 boolean     true
```

In addition, the `svccfg listcust` command can be used to list customizations *only*.

Services and instances that are delivered in standard locations (`/lib/svc/manifest`, `/var/svc/manifest`, and `/etc/svc/profile`) are now managed by the `manifest-import` SMF service. To completely remove these services from the system, an administrator should uninstall the package that delivers the supporting files. This change triggers the removal of the service or instance from the system. If the delivering files are not managed by a package, then removing the file and restarting the `manifest-import` service removes the services or instances that are delivered from the system entirely.

If the files cannot be removed, or the administrator does not want the service or instance to run on the system, and disabling the service or instance is not an option, the `svccfg delete` command can be used. The `svccfg delete` command is considered an administrative customization to the way the system is currently installed when the delivering files that are still present in the standard locations.

Note – The `svccfg delete` command does not delete the service. The command only hides the service from other SMF consumers.

To remove any administrative customizations, including customizations that were made by the `svccfg delete` command, and return to the configuration that is provided by the service manifest, use the `delcust` subcommand of the `svccfg` command *with care*. For example, you would list and delete all of the customization on `sendmail-client:default`, as follows:

```
# svccfg
  svc:> select svc:/network/sendmail-client:default
  svc:/network/sendmail-client:default> listcust
  config                        application admin           MASKED
  ...
  svc:/network/sendmail-client:default> delcust
   Deleting customizations for instance: default
```

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

SMF Manifest Creation Tool

New in Oracle Solaris 11.1, the `svcbundle` command enables you to generate SMF manifests. You can also use the command to generate profiles by specifying the `bundle-type` option. The generated bundle is entirely defined by the use of multiple `-s` options. Each NV pair is of the form, *name=value*. To generate a manifest, you must specify a `service-name` and `start-method`. When generating a manifest, the `svcbundle` command makes several basics

assumptions, which can be modified after the manifest is generated. For detailed instructions on using the `svcbundle` command, see `svcbundle(1M)`.

System Process Summary

Oracle Solaris 10 and Oracle Solaris 11 releases include system processes that perform a specific task, but typically do not require any administration.

Process	Description
`fsflush`	System daemon that flushes pages to disk
`init`	Initial system process that starts and restarts other processes and SMF components
`intrd`	System process that monitors and balances system load due to interrupts
`kmem_task`	System process that monitors memory cache sizes
`pageout`	System process that controls memory paging to disk
`sched`	System process that is responsible for OS scheduling and process swapping
`vm_tasks`	System process with one thread per processor that balances and distributes virtual memory related workloads across CPUs for better performance.
`zpool-`*pool-name*	System process for each ZFS storage pool containing the I/O `taskq` threads for the associated pool

System Console, Terminal Services, and Power Management Changes

The following system console, terminal services, and power management changes are introduced.

System Console and Terminal Services Changes

The `sac` command and the Service Access Facility (SAF) program are not supported in Oracle Solaris 11. The system console and locally connected terminal devices are represented as instances of the SMF console-login service, `svc:/system/console`. This service defines most of the behavior. Each instance can have specific overrides to the settings that are inherited from the service.

Note – The `sac` and `getty` modes of the `ttymon` command are no longer supported. However, the `ttymon express` mode is still supported.

If you want to offer login services on auxiliary terminals, use one of the following services:

- `svc:/system/console-login:terma`
- `svc:/system/console-login:termb`

The `ttymon` program is used to offer login services for these terminals. Each terminal uses a separate instance of the `ttymon` program. Command-line arguments that are passed by the service to the `ttymon` program govern the terminal's behavior. For more information, see Chapter 5, "Managing the System Console, Terminal Devices, and Power Services (Tasks)," in *Managing System Information, Processes, and Performance in Oracle Solaris 11.1*.

Power Management Configuration Changes

In Oracle Solaris 10, power management is administered by configuring the `/etc/power.conf` file and by using the `pmconfig` command. In Oracle Solaris 11, the `poweradm` command replaces the `pmconfig` command. Power administration now includes a small number of controls that manage platform and implementation details. The `poweradm` command enables you to simplify power administration by manipulating these small number of controls. For more information, see the `poweradm(1M)` man page.

Review the following potential power management transition issues:

- By default, suspend is not enabled on any system. To enable suspend and inspect this setting on systems that support this feature, use the `poweradm` command as follows:

  ```
  # poweradm set suspend-enable=true
  # poweradm get suspend-enable
  ```

- By default, the `administrative-authority` SMF service property of the `poweradm` command is set to the `platform` value. However, the power service goes into maintenance mode if the `administrative-authority` service property is set to the `smf` value *before* the `time-to-full-capacity` and `time-to-minimum-responsiveness` values have been set. If this problem occurs, you can recover as follows:

  ```
  # poweradm set administrative-authority=none
  # poweradm set time-to-full-capacity=
  # poweradm set time-to-minimum-responsiveness=
  # svcadm clear power
  # poweradm set administrative-authority=smf
  ```

- The GNOME power manager (GPM) feature, which runs when the GUI starts, changes the power management settings. This behavior is intentional to enable the integration of power management administration with GNOME Desktop behavior. See "Managing System Power Services" in *Managing System Information, Processes, and Performance in Oracle Solaris 11.1*.

System Configuration Tools Changes

An Oracle Solaris instance, which is defined as a boot environment in either a global or a non-global zone, is created and configured during an installation. After installing or creating an Oracle Solaris instance, you can unconfigure and reconfigure the instance by using the new `sysconfig` utility. This tool replaces the `sys-unconfig` and `sysidtool` utilities.

In Oracle Solaris 11, the `sysconfig configure` command produces similar results to the `sys-unconfig` command that is used to unconfigure and halt a system in Oracle Solaris 10. For example:

```
# sysconfig configure -s
This program will re-configure your system.
Do you want to continue (y/(n))? y
```

The following example shows how to unconfigure a previously configured Oracle Solaris instance and leave it in an unconfigured state:

```
# sysconfig unconfigure -g system
```

You can also reconfigure an Oracle Solaris instance by specifying an existing configuration XML profile:

```
# sysconfig configure -c profile-name.xml
```

If you do not specify an existing configuration profile prior to an installation, the SCI tool launches during the installation process. The SCI tool enables you to provide specific configuration information for that Oracle Solaris instance. The SCI tool consists of a series of interactive panels that enable you to provide configuration information as part of a text installation. You can also run the tool on an installed Oracle Solaris system to create a new system configuration profile that is based on specifications that you provide.

Start the SCI tool from the command line, as follows:

```
# sysconfig configure
```

See the `sysconfig`(1M) man page and See Chapter 6, "Unconfiguring or Reconfiguring an Oracle Solaris instance," in *Installing Oracle Solaris 11.1 Systems*.

System Registration and System Support Changes

Oracle Configuration Manager is used to personalize and enhance the customer support experience by collecting configuration information and uploading it to the Management Repository. This information is then analyzed by customer support representatives to provide better service to customers. Benefits of using this feature include reduced time for problem resolution, proactive problem avoidance, and access to best practices and the Oracle knowledge

base. In some Oracle Solaris 10 releases, the Auto Registration feature performs a similar function. Starting with the Oracle Solaris 10 1/13 release, Oracle Configuration Manager replaces the Auto Registration feature.

You can configure the Oracle Configuration Manager and the Oracle Auto Service Request features during an interactive installation, if you plan to install these features on your system. Several options are available to choose from during an installation, including the ability to start Oracle Configuration Manager in the *disconnected mode*. This option replaces the "opt out" choice that is available in the Oracle 11 11/11 release. If you choose the disconnected mode option, no data is sent to My Oracle Support during the first reboot after an installation. Note that you can manually activate Oracle Configuration Manager later. See "Using Oracle Configuration Manager" in *Installing Oracle Solaris 11.1 Systems*.

Oracle Auto Service Request (ASR) is a secure, customer-installable feature of your Oracle or Sun hardware warranty and Oracle Premier Support for Systems. ASR assists in resolving specific hardware faults that occur by automatically opening service requests for Oracle's qualified server, storage, Exadata and Exalogic systems. The Oracle Auto Service Request is integrated with My Oracle Support. For more information, go to `http://www.oracle.com/technetwork/systems/asr/overview/index.html`.

System Boot, Recovery, and Platform Changes

The system boots from a ZFS root file system in Oracle Solaris 11. By default, the ZFS root file system is contained within a ZFS root pool, named `rpool`. Creating a UFS file system is still supported, but you cannot boot from a UFS or a Solaris Volume Manager root file system in this release.

Review the following information that impacts the way the system is booted for recovery purposes:

- If you use a system's service processor (SP) or ILOM to recover from a system problem, accessing a system's SP or ILOM is identical to previous releases. The differences mostly pertain to how the system is booted after you get to a SPARC based system's `ok` PROM prompt or to an x86 based system's firmware screen (BIOS or UEFI).
- In Oracle Solaris 10, you use flash archive features to create a copy of a UFS or ZFS root environment and then restore the flash archive to recover the system environment, in the case of a system or device failure.

  In Oracle Solaris 11, the system recovery process includes the following steps:

  - Archiving the root pool snapshots on a remote system
  - Replacing any failed system component or device
  - Recreating the root pool and setting the `bootfs` property
  - Restoring the previously archive root pool snapshots
  - Manually installing the boot blocks

See Chapter 11, "Archiving Snapshots and Root Pool Recovery," in *Oracle Solaris 11.1 Administration: ZFS File Systems*.

- When attempting to boot a system for recovery purposes, if the system cannot be booted, but the failure is not because the root pool is unavailable, you can use new boot options from the installation media or from an installation server to resolve the problem. See "Booting for System Recovery" on page 123.

GRUB, Firmware, and Disk Labeling Changes

Starting with Oracle Solaris 11.1, the following changes are introduced:

- **GRUB 2 is the default boot loader on x86 platforms** – GRUB 2 replaces the original GRUB 0.97-based boot loader (GRUB Legacy). GRUB 2 fully supports booting from disks that are larger than 2TB. GRUB 2 also supports the Unified Extensible Firmware Interface (UEFI) and the GUID Partition Table (GPT) partitioning scheme.
- **GRUB menu changes** – Unlike the editable `menu.lst` file that is used by GRUB Legacy, GRUB 2 uses a configuration file named `grub.cfg` that is syntactically different than the legacy `menu.lst` file. The `grub.cfg` file stores most of the GRUB configuration and is managed *solely* by using the `bootadm` command. To accommodate this change, the `bootadm` command has been expanded to include several new subcommands and a new `-P` option that enables you to administer the GRUB configuration for multiple root pools.

  Note – Because GRUB configuration changes can automatically overwrite changes that are made to the `grub.cfg` file, *do not* manually edit this file. See Chapter 2, "Administering the GRand Unified Bootloader (Tasks)," in *Booting and Shutting Down Oracle Solaris 11.1 Systems* and `bootadm`(1M).

- **Management of non-Oracle Solaris boot entries** – GRUB 2 includes an additional configuration file named `custom.cfg`. This file can be used to add custom menu entries to the GRUB configuration. The `custom.cfg` file does not exist on the system by default. You must create the file and it must be stored in the same location as the `grub.cfg` file (*/pool-name*`/boot/grub/`). During the boot process, GRUB checks for the `custom.cfg` file in the top-level dataset of the root pool (`boot/grub`). If the file exists, GRUB sources the file and processes any commands within the file as if the contents were actually part of the `grub.cfg` file. See "Customizing the GRUB Configuration" in *Booting and Shutting Down Oracle Solaris 11.1 Systems*.
- **64-Bit UEFI firmware support** – Oracle Solaris now supports x86 based systems with 64-bit UEFI firmware. An installation on UEFI firmware is supported through the DVD, USB, and network installation methods. UEFI version 2.1+ is required.

  If you are booting a system with UEFI firmware from the network, the boot process has changed slightly. See "Booting Systems With UEFI and BIOS Firmware From the Network" in *Booting and Shutting Down Oracle Solaris 11.1 Systems* for more details.

- **Booting from GPT labeled disks**– GPT labeled disks are now supported on both SPARC and x86 platforms. Installing Oracle Solaris 11.1 on an x86 or SPARC based system with GPT-aware firmware applies a GPT disk label on the root pool disk that uses the entire disk in most cases. Otherwise, installing Oracle Solaris 11.1 on a SPARC based system applies an SMI (VTOC) label to the root pool disk with a single slice 0.

  For SPARC based systems that support a GPT labeled boot disk, see "Support for GPT Labeled Disk Added on SPARC Platforms " in *Oracle Solaris 11.1 Release Notes* for more information about how to apply the GPT-aware firmware update.

If you are running a release that supports GRUB Legacy and are moving to a release that supports GRUB 2, see "Upgrading Your GRUB Legacy System to a Release That Supports GRUB 2" in *Booting and Shutting Down Oracle Solaris 11.1 Systems*.

Booting for System Recovery

The following error and recovery scenarios are similar to previous releases:

- You can use the `boot -a` command to bypass a problem in the `/etc/system` file. When prompted, use syntax that is similar to the following:

  ```
  Name of system file [/etc/system]:  /dev/null
  ```

  Press Return at the other prompts, as needed.

- A backup BE is created automatically during most `pkg update` operations. This feature enables you to boot back to a previous BE in case some error occurs during the image update process. Consider creating a backup BE before you make a system configuration change.

  ```
  # beadm create solaris-backup
  # beadm list
  BE             Active Mountpoint Space  Policy Created
  --             ------ ---------- -----  ------ -------
  solaris        R      -          4.01G  static 2013-02-08 16:53
  solaris-backup N      /          47.95M static 2013-02-11 10:48
  ```

  See "How to Boot From a Backup BE for Recovery Purposes" on page 124 for the steps to boot from a backup BE.

- Boot from the installation media or from an install server over the network to recover from a problem that is preventing the system from booting or to recover from a lost `root` password.

  On SPARC based systems, the `boot net:dhcp` command replaces the `boot net` command that is used in Oracle Solaris 10 releases.

- Boot a system in single-user mode to resolve a minor problem, such as correcting the root shell entry in the `/etc/passwd` file or changing a NIS server.

- Resolving a boot configuration problem generally involves importing the root pool, mounting the BE, and fixing the problem, for example reinstalling a corrupt x86 boot loader.

How to Boot From a Backup BE for Recovery Purposes

Booting the failsafe archive is no longer supported on SPARC and x86 platforms. Whenever possible, use up-to-date backup BEs for recovery purposes. BEs are bootable instances of the Oracle Solaris image, plus any other application software packages that are installed into that image. Multiple BEs reduce risk when updating software because the backup BE preserves the original BE.

You can create a new BE based on an active or an inactive boot environment. Or, you can create a new BE based on a clone of your original BE. A clone copies the root dataset and everything hierarchically under the main root dataset of the original BE. See *Creating and Administering Oracle Solaris 11.1 Boot Environments*.

If the system does not boot from the active BE, select a backup BE from which to boot.

- **Boot from a backup BE, as follows:**

  - **SPARC: Boot the system so that you can select an alternate or backup BE.**

    a. **Boot with the `boot -L` command.**

    ```
    ok boot -L
    ```

    b. **Select an alternate or backup BE.**

    ```
    Boot device: /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@0,0:a
    File and args: -L
    1 Oracle Solaris 11.1 SPARC
    2 solaris-backup
    Select environment to boot: [ 1 - 2 ]: 2
    ```

    In the previous output, the active BE is `Oracle Solaris 11.1 SPARC`, which most likely does not match the actual BE name, but it represents the current BE.

    c. **Boot the backup BE.**

    After you select the BE from which to boot, identify the on-screen boot path and type that information at the prompt.

    ```
    To boot the selected entry, invoke:
    boot [<root-device>] -Z rpool/ROOT/solaris-backup

    Program terminated
    {0} ok boot -Z rpool/ROOT/solaris-backup
    ```

    If the system does not boot, review the additional boot recovery steps in "How to Boot the System For Recovery Purposes" on page 125.

- **x86: Boot the system to identify the alternate or backup BE from the GRUB menu.**

  a. **When the GRUB menu is displayed, identify the backup BE.**

  ```
  GNU GRUB  version 1.99,5.11.0.175.1.0.0.14.0

  ****************************************************************************
  *solaris                                                                   *
  *solaris-1                                                                 *
  *                                                                          *
  *                                                                          *
  *                                                                          *
  *                                                                          *
  ****************************************************************************
  ```

  b. **Select the backup BE, then press Return to boot that entry.**

  If the system does not boot from the backup BE, review the additional boot recovery steps in "How to Boot the System For Recovery Purposes" on page 125.

How to Boot the System For Recovery Purposes

1 **Select the appropriate boot method.**

Note – On x86 platforms, use the `reboot` command with the `-p` option to initiate a standard reboot of the system, which enables you to view the GRUB menu or select an installation option. Otherwise, the system defaults to a fast reboot.

- **x86: Live Media** – Boot from the installation media and use a GNOME terminal for the recovery procedure.
- **SPARC: Text installation** – Boot from the install media or from the network, and select option `3 Shell` from the text installation screen.
- **x86: Text installation** – From the GRUB menu, select the `Text Installer and command line` boot entry, then select the option `3 Shell` from the text installation screen.
- **SPARC: Automated installation** – Use the following command to boot directly from an installation menu that allows you to exit to a shell.

  ```
  ok boot net:dhcp
  ```

- **x86: Automated installation** – Boot from an install server on the network that supports a PXE boot. Select the `Text Installer and command line` entry from the GRUB menu. Then, select the option `3 Shell` from the text installation screen.

For example, after the system is booted, select option `3 Shell`:

```
1  Install Oracle Solaris
2  Install Additional Drivers
3  Shell
4  Terminal type (currently xterm)
```

```
        5  Reboot

Please enter a number [1]: 3
To return to the main menu, exit the shell
#
```

2 Select from the following boot recovery problems:

- **Resolve a bad root shell by booting the system to single-user mode and correcting the shell entry in the `/etc/passwd` file.**
  - **On x86 based systems, boot the system in single-user mode by editing the selected boot entry in the GRUB menu. Add the `-s` option to end of the `$multiboot` line.**

    ```
    $multiboot /ROOT/s11u1_24b/@/$kern $kern -B $zfs_bootfs -s
    ```

  - **On SPARC based systems, shut down the system and boot in single-user mode. After you log in as `root`, edit the `/etc/passwd` file, and fix the root shell entry.**

    ```
    # zpool import -f rpool
    # beadm list
    be_find_current_be: failed to find current BE name
    BE        Active Mountpoint Space Policy Created
    --        ------ ---------- ----- ------ -------
    solaris   -      -          7.74M static 2013-02-09 09:40
    solaris-1 R      -          4.08G static 2013-02-13 07:24
    # mkdir /a
    # beadm mount solaris-1 /a
    # TERM=vt100
    # export TERM
    # cd /a/etc
    # vi shadow
    <Carefully remove the unknown password>
    # cd /
    # beadm umount solaris-1
    # halt


    # init 0
    ok boot -s

    Boot device: /pci@780/pci@0/pci@9/scsi@0/disk@0,0:a File and args: -s
    SunOS Release 5.11 Version 11.1 64-bit
    Copyright (c) 1983, 2012, Oracle and/or its affiliates. All rights
    reserved.
    Booting to milestone "milestone/single-user:default".
    Hostname: tardis.central
    Requesting System Maintenance Mode
    SINGLE USER MODE

    Enter user name for system maintenance (control-d to bypass): root
    Enter root password (control-d to bypass): xxxxxxx
    single-user privilege assigned to root on /dev/console.
    Entering System Maintenance Mode

    Feb 13 00:21:31 su: 'su root' succeeded for root on /dev/console
    Oracle Corporation SunOS 5.11 11.1 September 2012
    ```

```
su: No shell /usr/bin/mybash.  Trying fallback shell /sbin/sh.
root@tardis.central:~# TERM=vt100; export TERM
root@tardis.central:~# vi /etc/passwd
root@tardis.central:~# <Press control-d>
logout
svc.startd: Returning to milestone all.
```

- **x86: Resolve a corrupt boot loader problem by reinstalling it, as follows:**

  a. **Follow the directions in Step 1 to boot the system from media or from the network, then import the root pool.**

     ```
     # zpool import -f rpool
     ```

  b. **Reinstall the boot loader.**

     ```
     # bootadm install-bootloader -f  -P pool-name
     ```

     where `-f` forces the installation of the boot loader and bypasses any versioning checks for not downgrading the version of the boot loader on the system. The `-p` option is used to specify the root pool.

     Note – Do not used the `-f` option unless you are sure that you want to overwrite the boot loader with the version that is on the media. See "Installing GRUB 2 by Using the bootadm install-bootloader Command" in *Booting and Shutting Down Oracle Solaris 11.1 Systems*.

     ---

  c. **Export the root pool.**

     ```
     # zpool export pool-name
     ```

  d. **Reboot the system.**

- **Resolve an unknown `root` password problem that prevents you from logging into the system.**

  a. **Follow the instructions in Step 1 to boot from media or from the network, then import the root pool (`rpool`) and mount the BE to remove the `root` password entry.**

     This process is identical on SPARC and x86 platforms.

  b. **Set the `root` password by booting in single-user mode and setting the password.**

     This step assumes that you have removed an unknown `root` password in the previous step.

     - **On x86 based systems, edit the selected boot entry in the GRUB menu, adding the `-s` option at the end of the `$multiboot` line.**

       ```
       $multiboot /ROOT/s11u1_24b/@/$kern $kern -B $zfs_bootfs -s
       ```

- **On SPARC based systems, boot the system to single-user mode, log in as `root`, and set the `root` password. For example:**

```
ok boot -s

Boot device: /pci@780/pci@0/pci@9/scsi@0/disk@0,0:a File and args: -s
SunOS Release 5.11 Version 11.1 64-bit
Copyright (c) 1983, 2012, Oracle and/or its affiliates. All rights reserved.
Booting to milestone "milestone/single-user:default".
Hostname: tardis.central
Requesting System Maintenance Mode
SINGLE USER MODE

Enter user name for system maintenance (control-d to bypass): root
Enter root password (control-d to bypass): <Press return>
single-user privilege assigned to root on /dev/console.
Entering System Maintenance Mode

Feb 13 00:58:42 su: 'su root' succeeded for root on /dev/console
Oracle Corporation      SunOS 5.11      11.1    September 2012
root@tardis.central:~# passwd -r files root
New Password: xxxxxx
Re-enter new Password: xxxxxx
passwd: password successfully changed for root
root@tardis.central:~# <Press control-d>
logout
svc.startd: Returning to milestone all.
```

Boot, Platform, and Hardware Changes

Note the following boot, platform, and hardware feature changes in Oracle Solaris 11:

- **x86 platform support is 64–bit only** – Support for booting a 32–bit kernel on x86 platforms has been removed. Systems that have 32-bit hardware must either be upgraded to 64–bit hardware or continue to run Oracle Solaris 10. Note that 32–bit applications are not impacted by this change.
- **Bitmapped console support** – Oracle Solaris 11 includes support for high resolution and color depth consoles. By default, your machine will boot with a 1024x768x16-bit console, unless your video card does not support this setting. In which case, the setting will fall back to 800x600, then finally to 640x480. The console type (and also the older VGA TEXT 640x480 console) can be controlled through both kernel parameters and through options that you specify by editing the GRUB menu at boot time, as follows:

  ```
  -B console={text|graphics|force-text}
  ```

  See "Redirecting the Oracle Solaris Console at Boot Time" in *Booting and Shutting Down Oracle Solaris 11.1 Systems*.
- **Fast Reboot support on x86 and SPARC platforms** – On x86 platforms, Fast Reboot implements an in-kernel boot loader that loads the kernel into memory and then switches to that kernel. For SPARC based systems that support the Fast Reboot feature, the boot process is accelerated by skipping certain POST tests.

The Fast Reboot feature works differently on SPARC platforms than it does on x86 platforms. To initiate a fast reboot of a SPARC based system, use the `-f` option with the `reboot` command. Because Fast Reboot is the default behavior on x86 platforms, the `-f` option is not required. Use either the `reboot` command or the `init 6` command to initiate a fast reboot of an x86 based system. The Fast Reboot feature is managed through SMF properties that can be enabled or disabled, as required. For more details, see "Accelerating the Reboot Process" in *Oracle Solaris Administration: Common Tasks*.

- **Removal of support for the SPARC `sun4u` architecture** – With the exception of the M-series (OPL) hardware, you cannot boot Oracle Solaris 11 on the `sun4u` architecture. If you attempt to boot Oracle Solaris 11 on one of these systems, the following error message is displayed:

```
Rebooting with command: boot
Error: 'cpu:SUNW,UltraSPARC-IV+' is not supported by this release of Solaris.
NOTICE: f_client_exit: Program terminated!
```

Printer Configuration and Management Changes

The legacy LP print service has been replaced by the Common UNIX Printing System (CUPS). CUPS is a modular, open-source printing system that uses the Internet Printing Protocol (IPP) as the basis for managing printers, print requests, and print queues. CUPS supports network printer browsing and PostScript Printer Description-based printing options. CUPS also provides a common printing interface across a local network.

Removal of the LP Print Service

The following important changes are a result of the removal of the LP print service:

- Solaris Print Manager is no longer available in the desktop. CUPS Print Manager replaces this tool. See "Setting Up Printers by Using CUPS Print Manager" in *Oracle Solaris Administration: Common Tasks*.
- Several LP print commands, files, and services are no longer available. Some LP print commands, for example `lp`, `lpadmin`, `lpc`, `lpr` are still available. However, these commands are now managed by CUPS. For a complete list of the commands, services, and files that have been removed, refer to "Removal of Legacy System Management Commands, Tools, Services, and Files" on page 20.
- Printer configuration that is stored in the NIS naming service in Oracle Solaris 10 is not used by CUPS. CUPS auto-discovers printers on a network, enabling you to print to these printers without any manual configuration. Administrators can share network printers that are configured by using CUPS by turning on the sharing feature. See "How to Unshare or Share a Printer" in *Configuring and Managing Printing in Oracle Solaris 11.1*.

- In Oracle Solaris 10 and previous releases, the `/etc/printers.conf` file is where details about all of the printers that are set up by using the LP print service is stored. In Oracle Solaris 11, this file is no longer generated after a fresh installation. Any information about printers that were configured by using `lp` print commands is removed. The resulting behavior is as though these printers were never configured on the system. Any previously configured printers must be reconfigured by using CUPS. Note that you do not need to delete existing printers prior to reconfiguring them. For information about setting up your printing environment to work with CUPS, see "How to Set Up Your Printing Environment After Installing Oracle Solaris 11" on page 130.
- Printers that are configured on a per-user basis in the `~/.printers` file no longer work. Printer configuration is now solely managed by using CUPS. The default printer can be set on a per-user basis, by setting either the `LPDEST` or `PRINTER` environment variables, or by using the new `lpoptions` command. The `lpoptions` command creates an `~/.lpoptions` file that has the default printer entry listed within that file. By default, all print jobs are directed to this printer.

  List specific options for a printer as follows:

  ```
  # lpoptions -l printer-name
  ```

  Set the default destination or instance for the default printer by using the `-d` option:

  ```
  # lpoptions -d printer-name
  ```

  See "Setting a Default Printer" in *Configuring and Managing Printing in Oracle Solaris 11.1*.
- The `lp` entry in the `/etc/passwd` file is now as follows:

  ```
  lp:x:71:8:Line Printer Admin:/:
  ```

  The `lp` entry in the `/etc/group` file remains as it is in previous releases.

See Chapter 1, "Setting Up and Administering Printers by Using CUPS (Overview)," in *Configuring and Managing Printing in Oracle Solaris 11.1*.

How to Set Up Your Printing Environment After Installing Oracle Solaris 11

Use the following procedure to set up your printing environment to work with CUPS after a fresh installation.

1 Verify that the `cups/scheduler` and the `cups/in-lpd` SMF services are online.

```
# svcs -a | grep cups/scheduler
# svcs -a | grep cups/in-lpd
```

2 If these services are not online, enable them.

```
# svcadm enable cups/scheduler
# svcadm enable cups/in-lpd
```

3. **Check that the `printer/cups/system-config-printer` package is installed.**

   ```
   # pkg info print/cups/system-config-printer
   ```

   - **If the package is already installed, you are ready to configure printers by using CUPS.**
   - **If the package is not installed, install the package:**

     ```
     # pkg install print/cups/system-config-printer
     ```

Next Steps For instructions, see "Setting Up and Administering Printers by Using CUPS Command-Line Utilities" in *Oracle Solaris Administration: Common Tasks*.

Internationalization and Localization Changes

Oracle Solaris 11 introduces the following internationalization and localization changes:

- **Language and locale support** – Oracle Solaris 11 supports over 200 locales. By default, only a core set of locales is installed on the system. Core locales typically provide better support at the level of localized messages than locales that are available for additional installation. Specific Oracle Solaris components, such as the installers or Package Manager, are localized for core locales *only*. Note that localized messages for third-party software, for example GNOME and Firefox, include additional locales.

  The core set of locales support the following languages:

  - Chinese – Simplified (`zh_CN.UTF-8`)
  - Chinese – Traditional (`zh_TW.UTF-8`)
  - English (`en_US.UTF-8`)
  - French (`fr_FR.UTF-8`)
  - German (`de_DE.UTF-8`)
  - Italian (`it_IT.UTF-8`)
  - Japanese (`ja_JP.UTF-8`)
  - Korean (`ko_KR.UTF-8`)
  - Portuguese – Brazilian (`pt_BR.UTF-8`)
  - Spanish (`es_ES.UTF-8`)

  Notable core locale changes include the addition of the Portuguese – Brazilian locale and the removal of the Swedish locale.

- **Oracle Solaris 11.1 locale changes** – The following locale changes are introduced in this release:

  - Japanese (`ja_JP.UTF-8@cldr`) locale – This locale is a new variant of the Japanese UTF-8 locale (`ja_JP.UTF-8`) that conforms to the Unicode Common Locale Data Repository (CLDR) for the Japanese locale. The locale is an optional component that is installable from the `system/locale/extra` package.

- Local data for Simplified Chinese, Traditional Chinese, Korean, and Thai UTF-8 locales has been updated to support Unicode 6.0.

- **Language and locale packaging** – The locale facet mechanism replaces the `localeadm` command in Oracle Solaris 11. In Oracle Solaris 10, optional package components, such as documentation, localization, or debug files are split into separate packages. In Oracle Solaris 11, IPS enables you to store these various package components in the same package by using special tags that are called *facets*. Facets simplify the packaging process, as well as minimize disk space usage. Locale facets are used to mark files or actions that are language or locale-specific.

  Display the status of the facets on a system as follows:

  ```
  $ pkg facet
  ```

  The following example shows how to install the `Danish` locale and any available translations:

  ```
  # pkg change-facet facet.locale.da=True
  # pkg change-facet facet.locale.da_DK=True
  ```

  Note – Non-UTF-8 locales, such as `da_DK.ISO8859-1` , are packaged separately. To enable these locales, install the `system/locale/extra` package.

  See "Controlling Installation of Optional Components" in *Adding and Updating Oracle Solaris 11.1 Software Packages*.

- **Setting a system's default locale** – In Oracle Solaris 10, the default system locale is configured in `/etc/default/init`. In Oracle Solaris 11, this file is obsoleted, and the configuration has moved to the corresponding properties of the `svc:/system/environment:init` SMF service. See "Locale and Time Zone Configuration Changes" on page 133.

- **Short form locales** – Solaris 10 supports a number of short form locale names that do not follow the *language_country.encoding*[*@modifier*] format, for example, `ja`,`de`, `de_AT`, and so on. These locales are not present in Oracle Solaris 11 in their original form, only as aliases to fully qualified locale names through the `locale_alias` mechanism. See `locale_alias(5)`. In Oracle Solaris 11, it is recommended that the fully qualified locale names be used instead. Or, if possible, use UTF-8 locales. For more information, see the end-of-feature announcements at `http://www.oracle.com/technetwork/systems/end-of-notices/eonsolaris11-392732.html`.

- **Locale aliasing** – Locale aliases are new in Oracle Solaris 11. Locale name aliases are accepted and mapped to the corresponding canonical locale names. For example, the `de` locale is mapped to the canonical locale `de_DE.ISO8859-1`. For all of the locale name mappings, see `locale_alias(5)`.

- **Keyboard layout setting for the console** – In Oracle Solaris 11, the keyboard layout setting for the console has migrated to SMF. To change the keyboard layout in the console, modify the `keymap/layout` property of the `system/keymap:default` SMF service. The following example shows how to set the `UK-English` layout for the console.

```
# svccfg -s keymap:default setprop keymap/layout = UK-English
# svcadm refresh keymap
# svcadm restart keymap
```

Note – The keyboard layout in the graphical interface is set separately.

Locale and Time Zone Configuration Changes

Locale and time zone configuration is set in the `/etc/default/init` file in Oracle Solaris 10.

In Oracle Solaris 11, this configuration is managed through the following SMF service properties:

- Locale: `svc:/system/environment:init`
- Time zone: `svc:/system/timezone:default`

For example, to change the default system locale to `fr_FR.UTF-8`, you would configure the SMF service property as follows:

```
# svccfg -s svc:/system/environment:init \
setprop environment/LANG = astring: fr_FR.UTF-8
# svcadm refresh svc:/system/environment
```

The service must be refreshed for changes to take affect.

1. For the time zone setting, make sure the `TZ` in the `/etc/default/init` file is set to `localtime`.

```
grep TZ /etc/default/init
TZ=localtime
```

2. Then, set the time zone SMF property to the required time zone.

```
# svccfg -s timezone:default setprop timezone/localtime= astring: US/Mountain
# svcadm refresh timezone:default
```

For other date and time configuration changes in this release, see "Configuring Date and Time Before and After an Installation" on page 42.

◆ ◆ ◆ CHAPTER 9

Managing Security

This chapter describes security feature changes in Oracle Solaris 11 releases.

The following topics are covered:

Security Feature Changes

Oracle Solaris 11 introduces the following key security changes:

- **Address Space Layout Randomization (ASLR)** – Starting with Oracle Solaris 11.1, ASLR randomizes addresses that are used by a given binary. ASLR causes certain types of attacks that are based on knowing the exact location of certain memory ranges to fail and detects the attempt when it likely stops the executable. Use the `sxadm` command to configure ASLR. Use the `elfedit` command to change the tagging on a binary. See `sxadm(1M)` and `elfedit(1)`.
- **Administrative Editor** – Starting with Oracle Solaris 11.1, you can use the `pfedit` command to edit system files. If defined by the system administrator, the value of this editor is `$EDITOR`. If undefined, the editor defaults to the `vi` command. Start the editor as follows:

  ```
  $ pfedit system-filename
  ```

  See the `pfedit(1M)` man page and Chapter 3, "Controlling Access to Systems (Tasks)," in *Oracle Solaris 11.1 Administration: Security Services*.
- **Auditing** – Auditing is a now a service and is enabled by default. No reboot is required when disabling or enabling this service. The `auditconfig` command is used to view information about audit policy and to change audit policy. The auditing of public objects generates less noise in the audit trail. In addition, auditing of non-kernel events has no performance impact.

For information about creating a ZFS file system for audit files, see "How to Create ZFS File Systems for Audit Files" in *Oracle Solaris 11.1 Administration: Security Services*.

- **Audit Remote Server (ARS)** – ARS is a feature that receives and stores audit records from a system that is being audited and is configured with an active `audit_remote` plug-in. To distinguish an audited system from an ARS, the audited system can be termed the locally audited system. This feature is new in Oracle Solaris 11.1. Refer to the information about the `-setremote` option in the `auditconfig(1M)` man page.
- **Basic Audit Reporting Tool (BART)** – The default hash that is used by BART is now SHA256, not MD5. In addition to SHA256 being the default, you can also select the hash algorithm. See Chapter 6, "Verifying File Integrity by Using BART (Tasks)," in *Oracle Solaris 11.1 Administration: Security Services*.
- **Cryptographic Framework** – This feature now includes more algorithms, mechanisms, plug-ins, and support for Intel and SPARC T4 hardware acceleration. Also, Oracle Solaris 11 provides better alignment with the NSA Suite B cryptography.
- **Kerberos DTrace providers** – A new DTrace USDT provider that provides probes for Kerberos messages (Protocol Data Unit) has been added. The probes are modeled after the Kerberos message types that are described in RFC4120.
- **Key Management enhancements:**
  - PKCS#11 keystore support for RSA keys in the Trusted Platform Module
  - PKCS#11 access to Oracle Key Manager for centralized enterprise key management
- `lofi` **command changes** – `lofi` now supports the encryption of block devices. See `lofi(7D)`.
- `profiles` **command changes** – In Oracle Solaris 10, the command is only used to list profiles for a specific user or role, or a user's privileges for specific commands. In Oracle Solaris 11, you can also create and modify profiles in files and in LDAP by using the `profiles` command, See `profiles(1)`.
- `sudo` **command** – The `sudo` command is new in Oracle Solaris 11. This command generates Oracle Solaris audit records when running commands. The command also drops the `proc_exec` basic privilege, if the `sudoers` command entry is tagged as `NOEXEC`.
- **ZFS file system encryption** – ZFS file system encryption is designed to keep your data secure. See "Encrypting ZFS File Systems" on page 144.
- `rstchown` **property** – The `rstchown` tunable parameter that is used in previous releases to restrict `chown` operations is now a ZFS file system property, `rstchown`, and is also a general file system mount option. See *Oracle Solaris 11.1 Administration: ZFS File Systems* and `mount(1M)`.

  If you attempt to set this obsolete parameter in the `/etc/system` file, the following message is displayed:

  ```
  sorry, variable 'rstchown' is not defined in the 'kernel'
  ```

Network Security Features

The following network security features are supported:

- **Internet Key Exchange (IKE) and IPsec** – IKE now includes more Diffie-Hellman groups and can also use Elliptic Curve Cryptography (ECC) groups. IPsec includes AES-CCM and AES-GCM modes and is now capable of protecting network traffic for the Trusted Extensions feature of Oracle Solaris (Trusted Extensions).
- **IPfilter Firewall** – IPfilter Firewall, which is similar to the open source IPfilter feature, is compatible, manageable, and now highly integrated with SMF. This feature enables selective access to ports, based on IP address.
- **Kerberos** – Kerberos is now capable of mutual authentication of clients and servers. Also, support for initial authentication by using X.509 certificates with the `PKINIT` protocol has been introduced. See Part VI, "Kerberos Service," in *Oracle Solaris 11.1 Administration: Security Services*.
- **Secure by Default** – In Oracle Solaris 10, this feature was introduced, but was `netservices limited` and was also turned off by default. In Oracle Solaris 11, this feature is enabled. The Secure by Default feature is used to disable and protect several network services from attack and provides minimization of network exposure. Note that only SSH is enabled.
- **SSH** – Support for host and user authentication by using X.509 certificates is now available.

Pluggable Authentication Module Changes

The following Pluggable Authentication Module (PAM) changes are introduced:

- **Module to enable per-user PAM stacks** – Enables you to configure the PAM authentication policy on a per-user basis, when used in conjunction with the new RBAC `pam_policy` key (`user_attr(4)`). The default `pam.conf` file has also been updated to enable you to use this feature by specifying the `pam_policy` in a user's extended attributes or in a profile that is assigned to a user. For example:

  ```
  # usermod -K pam_policy=krb5_only username
  ```

  See `pam_user_policy(5)`.

- **PAM configuration in** `/etc/pam.d` – Adds support for configuring PAM by using per-service files. As a result, the contents of the `/etc/pam.conf` file have been migrated to multiple files within the `/etc/pam.d/` directory, based on the relevant PAM service name. This mechanism is now the method for configuring PAM in Oracle Solaris and is the default method that is used for all *new* installations. The `/etc/pam.conf` file is still consulted, so any existing or new changes that are made to this file continue to be recognized.

  If you have never edited the `/etc/pam.conf` file, the file only contains comments that direct you to the per-service equivalents in the `/etc/pam.d/` directory. If you previously edited the `/etc/pam.conf` file, for example, to enable LDAP or Kerberos, a new file name named `/etc/pam.conf.new` is delivered with the changes you made. See `pam.conf(4)`.

- `definitive` **flag added to** `pam.conf` – The `pam.conf` file now includes the `definitive control_flag`. See `pam.conf(4)`.

Removed Security Features

The following security features are excluded from Oracle Solaris 11:

- **Automated Security Enhancement Tool (ASET)** – The ASET functionality is replaced by a combination of IPfilter, which includes `svc.ipfd`, BART, SMF, and other security features that are supported in Oracle Solaris 11.
- **Smartcard** – Smartcard support is no longer available.

Roles, Rights, Privileges, and Authorizations

The following information describes how roles, rights, privileges, and authorizations work in Oracle Solaris 11:

- **Assign versus delegate authorizations** – Oracle Solaris provides authorizations for delegating specific administrative rights to individual users and roles to implement separation of duty. In Oracle Solaris 10, authorizations ending in `.grant` are required to delegate an authorization to another user. In Oracle Solaris 11, two new suffixes, `.assign` and `.delegate`, are used, for example, `solaris.profile.assign` and `solaris.profile.delegate`. The former grants the right to delegate any rights profile to any user or role. The latter is more restrictive, in that only the rights profiles that are already assigned to the current user can be delegated. Since the `root` role is assigned `solaris.*`, this role can assign any authorization to any user or role. As a safety measure, no authorizations that end in `.assign` are included in any profiles by default.
- `groupadd` **Command Changes** – At group creation, the system now assigns the `solaris.group.assign/`*groupname* authorization to the administrator. This authorization gives the administrator complete control over that group, enabling the administrator to modify or delete the *groupname*, as needed. For more information, see the `groupadd(1M)` and `groupmod(1M)` man pages.
- **Media Restore rights profile** – This rights profile and set of authorizations can escalate the privileges of a non `root` account. The profile exists, but is not part of any other rights profile. Because the `Media Restore` rights profile provides access to the entire root file system, its use is a possible escalation of privilege. Deliberately altered files or substitute media could be restored. By default, the `root` role includes this rights profile.
- **Primary Administrator profile removed** – The initial user that is created at installation time is given the following roles and rights:
  - `root` role
  - `System Administrator` rights profile

  - Access to the `sudo` command for all commands that are run as `root`
- **Role authentication** – You can specify either `user` or `role` for the `roleauth` keyword. See `user_attr(4)`.
- **root as a Role** – `root` is now a role by default, therefore, not *anonymous* and cannot remotely log in to a system. For information about changing the `root` role to a user, see "How to Change the root Role Into a User" in *Oracle Solaris 11.1 Administration: Security Services*.
- **Oracle Solaris basic privileges include the following:**
  - `file_read`
  - `file_write`
  - `net_access`
- **Profile shell versions of regular shells** – Every regular shell now has its own profile version. The following profile shells are available:
  - `pfbash`
  - `pfcsh`
  - `pfksh`
  - `pfksh93`
  - `pfrksh93`
  - `pfsh`
  - `pftcsh`
  - `pfzsh`

  See `pfexec(1)`.
- **Rights profiles** – The `user_attr`, `prof_attr`, and `exec_attr` databases are now read-only. These local files databases are assembled from fragments that are located in `/etc/user_attr.d`, `/etc/security/prof_attr.d`, and `/etc/security/exec_attr.d`. The fragment files are not merged into a single version of the file, but left as fragments. This change enables packages to deliver complete or partial RBAC profiles. Entries that are added to the local files repository with the `useradd` and `profiles` commands are added to the `local-entries` file in the fragment directory. To add or modify a profile, use the `profiles` command. See "About Rights Profiles" on page 140.
- **Stop rights profile** – This profile enables administrators to create restricted accounts. See "RBAC Rights Profiles" in *Oracle Solaris 11.1 Administration: Security Services*.
- `pfsh` *script* **command** – This command now runs the same as the `pfsh -c` *script* command. Previously, commands within a script would not be able to take advantage of RBAC, unless the script specified a profile shell as its first line. This rule required you to modify any scripts to use RBAC, which is now unnecessary because the caller of the script (or an ancestor within the session) can specify a profile shell.

- `pfexec` **command** – This command is now no longer setuid root. The new `PF_PFEXEC` process attribute is set when the `pfexec` command or a profile shell is executed. Then, the kernel sets the appropriate privileges on `exec`. This implementation ensures that sub-shells are empowered or restricted, as appropriate.

  When the kernel is processing an `exec`(2), it treats setuid to `root` differently. Note that setuid to any other uid or setgid is as it was previously. The kernel now searches for an entry in the `Forced Privilege` RBAC profile in `exec_attr`(4) to determine which privileges the program should run with. Instead of having the program start with uid `root` and all privileges, the program runs with the current uid and only the additional privileges that the `Forced Privilege` RBAC execution profile have assigned to that path name.

About Rights Profiles

Rights profiles are collections of authorizations and other security attributes, commands with security attributes, and supplementary rights profiles. Oracle Solaris provides many rights profiles. You can modify existing rights profiles, as well as create new ones. Note that rights profiles must be assigned in order, from most to least powerful.

The following are some of the rights profiles that are available:

- **System Administrator** – Is a profile that is able to perform most tasks that are not connected with security. This profile includes several other profiles to create a powerful role. Use the profiles command to display information about this profile. See Example 9–1.
- **Operator** – Is a profile with limited capabilities to manage files and offline media.
- **Printer Management** – Is a profile that provides a limited number of commands and authorizations to handle printing.
- **Basic Solaris User** – Is a profile that enables users to use the system within the bounds of security policy. This profile is listed by default in the `policy.conf` file.
- **Console User** – Is a profile for the workstation owner. This profile provides access to authorizations, commands, and actions for the person who is seated at the computer.

Other rights profiles that are available in this release include the All rights profile and the Stop rights profile. For more information, see Chapter 10, "Security Attributes in Oracle Solaris (Reference)," in *Oracle Solaris 11.1 Administration: Security Services*.

EXAMPLE 9–1 Displaying Information About the System Administrator Rights Profile

Use the `profiles` command to display information about a specific rights profile. In the following example, information about the System Administrator rights profile is displayed:

```
$ profiles -p "System Administrator" info
name=System Administrator
        desc=Can perform most non-security administrative tasks
        profiles=Install Service Management,Audit Review,Extended Accounting Flow
```

EXAMPLE 9–1 Displaying Information About the System Administrator Rights Profile *(Continued)*

```
Management,Extended Accounting Net Management,Extended Accounting Process Management,
Extended Accounting Task Management,Printer Management,Cron Management,Device Management,
File System Management,Log Management,Mail Management,Maintenance and Repair,
Media Backup,Media Catalog,Media Restore,Name Service Management,Network Management
Object Access Management,Process Management,Project Management,RAD Management,
Service Operator,Shadow Migration Monitor,Software Installation,System
Configuration,User Management,ZFS Storage Management
        help=RtSysAdmin.html
```

Viewing Privileges and Authorizations

When a user is directly assigned privileges, in effect, the privileges are in every shell. When a user is not directly assigned privileges, then the user must open a profile shell. For example, when commands with assigned privileges are in a rights profile that is in the user's list of rights profiles, then the user must execute the command in a profile shell.

To view privileges online, see `privileges(5)`. The privilege format that is displayed is used by developers.

```
$ man privileges
Standards, Environments, and Macros                 privileges(5)

NAME
     privileges - process privilege model
...
     The defined privileges are:

     PRIV_CONTRACT_EVENT

         Allow a process to request reliable delivery  of  events
         to an event endpoint.

         Allow a process to include events in the critical  event
         set  term  of  a  template  which  could be generated in
         volume by the user.
...
```

EXAMPLE 9–2 Viewing Directly-Assigned Privileges

If you have been directly assigned privileges, then your basic set contains more than the default basic set. In the following example, the user always has access to the `proc_clock_highres` privilege.

```
$ /usr/bin/whoami
jdoe
$ ppriv -v $$
1800:   pfksh
flags = <none>
        E: file_link_any,...,proc_clock_highres,proc_session
        I: file_link_any,...,proc_clock_highres,proc_session
```

EXAMPLE 9–2 Viewing Directly-Assigned Privileges *(Continued)*

```
        P: file_link_any,...,proc_clock_highres,proc_session
        L: cpc_cpu,dtrace_kernel,dtrace_proc,dtrace_user,...,sys_time
$ ppriv -vl proc_clock_highres
        Allows a process to use high resolution timers.
```

To view authorizations, use the `auths` command:

```
$ auths list
```

The output of this command produces a more readable summary (one per line) of the authorizations that are assigned to a user. Starting with Oracle Solaris 11.1, several new options have been added to the `auths` command. For example, the `check` option is useful for scripting. Other new options provide the ability to add, modify, and remove authorizations to and from `files` or LDAP. See `auths(1)`.

File and File System Security Changes

The following sections describe changes to file and file system security.

aclmode Property Is Reintroduced

The `aclmode` property that determines how the ACL permissions on a file are modified during a `chmod` operation is reintroduced in Oracle Solaris 11. The `aclmode` values are `discard`, `mask`, and `passthrough`. The `discard` default value is the most restrictive, and the `passthrough` value is the least restrictive.

EXAMPLE 9–3 ACL Interaction With `chmod` Operations on ZFS Files

The following examples illustrate how specific `aclmode` and `aclinherit` property values influence the interaction of existing ACLs with a `chmod` operation that either reduces or expands any existing ACL permissions to be consistent with the ownership of a group.

In this example, the `aclmode` property is set to `mask` and the `aclinherit` property is set to `restricted`. The ACL permissions in this example are displayed in compact mode, which more easily illustrates changing permissions.

The original file and group ownership and ACL permissions are as follows:

```
# zfs set aclmode=mask pond/whoville
# zfs set aclinherit=restricted pond/whoville

# ls -lV file.1
-rwxrwx---+  1 root     root      206695 Aug 30 16:03 file.1
```

EXAMPLE 9–3 ACL Interaction With chmod Operations on ZFS Files *(Continued)*

```
              user:amy:r-----a-R-c---:-------:allow
             user:rory:r-----a-R-c---:-------:allow
         group:sysadmin:rw-p--aARWc---:-------:allow
            group:staff:rw-p--aARWc---:-------:allow
                 owner@:rwxp--aARWcCos:-------:allow
                 group@:rwxp--aARWc--s:-------:allow
               everyone@:------a-R-c--s:-------:allow
```

A chown operation changes the file ownership on `file.1` and the output is now seen by the owning user, amy. For example:

```
# chown amy:staff file.1
# su - amy
$ ls -lV file.1
-rwxrwx---+  1 amy      staff     206695 Aug 30 16:03 file.1
              user:amy:r-----a-R-c---:-------:allow
             user:rory:r-----a-R-c---:-------:allow
         group:sysadmin:rw-p--aARWc---:-------:allow
            group:staff:rw-p--aARWc---:-------:allow
                 owner@:rwxp--aARWcCos:-------:allow
                 group@:rwxp--aARWc--s:-------:allow
               everyone@:------a-R-c--s:-------:allow
```

The following chmod operation changes the permissions to a more restrictive mode. In this example, the modified sysadmin group's and staff group's ACL permissions do not exceed the owning group's permissions.

```
$ chmod 640 file.1
$ ls -lV file.1
-rw-r-----+  1 amy      staff     206695 Aug 30 16:03 file.1
              user:amy:r-----a-R-c---:-------:allow
             user:rory:r-----a-R-c---:-------:allow
         group:sysadmin:r-----a-R-c---:-------:allow
            group:staff:r-----a-R-c---:-------:allow
                 owner@:rw-p--aARWcCos:-------:allow
                 group@:r-----a-R-c--s:-------:allow
               everyone@:------a-R-c--s:-------:allow
```

The following chmod operation changes the permissions to a less restrictive mode. In this example, the modified sysadmin group's and staff group's ACL permissions are restored to allow the same permissions as the owning group.

```
$ chmod 770 file.1
$ ls -lV file.1
-rwxrwx---+  1 amy      staff     206695 Aug 30 16:03 file.1
              user:amy:r-----a-R-c---:-------:allow
             user:rory:r-----a-R-c---:-------:allow
         group:sysadmin:rw-p--aARWc---:-------:allow
            group:staff:rw-p--aARWc---:-------:allow
                 owner@:rwxp--aARWcCos:-------:allow
                 group@:rwxp--aARWc--s:-------:allow
               everyone@:------a-R-c--s:-------:allow
```

Encrypting ZFS File Systems

In previous Oracle Solaris releases and in this release, the Cryptographic Framework feature provides the `encrypt`, `decrypt`, and `mac` commands to encrypt files.

Oracle Solaris 10 does not support ZFS encryption, but Oracle Solaris 11 supports the following ZFS encryption features:

- ZFS encryption is integrated with the ZFS command set. Like other ZFS operations, key change and rekey operations are performed online.
- You can use your existing storage pools as long as they are upgraded. You have the flexibility of encrypting specific file systems.
- ZFS encryption is inheritable to descendent file systems. Key management can be delegated through ZFS delegated administration.
- Data is encrypted by using AES (Advanced Encryption Standard) with key lengths of 128,192, and 256 in the CCM and GCM operation modes.
- ZFS encryption uses the Cryptographic Framework feature, which gives it access to any available hardware acceleration or optimized software implementations of the encryption algorithms automatically.

Note – Currently, you cannot encrypt a ZFS root file system or other OS components, such as the `/var` directory, even if it is a separate file system.

EXAMPLE 9–4 Creating an Encrypted ZFS File System

The following example shows how to create an encrypted ZFS file system. The default encryption policy is to prompt for a `passphrase`, which must be a minimum of 8 characters in length.

```
# zfs create -o encryption=on tank/data
Enter passphrase for 'tank/data': xxxxxxxx
Enter again: xxxxxxxx
```

The default encryption algorithm is `aes-128-ccm` when a file system's encryption value is `on`.

After an encrypted file system is created, it cannot be unencrypted. For example:

```
# zfs set encryption=off tank/data
cannot set property for 'tank/data': 'encryption' is readonly
```

For more information, see "Encrypting ZFS File Systems" in *Oracle Solaris 11.1 Administration: ZFS File Systems*.

Immutable Zones

New in Oracle Solaris 11, the `file-mac-profile` property enables you to run zones with a read-only root file system. This feature enables you to choose between four predefined profiles that determine how much of a zone file system is read-only only, even for processes that have `allzone` privileges. See "zonecfg file-mac-profile Property" in *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.

◆ ◆ ◆ CHAPTER 10

Managing Oracle Solaris Releases in a Virtual Environment

This chapter describes the virtualization features that are supported in Oracle Solaris 11 releases.

The following topics are covered:

Installing and Managing Oracle Solaris 11 Virtualization Features

The following table provides a brief description of virtualization features that are supported in Oracle Solaris 11.

TABLE 10–1 Oracle Solaris 11 Virtualization Features

Oracle Solaris 11 Feature	Description	Oracle Solaris 10 Support	For More Information
Oracle Solaris Resource Manager product components (also referred to as resource management)	Features that enable you to control how applications use available system resources	Yes	Part I, "Oracle Solaris Resource Management," in *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*
Oracle VM Server for SPARC (Sun Logical Domains)	Hypervisor-based virtualization for SPARC servers	Yes	*Oracle VM Server for SPARC 2.2 Administration Guide*

TABLE 10–1 Oracle Solaris 11 Virtualization Features *(Continued)*

Oracle Solaris 11 Feature	Description	Oracle Solaris 10 Support	For More Information
Oracle VM Server 3.0 for x86 (Xen)	Hypervisor-based virtualization for x86 based servers	Yes	`http://www.oracle.com/us/technologies/virtualization/oraclevm/index.html`
Oracle VM VirtualBox	Hosted workstation and server virtualization for x86 based systems	Yes	`http://www.oracle.com/technetwork/server-storage/virtualbox/downloads/index.html`
Oracle Solaris Zones	A zone is a virtualized operating system environment created within a single instance of the Oracle Solaris operating system	Yes	Part II, "Oracle Solaris Zones," in *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*

Consolidating Legacy Solaris Systems With Oracle VM Server

You can use the Oracle VM Server for SPARC physical-to-virtual (P2V) conversion tool to automatically convert an existing physical system to a virtual system that runs the Oracle Solaris 10 OS in a logical domain on a chip multithreading (CMT) system. Run the `ldmp2v` command from a control domain that runs the Oracle Solaris 10 OS or the Oracle Solaris 11 OS to convert one of the following source systems to a logical domain:

- Any `sun4u` SPARC based system that runs at least the Solaris 8, Solaris 9, or Oracle Solaris 10 OS
- Any `sun4v` system that runs the Oracle Solaris 10 OS, but does not run in a logical domain

Note that the `ldmp2v` command does not support any SPARC based system that runs the Oracle Solaris 10 OS with a ZFS root or the Oracle Solaris 11 OS.

See Chapter 13, "Oracle VM Server for SPARC Physical-to-Virtual Conversion Tool," in *Oracle VM Server for SPARC 2.2 Administration Guide*.

Oracle Solaris 11 Zone Features

- **Oracle Solaris 10 branded zones** – Oracle Solaris 10 Zones provide an Oracle Solaris 10 environment on Oracle Solaris 11. You can migrate an Oracle Solaris 10 system or zone to a `solaris10` zone on an Oracle Solaris 11 system in the following ways:
  - Create a zone archive and use the archive to create an `s10zone` on the Oracle Solaris 11 system. See "Transitioning an Oracle Solaris 10 Instance to an Oracle Solaris 11 System" on page 152.

  - Detach the zone from the Oracle Solaris 10 system and attach the zone on the Oracle Solaris 11 zone. The zone is halted and detached from its current host. The `zonepath` is moved to the target host, where it is attached. See "About Detaching and Attaching the solaris10 Zone" in *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.

- **Oracle Solaris 11 installation support** – You can specify the configuration and installation of non-global zones as part of an AI client installation. Non-global zones are installed and configured on the first reboot after the global zone is installed. See Chapter 12, "Installing and Configuring Zones," in *Installing Oracle Solaris 11.1 Systems*.
- **Whole root zones only** – Oracle Solaris Zones are whole-root type only, but you can configure your zones in a more flexible way, such as when disk space is limited or if you prefer a read-only zone root configuration. By default, zone boot environments are compressed.

  In addition, you can automatically update any non-global zone to ensure consistency across the system. An added benefit is individual software stacks for each non-global zone are independent of the global zone.
- **Legacy branded zones** – The following legacy branded zone features are only supported in Oracle Solaris 10 releases:
  - Linux brand (`lx`)
  - Oracle Solaris 8 Containers (`solaris8`)
  - Oracle Solaris 9 Containers (`solaris9`)
- **Exclusive IP zones by default** – Exclusive-IP zones enable you to assign a separate IP stack per zone. Each zone has the flexibility to configure IP within that stack completely separate to other zones. You can easily observe network traffic, per zone, and apply individual network resources. In previous versions of Oracle Solaris this was dependent on the number of physical NICs per system. The addition of network virtualization provides enhanced flexibility when managing zones, without the restrictions of physical network hardware. Newly created zones in Oracle Solaris 11 are exclusive-IP zones with a VNIC, `net0`, whose underlying lower link is automatically selected at boot time. See Part II, "Oracle Solaris Zones," in *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.
- **Network virtualization for zones** – Oracle Solaris 11 network virtualization features can be applied to a zone by creating a virtual NIC (VNIC) for the zone and applying bandwidth limits and traffic flows to the zone's assigned VNIC. The VNIC is created when the zone boots and deleted when the zone halts, and will be created within the non-global zone's datalink namespace. This feature allows you to provision a zone without having to learn the details of network configuration and topology. If you want to assign a preexisting datalink to an exclusive-IP zone, you can still do so during the zone configuration.

  For example, create a virtual NIC, limit the SPEED of the VNIC, create an address for it, and then assign it to zone.

```
# dladm create-vnic -l net0 -p maxbw=600 vnic0
# ipadm create-addr -T static -a local=x.x.x.x/24 vnic0/v4static
```

```
.
.
zonecfg:s11zone> set ip-type=exclusive
zonecfg:s11zone> add net
zonecfg:s11zone:net> set physical=vnic0
zonecfg:s11zone:net> end
.
.
```

The `ip-type` value for the zone can either be `shared` or `exclusive`:

- The `ip-type=exclusive` value means that you are dedicating a datalink, which can be a virtual (VNIC) for exclusive use by the zone. This strategy brings some of the network stack management benefits to the zone. Historically, this has not been practical if a system has many zones, but only a maximum of 4 network interfaces.

  Exclusive IP is now the recommended *ip-type* for zones. The `set physical` value identifies the system's network interface cards that is assigned to the zone. Using an `ip-type` of exclusive allows the zone to manage its IP stack directly.
- If `ip-type=shared` was identified in the above example, you would have to specify an IP address and other resources.

- **NFS server support in non-global zones** – You can share file systems in a non-global zones by using the NFS protocol. The SMB (CIFS) sharing protocol is not currently available in a non-global zone.
- **Zone monitoring** – System resources that are consumed by non-global zones can be monitored by using the `zonestat` command.
- **Immutable Zones** – The `file-mac-profile` property enables you to run a non-global zone with a read-only root file system. See "zonecfg file-mac-profile Property" in *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.

Oracle Solaris 11.1 Zones Features

The following Oracle Solaris zone enhancements are available in the Oracle Solaris 11.1 release.

- **Parallel zone updates** – A system with multiple Oracle Solaris zones is now updated in parallel. The increase in speed for updating 20 zones is in the 4x range.
- **Improved install and attach performance** – Installing a zone is 27 percent faster and attaching a zone is 91 percent faster. These performance improvements mean that a planned service window of a system with Oracle Solaris zones can be shorter because installing and updating Oracle Solaris zones is much faster.
- **Zone file system statistics** – A per-fstype `kstat` (kernel statistic) for each zone is provided so that you can monitor file system activity in each non-global zone. In addition, a `kstat` is available for monitoring the global zone.

- **Zones on shared storage** – Deployment and migration of Oracle Solaris zones can be simplified by running zones on arbitrary storage objects like Fibre Channel devices or iSCSI targets. You can configure a device path directly with the `zonecfg` command. The zone is automatically encapsulated into its own ZFS storage pool.

See *Oracle Solaris 11.1 Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.

Oracle Solaris 10 Branded Zones Preparation

Prepare for migrating a Oracle Solaris 10 OS instance or zone to your Oracle Solaris 11 system.

- Confirm that your Oracle Solaris 10 instance or zone is running the Oracle Solaris 10 9/10 release, which is the minimum OS requirement.
- Confirm that your Oracle Solaris 10 instance or zone is the same platform as the system migration target. You can only migrate a SPARC instance to a SPARC system and an x86 instance to an x86 based system.
- Download and run the `/usr/sbin/zonep2vchk` script on the Oracle Solaris 10 system to determine if any issues would prevent the Oracle Solaris 10 zone or instance from running successfully on an Oracle Solaris 11 system.

  On an Oracle Solaris 10 1/13 system, the `/usr/sbin/zonep2vchk` utility is included in the release. For a system running an older Oracle Solaris 10 release, download the unbundled package from Oracle Technology Network (OTN):

  `http://www.oracle.com/technetwork/server-storage/solaris10/downloads`

  Keep in mind that this script is only for system migration purposes.
- Enable the Oracle Solaris 10 package and patch tools.

  To use the Oracle Solaris 10 package and patch tools in your Oracle Solaris 10 zones, install the following patches on your source Oracle Solaris 10 system before the image is created.

  - 119254-75, 119534-24, 140914-02 (SPARC platforms)
  - 119255-75, 119535-24 and 140915-02 (x86 platforms)

  The physical to virtual (P2V) process works without the patches, but the package and patch tools do not work properly within the Oracle Solaris 10 zones unless these patches are installed.

Transitioning an Oracle Solaris 10 Instance to an Oracle Solaris 11 System

You can transition your Oracle Solaris 10 environment to a non-global zone on an Oracle Solaris 11 system by creating a zone archive and migrating the zone archive to an Oracle Solaris 11 system. The following steps describe this process.

1. Install the Oracle Solaris 10 zone package on your Oracle Solaris 11 system. For example:

   ```
   s11sysB# pkg install system/zones/brand/brand-solaris10
   ```

2. Run the `zonep2vchk` script to identify any issues that might prevent the instance from running as a `solaris10` zone.

   ```
   s10sys# ./zonep2vchk
   --Executing Version: 1.0.5-11-15652

     - Source System: tardis
         Solaris Version: Oracle Solaris 10 8/11 s10s_u10wos_17b SPARC
         Solaris Kernel:  5.10 Generic_147440-01
         Platform:        sun4u SUNW,Sun-Fire-V440

     - Target System:
         Solaris_Version: Solaris 10
         Zone Brand:      native (default)
         IP type:         shared

   --Executing basic checks
   .
   .
   ```

3. Create a ZFS file system that will contain the flash archive of the Oracle Solaris 10 system instance, if necessary.

   Then, create a NFS share of the ZFS file system on your Oracle Solaris 11 system. For example:

   ```
   s11sysB# zfs create pond/s10archive
   s11sysB# zfs set share=name=s10share,path=/pond/s10archive,prot=nfs,root=s10sysA
   pond/s10archive
   name=s10share,path=/pond/s10archive,prot=nfs,sec=sys,root=s10sysA
   s11sysB# zfs set sharenfs=on pond/s10archive
   ```

4. Select an Oracle Solaris 10 instance, which could be a virtual environment or a global zone on a Solaris 10 system. Note the Oracle Solaris 10 system's `hostid`.

   ```
   s10sysA# hostid
   8439b629
   ```

5. Create an archive of the Oracle Solaris 10 instance that you would like to migrate to a non-global zone on the Oracle Solaris 11 system.

   ```
   s10sysA# flarcreate -S -n s10sysA -L cpio /net/s11sysB/pond/s10archive/s10.flar
   ```

6. Create a ZFS file system for the Oracle Solaris 10 zone.

   ```
   s11sysB# zfs create -o mountpoint=/zones pond/zones
   s11sysB# chmod 700 /zones
   ```

7. Create the non-global zone for the Oracle Solaris 10 instance.

```
s11sysB# zonecfg -z s10zone
s10zone: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:s10zone> create -t SYSsolaris10
zonecfg:s10zone> set zonepath=/zones/s10zone
zonecfg:s10zone> set ip-type=exclusive
zonecfg:s10zone> add anet
zonecfg:s10zone:net> set lower-link=auto
zonecfg:s10zone:net> end
zonecfg:s10zone> set hostid=8439b629
zonecfg:s10zone> verify
zonecfg:s10zone> commit
zonecfg:s10zone> exit
```

8. Install the Oracle Solaris 10 non-global zone.

```
s11sysB# zoneadm -z s10zone install -u -a /pond/s10archive/s10.flar
A ZFS file system has been created for this zone.
Progress being logged to /var/log/zones/zoneadm.20110921T135935Z.s10zone.install
    Installing: This may take several minutes...
 Postprocess: Updating the image to run within a zone
   Postprocess: Migrating data
        from: pond/zones/s10zone/rpool/ROOT/zbe-0
          to: pond/zones/s10zone/rpool/export
.
.
.
```

9. Boot the Oracle Solaris 10 zone.

```
# zoneadm -z s10zone boot
```

10. Configure the Oracle Solaris 10 non-global zone.

```
s11sysB# zlogin -C s10zone
[Connected to zone 's10zone' console]
.
.
.
s10zone console login: root
Password: xxxxxxxx
# cat /etc/release
                  Oracle Solaris 10 8/11 s10s_u10wos_17b SPARC
  Copyright (c) 1983, 2011, Oracle and/or its affiliates. All rights reserved.
                           Assembled 23 August 2011
# uname -a
SunOS supernova 5.10 Generic_Virtual sun4v sparc SUNW,Sun-Fire-T1000
# zfs list
NAME                USED  AVAIL  REFER  MOUNTPOINT
rpool              4.53G  52.2G   106K  /rpool
rpool/ROOT         4.53G  52.2G    31K  legacy
rpool/ROOT/zbe-0   4.53G  52.2G  4.53G  /
rpool/export         63K  52.2G    32K  /export
rpool/export/home    31K  52.2G    31K  /export/home
```

◆ ◆ ◆ CHAPTER 11

Managing User Accounts and User Environments

This chapter provides information about the management of user accounts, groups, roles, and a user's environment in Oracle Solaris 11 releases.

The following topics are covered:

Commands and Tools for Managing User Accounts

Note – The Solaris Management Console graphical tool and its associated command line interface have been removed. To create and manage user accounts, use the command-line and graphical tools that are described or referenced in this chapter.

TABLE 11–1 Commands and Tools for Managing User Accounts

Command/Tool Name	Description	For More Information
`useradd`, `groupadd`, `roleadd`	Commands for adding users, groups, and roles.	Managing User Accounts "How to Create a Role" in *Oracle Solaris 11.1 Administration: Security Services*
`usermod`, `groupmod`, `rolemod`	Commands for modifying users, groups, and roles.	*Oracle Solaris 11.1 Administration: Security Services*

TABLE 11–1 Commands and Tools for Managing User Accounts *(Continued)*

Command/Tool Name	Description	For More Information
`userdel`, `groupdel`, `roledel`	Commands for deleting users, groups, and roles.	"How to Delete a User" in *Managing User Accounts and User Environments in Oracle Solaris 11.1* and `userdel`(1M) `groupdel`(1M), `roledel`(1M)
User Manager GUI	GUI for creating and managing users.	Chapter 3, "Managing User Accounts by Using the User Manager GUI (Tasks)," in *Managing User Accounts and User Environments in Oracle Solaris 11.1*

Managing User Accounts

In this release, you can create and manage user accounts from the command line or with the User Manager GUI. The GUI replaces some of the functionality of the Solaris Management Console and its associated command-line. For more information, see *Managing User Accounts and User Environments in Oracle Solaris 11.1*.

User Account Management Changes

The following features are new or have changed in this release:

- **Addition of the User Manager GUI** – The User Manager GUI is part of the Visual Panels project and is accessible from the desktop. The GUI replaces some of the functionality of the Solaris Management Console. See Chapter 3, "Managing User Accounts by Using the User Manager GUI (Tasks)," in *Managing User Accounts and User Environments in Oracle Solaris 11.1*.
- **Creating user accounts** – User account creation has changed in the following ways:
  - In Oracle Solaris 11, user accounts are created as individual ZFS file systems, which enables users to have their own file system and their own ZFS dataset. Every home directory that is created with the `useradd` and `roleadd` commands places the user's home directory on `/export/home` as an *individual* ZFS file system.
  - The `useradd` command relies on the automount service, `svc:/system/filesystem/autofs`, to mount home directories. This service should never be disabled. Each home directory entry for a user in the `passwd` database uses the format, `/home/`*username*, which is an `autofs` trigger that is resolved by the automounter through the `auto_home` map.

  - The `useradd` command automatically creates entries in the `auto_home` map that correspond to the pathname that is specified by the `-d` option of this command. If the pathname includes a remote host specification, for example, `foobar:/export/home/jdoe`, then the home directory for `jdoe` must be created on the system `foobar`. The default pathname is `localhost:/export/home/`*user*. Because `/export/home` is the mount point for a ZFS dataset, the user's home directory is created as a child ZFS dataset, with the ZFS permission to take snapshots delegated to the user. If a pathname that does not correspond to a ZFS dataset is specified, then a regular directory is created. If the `-S ldap` option is specified, then the `auto_home` map entry is updated on the LDAP server instead of the local `auto_home` map.

- **Modifying user accounts** – In Oracle Solaris 11, the `usermod` command works with LDAP and files. All security attributes can be assigned to a user by using this mechanism. For example, an administrator can add a role to a user's account by using the `usermod` command.

  ```
  # roleadd -K roleauth=user -P "Network Management" netmgt
  # usermod -R +netmgt jdoe
  ```

  See `usermod(1M)` for additional examples.

- **Creating and managing groups** – An administrator who has the `solaris.group.manage` authorization can create a group. At group creation, the system assigns the `solaris.group.assign/`*groupname* authorization to the administrator, which gives the administrator complete control over that group. The administrator can then modify or delete that *groupname*, as needed. See the `groupadd(1M)` and `groupmod(1M)` man pages.
- **Creating and managing roles** – Roles can be created locally and in an LDAP repository. To create a role and assign an initial password, you must be assigned the `User Management` rights profile. For instructions on creating a role, see "How to Create a Role" in *Oracle Solaris 11.1 Administration: Security Services*.

User Password and Login Changes

User password management and login information have changed in the following ways:

- **Property definition refinements for the `password` command** – This change clarifies which user accounts can and cannot be locked. The primary changes impact the `LK` and `NL` property definitions, as follows:

  `LK` The account is locked for UNIX authentication. The `passwd -l` command was run, or the account was automatically locked due to the number of authentication failures reaching the configured maximum that is allowed. See the `policy.conf(4)` and `user_attr(4)` man pages.

  `NL` The account is a `no login` account. The `passwd -N` command was run.

- **Failed login count notification** – The system now notifies users of failed authentication attempts, even if the user account is not configured to enforce failed logins. Users who fail to authenticate correctly, will see a message similar to following upon successful authentication:

  ```
  Warning: 2 failed authentication attempts since last successful
  authentication. The latest at Thu May 24 12:02 2012.
  ```

  To suppress such notifications, create a `~/.hushlogin` file.

- **Default password hashing algorithm** – The default password hashing algorithm is now SHA256. This password hash is similar to the following:

  ```
  $5$cgQk2iUy$AhHtVGx5Qd0.W3NCKjikb8.KhOiA4DpxsW55sP0UnYD
  ```

  Also, there is no longer an eight character limitation for user passwords. The eight character limitation only applies to passwords that use the older `crypt_unix`(5) algorithm, which has been preserved for backwards compatibility with any existing `passwd` file entries and NIS maps. Starting with Oracle Solaris 11, the `crypt_sha256` algorithm is the default

  Passwords are encoded by using one of the other `crypt`(3c) algorithms, including the SHA256 algorithm, which is the default in the `policy.conf` file. Thus, passwords can now be much longer than eight characters. See `policy.conf(4)`.

Sharing Home Directories That Are Created as ZFS File Systems

An NFS or a SMB share of a ZFS file system is created and then shared as follows:

- **Oracle Solaris 11:** The file system share is created by using the `zfs set share` command. At this time, specific share properties can be defined. If share properties are not defined, the default property values are used.

  The NFS or SMB share is published by setting the `sharenfs` or `sharesmb` property. The share is published permanently until the property is set to `off`.
- **Oracle Solaris 11.1:** The following sharing features are provided on ZFS storage pool version 34:
  - The `share.nfs` property replaces the `sharenfs` property in previous releases to define and publish an NFS share.
  - The `share.smb` property replaces the `sharesmb` property in previous releases to define and publish an SMB share.
  - ZFS share administration is simplified by leveraging ZFS property inheritance. If you want to share the `tank/home` file system, use syntax similar to the following:

    ```
    # zfs set share.nfs=on tank/home
    ```

    The `share.nfs` property value is inherited to any descendent file systems.

    ```
    # zfs create tank/home/userA
    # zfs create tank/home/userB
    ```

See "How to Share Home Directories That Are Created as ZFS File Systems" in *Managing User Accounts and User Environments in Oracle Solaris 11.1*.

How Home Directories Are Mounted in Oracle Solaris

Because home directories are created as ZFS file systems in Oracle Solaris 11, you typically do not need to manually mount home directories. The home directory is automatically mounted during its creation and also at boot time from the SMF local file system service. For instructions on manually mounting a user's home directory, see "Manually Mounting a User's Home Directory " in *Managing User Accounts and User Environments in Oracle Solaris 11.1*.

User Environment Feature Changes

Oracle Solaris 11 includes the following user environment and command-line argument changes:

- **Addition of** `/var/user/$USER` – Starting with Oracle Solaris 11.1, whenever a user logs in and successfully authenticates by using the `pam_unix_cred` module, a `/var/user/$USER` directory is explicitly created, if the directory does not already exist. This directory enables applications to store persistent data that is associated with a particular user on the host system. The `/var/user/$USER` directory is created upon initial credential establishment, as well during a secondary authentication when changing users by using the `su`, `ssh`, `rlogin`, and `telnet` commands. The `/var/user/$USER` directory does not require any administration. However, users should be aware of how the directory is created, its function, and that it is visible in the `/var` directory.
- **Shell changes** - The default shell, `/bin/sh`, is now linked to `ksh93`. The default user shell is the Bourne-again (`bash`) shell.
  - The legacy Bourne shell is available as `/usr/sunos/bin/sh`.
  - The legacy `ksh88` is available as `/usr/sunos/bin/ksh` from the `shell/ksh88` package.
  - Korn shell compatibility information is available in `/usr/share/doc/ksh/COMPATIBILITY`.
- **Default path** – The default user path is `/usr/bin`. The default path for the root role is `/usr/bin:/usr/sbin`.
- **Command locations** – Administration commands that were previously in `/sbin` have moved to `/usr/sbin`. In addition, the `/sbin` directory has been replaced by an `/sbin` → `/usr/sbin` symbolic link.
- `MANPATH` **variable** – The `MANPATH` environment variable is no longer required.

  The `man` command determines the appropriate `MANPATH`, based on the `PATH` environment variable setting.
- **Developer tools locations** – Developer tools that were previously in `/usr/ccs/bin` have moved to `/usr/bin`. The `/usr/ccs/bin` directory is replaced by a `/usr/ccs/bin` → `/usr/bin` symbolic link.
- **File locations** – Files that were previously in the `/usr/sfw` directory are now in `/usr/bin`.
- **Editor changes** – The `vi` family of editors, including `/usr/bin/vi`, `/usr/bin/view`, and `/usr/bin/ex`, are now links to the `vim` open source implementation of the `vi` editor. The traditional SunOS versions of these commands are available in `/usr/sunos/bin/`.
- **Java version** – Java 7 is the default Java version in this release. Java 7 includes several feature, security, and performance enhancements for Oracle Solaris, including the new OracleUcrypto Provider, which on SPARC T4 platforms directly accesses the underlying

native (on-chip) T4 crypto capabilities for maximum performance while minimizing CPU load. For more details, go to http://www.oracle.com/technetwork/java/javase/compatibility-417013.html.

Change the default version to Java 7, as follows:

```
# pkg set-mediator -V 1.7 java
```

Default Login Shell and PATH Environment Variable

In Oracle Solaris 10, the default scripting shell (/bin/sh) is the Bourne shell. In Oracle Solaris 11, /bin/sh is the Korn shell (ksh93), and the default interactive shell is the Bourne-again (bash) shell. When used as a login shell, bash retrieves configuration information from the first instance of .bash_profile, .bash_login, or .profile file.

The default PATH environment variable for bash is:

```
/usr/bin:/usr/sbin
```

Oracle Solaris Man Page Changes

The following man page features are new or have changed:

- **Locating information in man pages** – This release has the capability of searching man pages with query strings by using the man -K *keywords* command. The -K (uppercase) option works similarly to the -k (lowercase) option, with the exception that the -k option is limited to searching only the NAME subsection of all of the man page sections.

  The -k and -K options uses index files for searching. A new SMF service, svc:/application/man-index:default, triggers the automatic regeneration of new index files whenever new man pages are added to the /usr/share/man and /usr/gnu/share/man directories, if these directories exist. This service is enabled by default.
- **Package Name Change** – The SUNWman package that contained the Oracle Solaris man pages in previous releases has changed to the smaller system/manual package. The bulk of the man pages are now packaged separately with their component technology packages. For example, ls.1m for the /usr/bin/ls command is part of the system/core-os package.
- **Man Page Display** – If the man pages are not displaying on your system, you can use the following command to toggle whether man pages are installed on the system:

  ```
  # pkg change-facet facet.doc.man=true
  ```

  Note – Be aware that running the previous command downloads several files to local disk, which effectively is the reverse of the command that you run to remove all man pages.

◆ ◆ ◆ CHAPTER 12

Managing Desktop Features

This chapter describes the desktop features that are supported in Oracle Solaris 11 releases.

The following topics are covered:

Oracle Solaris Desktop Feature Summary

The default desktop environment in Oracle Solaris 11is the Oracle Solaris Desktop, which includes GNOME 2.30 from the GNOME Foundation. Also included is the Firefox web browser, Thunderbird Email client, and the Lightning calendar manager from the Mozilla Foundation.

Note – If you use the text installation method, the Oracle Solaris Desktop package (`solaris-desktop`) is not installed on your system by default. Also, the `solaris-desktop` package cannot be applied directly to a running system. For more information, see "Installing the Oracle Solaris Desktop Software Package After an Installation" on page 168.

Other new desktop features include the following:

- Accessibility feature enhancements
- Bluefish HTML editor
- Compiz OpenGL based window manager
- D-Bus IPC Framework
- Evince PDF viewer

- GIMP image editing program
- GNOME Python bindings
- Gobby text editing collaboration tool
- Multimedia support enhancements
- Planner and `openproj` project management tools
- Trusted Extensions integration
- `xchat` IRC client
- Xserver features that augment the desktop, such as virtual terminal (VT) switching

Key Desktop Features

The following key features are new or enhanced in Oracle Solaris 11:

- **Accessibility enhancements** – Users with disabilities can use a wide range of accessibility features, including Orca, `espeak`, and `brltty`. These features replace `gnopernicus` and provide better text-to-speech support. The dasher on-screen keyboard also has been added in this release.

  Note that the GNOME On-screen Keyboard (GOK) program that is used Oracle Solaris 10 is no longer available. The new dasher application can be used as a replacement for some users.
- **Command Assistant** – Locates command-line information in Oracle Solaris managed content, for example books and man pages. To add Command Assistant to the desktop panel, use the Add to Panel → Command Assistant dialog box.
- **Graphical login manager** – Oracle Solaris 10 uses the Common Desktop Environment (CDE) and `dtlogin` as the default login GUI. The GNOME graphical desktop manager (GDM) is also available in Oracle Solaris 10. In this release, GDM is the only graphical login option.

  The GDM configuration process has also changed considerably in Oracle Solaris 11. To learn more, consult the `gdm` and `console-kit-daemon` man pages. ConsoleKit configuration features are now used to manage multi-seat environments. To troubleshoot transition issues, see "GNOME Desktop Manager Issues" on page 169.
- **Multimedia support:**
  - **FreeDesktop GStreamer** – The FreeDesktop GStreamer module is a desktop tool that provides multimedia support. GStreamer uses a plug-in infrastructure that enables the use of additional media formats.
  - `gksu` – Is the graphical version of the `sudo` command. When launched, the tool displays a prompt that enables you to type an additional password to run an administrative tool.

- **Multimedia formats** – The FLAC, Speex, Ogg Vorbis, and Theora media formats are supported through the use of GStreamer plugins. Oracle Solaris 11 provides GStreamer 0.10, while Oracle Solaris 10 uses GStreamer 0.8.
- **Open Sound System** – The Open Sound System (OSS) framework manages audio devices and provides better audio support. Some audio devices that were previously supported are no longer supported. Programs that use the Sun Audio Device Architecture (SADA) interfaces continue to be supported. If your audio device is not working properly, you can launch a dialog box from the desktop that enables you to choose which audio device and GStreamer audio input/output plugins to use:

  ```
  $ /usr/bin/gstreamer-properties
  ```

  This program also includes a Test button that enables you to determine whether your audio settings are correct. Note that some audio cards present themselves as having more than one device, for example, one for analog audio and one for digital audio. If you are currently using RealPlayer, you will need to transition to the multimedia tools that are currently supported.
- **PulseAudio sound server** – Introduced in Oracle Solaris 11.1, the PulseAudio sound server supports improved audio mixing. The `/usr/bin/gnome-volume-control` Device combo-box displays additional PulseAudio devices. For desktop and laptop computers, the "OSS" device choice should work best. To determine the best setting for your audio hardware, some initial trial-and-error might be required. If you continue to experience audio problems, run the following command to verify that the correct default Input/Output audio plug-ins are selected:

  ```
  $ /usr/bin/gstreamer-properties
  ```

  PulseAudio additionally provides CLI configuration capabilities: `$HOME/.pulse`, and `$HOME/.pulse-cookie`. See `pulseaudio`(1) for details. On systems with a working audio card, you will notice that the `/usr/bin/pulseaudio` process is running for GNOME sessions. More information can be found at `http://www.freedesktop.org/wiki/Software/PulseAudio`.
- **Other media tools** – The Rhythmbox media player, Cheese photo/video tool, the Ekiga video conference tool, and the Brasero CD/DVD burner are included in this release.

- **Network configuration management** – The network administration GUI (formerly NWAM) is used to manage network configuration from the desktop. This tool functions similarly to the various networking command-line tools. See "Managing Network Configuration From the Desktop" on page 105..
- **Package Manager and Update Manager** – Are the graphical versions of the IPS command-line tools. Package Manager and Update Manager can be used to manage and update software packages from the desktop. See Chapter 2, "IPS Graphical User Interfaces," in *Adding and Updating Oracle Solaris 11.1 Software Packages* for instructions on using these tools.

- **Print management** – CUPS is the default print service in Oracle Solaris 11, replacing the LP print service. Solaris Print Manager is no longer available. CUPS has a print manager that can be started from the desktop by choosing System → Administration → Print Manager. See "Setting Up Printers by Using CUPS Print Manager" in *Oracle Solaris Administration: Common Tasks*.
- **Removable media** – Oracle Solaris 11 includes various removable media enhancements, including support for hot-pluggable device discovery, content recognition, usability, security, and performance across all layers of the software stack, from device drivers to the GUI. You can use the Eject button on a CD/DVD drive's front panel to eject a disc, even if it is mounted. The Nautilus file manager automatically registers when external hard drives or flash cards are inserted.

  The functions of the `vold` daemon and the `volcheck` command are now performed by the Hardware Abstraction Layer (HAL) through the `rmvolmgr` and `gvfs-hal-volume-monitor` commands, which are HAL-aware. See `rmvolmgr(1M)`.
- **Seahorse** – GnuPG is now supported. The Seahorse application manages encryption keys and passwords in the `gnome-keyring`. Seahorse also replaces the `gnome-keyring-manager` for managing SSH and GnuPG keys.
- **Trusted Extensions (GNOME) desktop** – The Trusted Extensions feature of Oracle Solaris is now supported only in the Oracle Solaris Desktop (GNOME 2.30). In Oracle Solaris 10, this feature is supported in both CDE and the GNOME Desktop. In Solaris 8, this support is limited to CDE.

  This version of the Trusted Extensions desktop includes significant changes that improve usability, robustness, and functionality, which also includes zones and RBAC improvements. For example, the `txzonemgr` GUI has significantly improved. This tool can now be used to manage most aspects of Trusted Extensions. If you are currently using Trusted CDE, you will need to migrate to the a currently supported version of the product.

  For up-to-date information about Sun Ray software support for the Oracle Solaris Desktop and Trusted Extensions, go to `http://www.oracle.com/technetwork/server-storage/sunrayproducts/overview/index.html`.
- **Time Slider** – Manages ZFS snapshots. The tool can be used to regularly back up data by taking timed ZFS snapshots.
- **Virtual console terminals** – You can now switch between an X session and a virtual console terminal. This service is enabled by default. To switch between sessions, use the Alt + Ctrl + F# hotkey combination. For example, to switch to vt2, press Alt + Ctrl + F2. Also, you can create graphical VT sessions and then switch between those sessions by using the User Switcher panel applet. To add the applet to the desktop, right click the panel, then select the Add to Panel... option. To switch to a new or different graphical login session, click the applet, then select Switch User.
- **Web browser and Email** – Oracle Solaris 11 includes the new Firefox and Thunderbird applications.

Desktop Features That Have Been Removed

The following desktop features have been replaced or removed. Note that some features were introduced later than Oracle Solaris 10:

- Adobe Flash Player – This feature was present in Oracle Solaris 11, but removed in Oracle Solaris 11.1. You can download older versions from Adobe's web site, but Adobe no longer produces or supports Flash for Oracle Solaris.
- Common Desktop Environment (CDE) – CDE is replaced by the Oracle Solaris Desktop (GNOME 2.30).
- ESounD – Migrate to the GStreamer programs, such as `gst-launch`.
- `gnome-keyring-manager` – Seahorse replaces this feature.
- GNOME On-screen Keyboard (GOK) program – The dasher application can be used as a replacement in some instances.
- GNOME System tools (Introduced in an earlier Oracle Solaris 11 release):
  - `network-admin` – NWAM replaces this feature.
  - `services-admin` – Use the `/usr/bin/vp svcs` command.
  - `shares-admin` – Use the `/usr/bin/vp sharemgr` command.
  - `time-admin` – Use the `/usr/bin/vp time` command.
  - `users-admin` (GNOME Users and Groups tool) – No replacement is currently available. See "Commands and Tools for Managing User Accounts" on page 155.

    The GNOME System tools are not available in Oracle Solaris 10.
- Solaris Management Console – This tool and its equivalent command-line are no longer available. The User Manager GUI replaces this tool in Oracle Solaris 11.1. See "Commands and Tools for Managing User Accounts" on page 155.
- Solaris Print Manager – This tool is replaced by CUPS Print Manager. See "Printer Configuration and Management Changes" on page 129.
- `Xsun` family of servers – The `Xorg` family of servers is still supported. See "Xorg Family of Servers" on page 167.

Xorg Family of Servers

While Oracle Solaris 10 includes both the `Xsun` family of X servers, with `Xsun` as the default on SPARC platforms, and `Xorg` as the default on x86 platforms, Oracle Solaris 11 only supports the `Xorg` family of servers. X server information has moved from `/usr/X11/bin` to `/usr/bin`. Note that `Xorg` packages are included on the Live Media, but not with the text installer. The following table lists legacy Oracle Solaris X server commands and the corresponding Oracle Solaris 11 commands.

TABLE 12–1 Oracle Solaris 11 X Server Commands

Legacy Command	Oracle Solaris 11 Command
`/usr/openwin/bin/Xsun`	`/usr/bin/Xorg`
`/usr/openwin/bin/Xnest`	`/usr/bin/Xephyr`
`/usr/openwin/bin/Xvfb`	`/usr/bin/Xvfb`

X Server Key Maps

Oracle Solaris 11 has moved to more common `Xorg` key mappings. For example, the Copy key is now mapped to `XF86Copy`.

▼ How to Update Custom Hot Key Configurations or Enable Legacy Mappings

1. **To update custom hot key configurations or enable the legacy mappings from the desktop, open the Keyboard panel from the System → Preferences menu.**
2. **Select the Layouts tab, then click the Options... button to open the Keyboard Layout Options dialog box.**
3. **Select the Maintain key compatibility with old Solaris keycodes option, then select the Sun Key Compatibility checkbox.**

Troubleshooting Desktop Transition Issues

Refer to the following troubleshooting information when transitioning to the Oracle Solaris Desktop (GNOME 2.30).

Installing the Oracle Solaris Desktop Software Package After an Installation

The Oracle Solaris 11 text installer does not include the primary software package that includes the GNOME 2.30 desktop. If you use this installation method, you will need to install the `solaris-desktop` package afterwards. For information about using the `pkg install` command to add packages after a text installation, see "Adding Software After Text Installation" in *Installing Oracle Solaris 11.1 Systems*.

If you have a situation where you need to install the `solaris-desktop` package on a system running a live session, create a new boot environment, install the `solaris-desktop` package, then activate the new boot environment, as follows:

```
# beadm create be-name
# beadm mount be-name /mnt
# pkg -R /mnt install group/system/solaris-desktop
# bootadm update-archive -R /mnt
# beadm umount be-name
# beadm activate be-name
```

GNOME Desktop Manager Issues

Note the following potential GDM login issues:

- **CDE to GDM login configuration** – If you customized your CDE login in Oracle Solaris 10, you will likely need to reintegrate your configuration choices to work with GDM in Oracle Solaris 11. Note that an exact one-to-one mapping between CDE and GDM login features does not exist. Some CDE login configuration choices are not available in the GDM login, and there are some GDM login configuration choices that are not available in the CDE login. For example, the GDM login screen does not offer a chooser screen by default.

  Another example is the X Display Manager Control Protocol (XDMCP) feature, which is configured and enabled differently in Oracle Solaris 11 than Oracle Solaris 10. The new GDM provides the ability to run an XDMCP server, but this feature is disabled by default. You can enable the feature by modifying the GDM configuration file.

  Another requirement of XDMCP is that X11 allow TCP/IP connections, which is also disabled by default. Refer to the `Xserver`(1) man page for instructions on how to enable this feature. See also the `gdm`(1) man page, the yelp-tools manual, and the online help.
- **Support for Oracle Solaris 10 GDM themes in Oracle Solaris** – In Oracle Solaris 10, GDM is shipped as a non-default login program, which includes a GUI configuration tool. In Oracle Solaris 11, GDM does *not* have this GUI configuration tool. Also, the GDM *themes* that work with GDM in Oracle Solaris 10 are not supported in this release. You can change the appearance of the new GDM login GUI by modifying the `/usr/share/gdm/gdm-greeter-login-window.ui` file, as desired.

◆ ◆ ◆ APPENDIX A

SPARC Automated Installation Scenario

This appendix includes an end-to-end example of installing a SPARC based system with Oracle Solaris 11.1 by using the automated installation method.

The following topics are covered:

Installing a System by Using AI

There are many different ways to configure an AI server and perform automatic installations. This section provides a minimal AI installation example.

- **Minimum installation server configuration**
  - Install Oracle Solaris 11.1
  - Configure a static IP address, set a default router, and enable multicast DNS
  - Download an Oracle Solaris 11.1 IPS image
  - Confirm `install/installadm` package is available
  - Create an install service
  - Associate any clients with the install service
  - Provide access to a package repository server – Client systems install software from a package server so they need to access to the release repository (`http://pkg.oracle.com/solaris/release`), the My Oracle Support (MOS) repository (`https://pkg.oracle.com/solaris/support/`) or a local package repository. If you use a local package repository, then you will need to customize a manifest to include the local repository server.

- DHCP and DNS information – When you need to install many client systems, providing access to DHCP and DNS info is a best practice. It is not required, however, if you want to install SPARC client systems individually.

- **No DHCP service** – You can install a SPARC client system without a DHCP server, but an AI installation does not support RARP. This means the SPARC client must specify the network boot arguments at the PROM level. See "Booting the Installation Client" on page 178 for a SPARC client boot example.

  In addition, if a local package repository is not available, the SPARC client installation fails when attempting to resolve the `pkg.oracle.com` address without DHCP. The best resolution is to provide a local package repository or use DHCP. The x86 text-based installation installs without access to a package repository.

- **Additional installation configuration considerations** – There are many configuration options, but the following example provides a local package repository, accessible DHCP and DNS information, and a minimally customized manifest.

  A default AI manifest is customized to add a local repository.

  - Installation client selection criteria – You can include criteria keywords that identify specific client configuration information. This information is provided to the install service. Selection criteria is somewhat similar to a JumpStart `rules` file. Installation selection criteria is not included in the example that follows.
  - AI manifests – You can customize an existing default manifest to install from a local package repository or to install a specific package group, or to modify target disks or file systems. An AI manifest is somewhat similar to a JumpStart `profile` file. See "Configure an AI Manifest" on page 176.
  - System configuration profiles – You can create a customized system profile to configure a specific root password, user account, or keyboard layout. A profile is somewhat similar to the output that is generated by the previous `sysidcfg` tool. A system configuration profile is now generated by the `sysconfig` tool. A customized profile is not included in the example that follows.

For more information about customizing manifests and profiles, see *Installing Oracle Solaris 11.1 Systems*.

Configuring an AI Server

The following example illustrates the primary tasks of configuring an AI server to install a SPARC based system on an IPv4 network with a local package repository, and a minimally customized AI manifest.

This example uses the following configuration options:

- DNS is configured for name resolution
- The AI server provides a local package repository

- DHCP is configured for the install service
- The install client is a SPARC based system

Confirm Network Related Resources Are Available

Configuration steps covered in this task:

- Confirm the installation server's static IP and router information.
- Identify the DHCP range of IPs addresses to install, if necessary.

  When you create the install service, you need to specify the starting address of the IP address range and the number of addresses.
- You can specify fixed IP addresses for client systems in the following ways:
  - Generate a system configuration profile with the `sysconfig` tool or customize a profile in `/usr/share/auto_install/sc_profiles` and add it to the install service.

    ```
    # sysconfig create-profile -o /var/tmp/manifests/client_sc.xml
    ```
  - Include fixed IP addresses in the `/etc/inet/dhcpd.conf` file. See Example A–3.
  - Create a customized script that associates client systems' MAC addresses with a specific IP address and install server.
- If you want to use DNS for name resolution, make sure the DNS server is accessible.

EXAMPLE A–1 How to Confirm Network Related Resources Are Available

Confirm the install server has a static IP and router information.

```
# ipadm show-addr
ADDROBJ           TYPE     STATE        ADDR
lo0/v4            static   ok           127.0.0.1/8
e1000g0/v4static1 static   ok           1.2.3.10/24
lo0/v6            static   ok           ::1/128
e1000g0/v6dhcp    addrconf ok           fe80::aaa:bbbb:cccc:8988/10

# netstat -nr
Routing Table: IPv4
  Destination           Gateway           Flags  Ref     Use     Interface
-------------------- -------------------- ----- ----- ---------- ---------
default              1.2.3.1              UG        3     115957 net0
1.2.3.64             1.2.3.10             U         4     287300 net0
127.0.0.1            127.0.0.1            UH        2        116 lo0
.
.
.
```

Make sure the DNS server is accessible.

```
# getent hosts daleks
1.2.3.99        daleks
```

Create a Local Package Repository

Configuration steps covered in this task:

- Download a repository image from `pkg.oracle.com` and mount the image
- Create a file system for the repository and copy the image into the repository file system
- Refresh the repository
- Enable the package repository service so that it can be available from an `http:` location

EXAMPLE A–2 How to Create a Local Package Repository

These steps are performed on the install server or on another system that is accessible over the network.

Copy the repository image by using your favorite tool, such as `rsync` or `tar`. Using `tar` is generally faster than `rsync`, but the `tar` operation might take a while.

```
# mount -F hsfs /tmp/sol-11_1-repo-full.iso /mnt
# zfs create rpool/export/s11.1repo
# cd /mnt/repo; tar cf - . | (cd /export/s11.1repo; tar xfp - )
# pkgrepo -s /export/s11.1repo refresh
Initiating repository refresh.
```

Next, start the package service so that the package repository is accessible.

```
# svccfg -s application/pkg/server setprop pkg/inst_root=/export/s11.1repo
# svccfg -s application/pkg/server setprop pkg/readonly=true
# svcadm refresh application/pkg/server
# svcadm enable application/pkg/server
# pkg set-publisher -G "*" -g http://tardis.dw.com/ solaris
```

Create an AI Install Service

Configuration steps covered in this task:

- Confirm that the `installadm` package is installed
- Create a file system for the image that is created from the local package repository
- Start the multicast DNS service
- Create the install service
- Add the client information to the install service

EXAMPLE A–3 How to Create an Install Service

The following steps are performed on the install server to create the install service. In Oracle Solaris 11.1, the install service image can be created directly from a local package repository.

Confirm that the `install/installadm` package is available.

EXAMPLE A–3 How to Create an Install Service *(Continued)*

```
# pkg info installadm
          Name: install/installadm
       Summary: installadm utility
   Description: Automatic Installation Server Setup Tools
      Category: System/Administration and Configuration
         State: Installed
     Publisher: solaris
       Version: 0.5.11
 Build Release: 5.11
        Branch: 0.175.1.0.0.24.1736
Packaging Date: Wed Sep 12 19:32:53 2012
          Size: 1.23 MB
          FMRI: pkg://solaris/install/installadm@0.5.11,5.11-0.175.1.0.0.24.1736:
20120912T193253Z
```

If not installed, install it:

```
# pkg install install/installadm
```

Create a file system for the image and enable the multicast DNS service. Then, create the install service. In this example, the starting DHCP address is specified with the `-i` option and the `-c` option identifies the number of addresses. This syntax creates an ISC DHCP server on the install server if it does not already exist.

```
# zfs create rpool/export/image
# svcadm enable svc:/network/dns/multicast:default
# installadm create-service -n sol-11u1-sparc -d /export/image -i 1.2.3.66
-c 20
Creating service from: Creating service from: pkg:/install-image/solaris-auto-install
Setting up the image ...

DOWNLOAD                                  PKGS         FILES    XFER (MB)   SPEED
Completed                                  1/1         45/45  237.8/237.8    0B/s

PHASE                                        ITEMS
Installing new actions                     187/187
Updating package state database               Done
Updating image state                          Done
Creating fast lookup database                 Done
Reading search index                          Done
Updating search index                          1/1

Creating sparc service: sol-11u1-sparc

Image path: /export/image

Service discovery fallback mechanism set up
Creating SPARC configuration file
Starting DHCP server...
Adding IP range to local DHCP configuration
Creating default-sparc alias
Service discovery fallback mechanism set up

Refreshing install services
```

EXAMPLE A–3 How to Create an Install Service *(Continued)*

The DCHP configuration information is included in the `/etc/inet/dhcpd4.conf` file. The range of dynamic IP addresses looks similar to this:

```
range 1.2.3.66 1.2.3.86
```

You can also identify fixed addresses for clients by including syntax similar to the following in the `/etc/inet/dhcpd4.conf` file:

```
host neo {
   hardware ethernet 0:3:3:4:5:2 ;
   fixed-address 1.2.3.88 ;
}
```

Add the client's MAC address to the install service:

```
# installadm create-client -e 0:3:ba:dd:ff:2 -n sol-11u1-sparc
```

If you do not specify the DHCP information when the install service is created, the SPARC client boot information is provided on the screen:

```
# installadm create-service -n sol-11u1-sparc -d /export/image
Creating service from: Creating service from: pkg:/install-image/solaris-auto-install
Setting up the image ...
DOWNLOAD                                PKGS         FILES    XFER (MB)   SPEED
Completed                                1/1         45/45  237.8/237.8    0B/s

PHASE                                          ITEMS
Installing new actions                       187/187
Updating package state database                 Done
.
.
.
Creating sparc service: sol-11u1-sparc

Image path: /export/image

Creating default-sparc alias

Service discovery fallback mechanism set up
Creating SPARC configuration file
No local DHCP configuration found. This service is the default
alias for all SPARC clients. If not already in place, the following should
be added to the DHCP configuration:
Boot file: http://1.2.3.10:5555/cgi-bin/wanboot-cgi
```

Configure an AI Manifest

Configuration steps covered in this task:

- Create an optional file system for the AI manifest
- Export a default manifest template copy that is called `sclient.xml`

- Modify the `sclient.xml` manifest to match your installation needs
- Create the manifest and associate it with the install service
- Display the install service and manifest to ensure that it is available
- Activate the manifest

EXAMPLE A–4 How to Configure an AI Manifest

The following example describes how to create a file system for manifests, modify a default manifest, and associate it with the install service.

```
# zfs create rpool/export/manifests
# cd /export/manifests
# installadm export -n sol-11u1-sparc -m orig_default -o sclient.xml
# vi sclient.xml
# installadm create-manifest -n sol-11u1-sparc -f ./sclient.xml -m sclient
# installadm list -n sol-11u1-sparc -m
# installadm set-service -o default-manifest=sclient sol-11u1-sparc
# installadm list -n sol-11u1-sparc -m
```

The `sclient` manifest is modified as follows:

- Add the `auto_reboot` keyword so that the client is rebooted automatically.
- Include a local package repository (`tardis.dw.com`) instead of `pkg.oracle.com`.

The `sclient` manifest changes are included in **bold** as follows:

```
.
.
.
<ai_instance name="orig_default" auto_reboot="true">
.
.
.
<source>
  <publisher name="solaris">
    <origin name="http://tardis.dw.com"/>
  </publisher>
</source>
.
.
.
```

If you change the manifest after it has been activated, update it so the changes take affect.

```
# installadm update-manifest -n sol-11u1-sparc -f ./sclient.xml -m sclient
```

Booting the Installation Client

The following examples show how to boot the SPARC client with or without DHCP and how to monitor the installation process. A brief description of the client configuration after an installation is also provided.

EXAMPLE A–5 How to Boot the Installation Client

If DHCP was configured, boot the SPARC client as follows.

```
ok boot net:dhcp - install
```

If DHCP was not configured, boot the SPARC client with the `network-boot-arguments` syntax. Be sure to supply the install server information from the output of the `installadm create-service` command.

In the following example, `host-ip=1.2.3.88` identifies the client's IP address, `router-ip=1.2.3.1` is the IP address of the router, `hostname=neo` identifies the client's hostname, and the `http: wanboot` string address includes the AI server's system name and IP address.

```
ok setenv network-boot-arguments host-ip=1.2.3.88,router-ip=1.2.3.1,
subnet-mask=255.255.255.0,hostname=neo,file=http://1.2.3.10:5555/cgi-bin/
wanboot-cgi
network-boot-arguments =  host-ip=1.2.3.88,router-ip=1.2.3.1,
subnet-mask=255.255.255.0,hostname=neo,file=http://1.2.3.10:5555/
cgi-bin/wanboot-cgi
ok boot net - install
```

The network boot arguments are echoed to the screen and any errors are displayed automatically.

After the client boots and the installation process begins successfully, you can log in to the client system to monitor the installation process.

```
Automated Installation started
The progress of the Automated Installation will be output to the console
Detailed logging is in the logfile at /system/volatile/install_log
Press RETURN to get a login prompt at any time.
solaris login: root
password: solaris
# tail -f /system/volatile/install_log
```

After the installation, the log file is available here:

```
# more /var/log/install/install_log
```

If client configuration information is not provided by using DHCP, DNS, or through a customized profile or criteria, you are prompted to provide the system's configuration information, such as network, host name, time zone, and root password, after the client is first booted.

EXAMPLE A–5 How to Boot the Installation Client *(Continued)*

This information is prompted by the `/usr/share/auto_install/sc_profiles/enable_sci.xml` profile found on an installed system.

A message similar to the following identifies where the system information is stored on the local client system:

```
SC profile successfully generated.
Exiting System Configuration Tool. Log is available at:
/system/volatile/sysconfig/sysconfig.log.553
```