

Transitioning From Oracle® Solaris 10 to Oracle Solaris 11.2

ORACLE®

Part No: E39134-03
December 2014

Copyright © 2011, 2014, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Copyright © 2011, 2014, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf disposition de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée d'The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation.

Contents

- Using This Documentation** 9

- 1 About Transitioning From Oracle Solaris 10 to an Oracle Solaris 11 Release** 11
 - Welcome to Oracle Solaris 11.2 11
 - Oracle Solaris 10 and Oracle Solaris 11 Feature Comparison 12
 - Removal of Legacy System Management Commands, Files, and Services 17
 - Transitioning an Oracle Solaris 10 System to an Oracle Solaris 11 Release 19
 - Installation Tools and Methods 20
 - Software Management Features 21
 - Networking Features 21
 - System Configuration and SMF Features 22
 - Storage and File Systems Features 23
 - Security Features 24
 - Virtualization Features 24
 - User Account Management and User Environment Features 24
 - Observability, Debugging, and Tuning Features 25
 - Desktop Features 26

- 2 Transitioning to an Oracle Solaris 11 Installation Method** 27
 - Oracle Solaris Installation Features and Methods 27
 - Oracle Solaris Installation Requirements 28
 - ZFS Root Pool Installation Requirements 29
 - Oracle Solaris Preinstallation Tasks 29
 - Installing Oracle Solaris by Using Installation Media 30
 - Transitioning From JumpStart to AI 32
 - Installing Oracle Solaris by Using AI 33
 - AI Feature Enhancements 33
 - AI Preinstallation Tasks 35
 - Setting Up an Install Client 35

Booting the Client and Initiating an Oracle Solaris Installation	36
Installing and Configuring Zones During the AI Process	37
Download Locations for AI Files	37
Additional Installation Tasks	37
Reconfiguring Date and Time Before and After an Installation	38
Monitoring the Live Media Startup Process	39
x86: Adding Custom Entries to the GRUB Menu After an Installation	39
Additional Installation Troubleshooting Information	40
3 Managing Devices	41
Device and Driver Management Changes	41
Preparing Disks for ZFS Storage Pools	43
ZFS Root Pool Installation Improvements	44
ZFS Root Pool Device Requirements	45
ZFS Root Pool Disk and Boot Administration	46
Swap and Dump Device Configuration Changes	48
4 Managing Storage Features	51
Comparing Solaris Volume Manager Configurations to ZFS Configurations	51
ZFS Storage Pool Best Practices	52
ZFS Storage Pool Creation Best Practices	52
ZFS Storage Pool Monitoring Best Practices	54
Troubleshooting ZFS Storage Pool Issues	54
COMSTAR Replaces iSCSI Target Daemon	55
5 Managing File Systems	57
File System Changes	57
Root File System Requirements	58
Mounting File Systems	58
Managing ZFS File Systems	59
Displaying ZFS File System Information	60
Making ZFS File Systems Available	61
Monitoring File Systems	62
Managing Memory Between ZFS and Applications	62
NFS nfsmapid Syntax Changes	63
ZFS File System Sharing Changes	63
ZFS Sharing Migration Issues	64
ZFS Data Deduplication Requirements	65

Considering ZFS Backup Features	66
Migrating File System Data to ZFS File Systems	66
UFS to ZFS Data Migration Best Practices	66
Migrating Data With ZFS Shadow Migration	66
Migrating UFS Data to a ZFS File System	67
6 Managing Software and Boot Environments	69
Software Package Changes	69
Oracle Solaris 10 SVR4 and IPS Package Comparison	70
IPS Installation Package Groups	72
Displaying Information About Software Packages	72
Updating the Software on an Oracle Solaris System	74
Installing Maintenance Updates on an Oracle Solaris 11 System	75
▼ How to Configure the Oracle Solaris Support Repository	75
Managing Boot Environments	76
Tools for Managing Boot Environments	76
Reviewing the Initial ZFS BE After an Installation	77
▼ How to Update Your ZFS Boot Environment	77
7 Managing Network Configuration	79
Network Administration Features	79
Network Virtualization and Advanced Networking Features	82
Comparing the Oracle Solaris 10 Network Protocol Stack to the Oracle Solaris 11 Network Protocol Stack	83
Network Administration Command Changes	86
Comparing the <code>ifconfig</code> Command to the <code>ipadm</code> Command	87
<code>ifconfig</code> Replacement Commands	89
Comparing the <code>ndd</code> Command to the <code>ipadm</code> Command	91
Comparing the <code>ndd</code> Command and <code>driver.conf</code> Configuration to the <code>dladm</code> Command	92
Configuring the Network in Oracle Solaris 11	94
How the Network Is Configured During an Installation	94
Network Administration Task Comparison	95
Administering Datalink Configuration	96
Configuring IP Interfaces and Addresses	97
Configuring Persistent Routes	98
Configuring Naming and Directory Services	99
Administering DHCP	100
Setting a System's Host Name	100

Managing Network Configuration in Reactive Mode	101
8 Managing System Configuration	103
System Configuration Changes	103
Oracle Solaris 10 and Oracle Solaris 11 System Configuration Feature Comparison	105
Service Management Facility Changes	106
Naming and Directory Services Migration to SMF	106
SMF Administrative Changes	107
SMF Manifest Creation Tool	109
System Process Summary Information	110
System Console and Terminal Services Changes	110
Power Management Configuration Changes	111
System Configuration Tools Changes	111
System Registration and Customer Support Changes	112
Boot, Recovery, Platform, Hardware, and Disk Labeling Changes	113
x86: GRand Unified Bootloader Changes	113
Firmware, Disk Labeling, and EEPROM Changes	114
Additional Boot, Platform, and Hardware Changes	115
Booting a System for Recovery	116
System Recovery and Cloning With the Oracle Solaris Unified Archives Feature	122
Printer Configuration and Management Changes	122
Removal of the LP Print Service	123
▼ How to Set Up Your Printing Environment After an Installation	124
Internationalization and Localization Changes	124
Locale, Timezone, and Console Keymap Configuration Changes	127
9 Managing Security	129
Security Feature Changes	129
Network Security Features	131
Pluggable Authentication Module Changes	132
Removed Security Features	132
Roles, Rights, Privileges, and Authorizations	133
About Rights Profiles	135
Viewing Privileges and Authorizations	136
File and File System Security Changes	137
acLmode Property Reintroduced	137
Encrypting ZFS File Systems	139
Immutable Zones	140

10 Managing Oracle Solaris Releases in a Virtual Environment	141
Oracle Solaris Virtualization Features	141
Consolidating Legacy Oracle Solaris Systems With Oracle VM Server	142
Oracle Solaris Zone Features	143
Oracle Solaris Zones Feature Enhancements	145
Oracle Solaris 10 Branded Zones Preparation	145
Transitioning an Oracle Solaris 10 Instance to a Non-Global Zone on an Oracle Solaris 11 System	146
11 Managing User Accounts and User Environments	149
Commands and Tools for Managing User Accounts	149
Managing User Accounts	150
User Account Management Changes	150
User Password and Login Changes	151
Sharing Home Directories That Are Created as ZFS File Systems	152
How Home Directories Are Mounted in Oracle Solaris	153
User Environment Feature Changes	153
Oracle Solaris Man Page Changes	155
12 Managing the Oracle Solaris Desktop	157
Oracle Solaris Desktop Features	157
Key Desktop Features	158
Desktop Features That Have Been Removed	160
Xorg Family of Servers	161
▼ How to Update Custom Hot Key Configurations or Enable Legacy Mappings	162
Troubleshooting Desktop Transition Issues	162
Installing the Oracle Solaris Desktop Software Package After an Installation	162
GNOME Desktop Manager Issues	163

Using This Documentation

- **Overview** – Describes topics for transitioning from Oracle Solaris 10 to Oracle Solaris 11 releases.
- **Audience** – Technicians, system administrators, and authorized service providers.
- **Required knowledge** – Basic knowledge of Oracle Solaris.

Product Documentation Library

Late-breaking information and known issues for this product are included in the documentation library at <http://www.oracle.com/pls/topic/lookup?ctx=E36784>.

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.

Feedback

Provide feedback about this documentation at <http://www.oracle.com/goto/docfeedback>.

About Transitioning From Oracle Solaris 10 to an Oracle Solaris 11 Release

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 11.2” on page 11](#)
- [“Oracle Solaris 10 and Oracle Solaris 11 Feature Comparison” on page 12](#)
- [“Removal of Legacy System Management Commands, Files, and Services” on page 17](#)
- [“Transitioning an Oracle Solaris 10 System to an Oracle Solaris 11 Release” on page 19](#)
- [“Installation Tools and Methods” on page 20](#)
- [“Software Management Features” on page 21](#)
- [“Networking Features” on page 21](#)
- [“System Configuration and SMF Features” on page 22](#)
- [“Storage and File Systems Features” on page 23](#)
- [“Security Features” on page 24](#)
- [“Virtualization Features” on page 24](#)
- [“User Account Management and User Environment Features” on page 24](#)
- [“Observability, Debugging, and Tuning Features” on page 25](#)
- [“Desktop Features” on page 26](#)

Welcome to Oracle Solaris 11.2

The Oracle Solaris 11 operating system (OS) is the an operating system for an enterprise-level environment. Oracle Solaris 11.2, the latest Oracle Solaris release, 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. For specific information about which features are supported in a particular Oracle Solaris 11 release, see the product documentation.

Most Oracle Solaris 10 applications are known to work on Oracle Solaris 11. You can run supported applications *as is*. To determine the readiness of Oracle Solaris 10 applications to run on Oracle Solaris 11, use the Oracle Solaris 11 compatibility checking tool at <http://www.oracle.com/technetwork/server-storage/solaris11/downloads/preflight-checker-tool-524493.html>.

Alternatively, 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”](#).

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

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

- For more details about new features, see [“What’s New in Oracle Solaris 11.2”](#).
- For more details about excluded features, see <http://www.oracle.com/technetwork/systems/end-of-notice/index.html>.
- For information about updating your system to Oracle Solaris 11.2, see [“Updating to Oracle Solaris 11.2”](#).
- For information about Oracle's Sun hardware platforms and any corresponding Oracle Solaris operating system requirements, see <http://www.oracle.com/technetwork/systems/software-stacks/stacks/index.html>.

Oracle Solaris 10 and Oracle Solaris 11 Feature Comparison

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
x86: boot loader (GRUB)	GRUB Legacy (0.97)	GRUB 2

Feature or Command	Oracle Solaris 10	Oracle Solaris 11
		“GRand Unified Bootloader Changes” on page 113
boot loader (administration)	SPARC: installboot x86: installgrub	bootadm install-bootloader (SPARC and x86)
Booting (from a root device)	From a ZFS, UFS, or Solaris Volume Manager root device	From a ZFS root file system “Boot, Recovery, Platform, Hardware, and Disk Labeling Changes” on page 113
Booting (from the network)	SPARC: From the OpenBoot PROM (OBP) ok 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: PXE boot process has changed for UEFI firmware <i>only</i> “Booting Systems With UEFI and BIOS Firmware From the Network” in “Booting and Shutting Down Oracle Solaris 11.2 Systems”
Booting (recovery)	SPARC: From the OBP ok prompt: boot -F failsafe x86: Select the failsafe boot entry in the GRUB menu at boot time	Failsafe mode no longer supported on x86 and SPARC platforms. “Boot, Recovery, Platform, Hardware, and Disk Labeling Changes” on page 113 Oracle Solaris Unified Archives “System Recovery and Cloning With the Oracle Solaris Unified Archives Feature” on page 122
Crash dump directory location	/var/crash/system-name	/var/crash
Database management system (MySQL)	Release series 5.1	Release series 5.1 and release series 5.5 Upgrade from MySQL 5.1 to 5.5. See “Update from MySQL 5.1 to MySQL 5.5” in “Oracle Solaris 11.2 Release Notes” .
Desktop environment	Common Desktop Environment (CDE) (default) and GNOME 2.6 (optional)	Oracle Solaris Desktop (GNOME 2.30) Chapter 12, “Managing the Oracle Solaris Desktop”
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)	x86 and SPARC with GPT-aware firmware: 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)
Ensuring system is configured securely	Solaris Security Toolkit (SST)	sysconfig profiles

Feature or Command	Oracle Solaris 10	Oracle Solaris 11
	netserVICES limited	Secure by Default (SBD) compliance command
File systems (default)	ZFS, UFS, or Solaris Volume Manager root file systems	ZFS root file system (default) Chapter 5, “Managing File Systems”
x86: Firmware support	BIOS	UEFI and BIOS Chapter 3, “Managing Devices”
GRUB configuration file (default)	menu.lst	grub.cfg “GRand Unified Bootloader Changes” on page 113
GRUB configuration file (custom)	menu.lst	custom.cfg
Installation (graphical user interface (GUI))	GUI installation program on DVD or CD	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)
Installation (automated)	JumpStart feature of Oracle Solaris 10	Automated Installer (AI) feature of Oracle Solaris 11 Oracle VM Manager Ops Center
Installation (automated client configuration)	JumpStart profile files	AI manifests
Installation (other)	Oracle Solaris Flash Archive installation	Oracle Solaris Unified Archives “System Recovery and Cloning With the Oracle Solaris Unified Archives Feature” on page 122
Internationalization and localization configuration	localeadm	nlsadm You might need to install the appropriate software package prior to using the nlsadm command on your Oracle Solaris 11.2 system. “Internationalization and Localization Changes” on page 124
Java version (default)	Java 6	Java 7 Java 8 (optional)
Network administration (fixed mode)	ifconfig Edit /etc/hostname.*	dladm for datalinks, ipadm for IP configuration

Feature or Command	Oracle Solaris 10	Oracle Solaris 11
	ndd for configuring protocols (tunables)	Chapter 7, “Managing Network Configuration”
Network administration (reactive mode)	Not applicable	netcfg and netadm Chapter 7, “Managing Network Configuration”
Network administration (DHCP)	Sun DHCP and other name services configuration	ISC DHCP and legacy Sun DHCP
Network administration (IP network multipathing (IPMP))	ifconfig, plumb, and umplumb	dladm and ipadm “Comparing the ifconfig Command to the ipadm Command” on page 87
Network administration (TCP/IP properties or tunables)	ndd driver.conf	dladm and ipadm “Comparing the ndd Command to the ipadm Command” on page 91 and “Comparing the ndd Command and driver.conf Configuration to the dladm Command” on page 92
Network administration (wireless)	wifconfig	Fixed mode: dladm and ipadm Reactive mode: netcfg, and netadm From the desktop: Network administration GUI
Packaging (software management)	SVR4 package and patch commands	IPS pkg(1) commands and utilities
Print service (default)	LP print service, lp print commands, Solaris Print Manager GUI	CUPS “Printer Configuration and Management Changes” on page 122
Security management	root as a system account	root as a role Chapter 9, “Managing Security”
Sun Oracle server management	SPARC and x86: Oracle Hardware Management Pack available as a separate download	SPARC and x86: Oracle Hardware Management Pack: a set of commands and agents for managing Sun Oracle servers (Packages are included, starting with Oracle Solaris 11.2) www.oracle.com/goto/ohmp/solarisdocs
System clustering	Oracle Solaris Cluster 3.3	Oracle Solaris Cluster 4.2
System configuration and reconfiguration)	sysidtool, sys-unconfig, sysidconfig, and sysidcfg	sysconfig, SCI tool, SC profiles

Feature or Command	Oracle Solaris 10	Oracle Solaris 11
System configuration (Oracle Solaris kernel configuration)	Add to /etc/system	Add to /etc/system Add to files within /etc/system.d
System configuration (naming services)	Configured in various files within /etc and /var	Managed by Service Management Facility (SMF) commands
System configuration (setting host name)	Edit /etc/nodename	hostname command “System Configuration Changes” on page 103
System management (centralized)	All versions of Ops Center support Oracle Solaris 10	For support information, see the <i>Certified Systems Matrix</i> document at http://www.oracle.com/pls/topic/lookup?ctx=oc122
System recovery and cloning (automated)	Oracle Solaris Flash Archive feature	Oracle Solaris Unified Archives “System Recovery and Cloning With the Oracle Solaris Unified Archives Feature” on page 122
System registration and service request support	Auto Registration feature Oracle Configuration Manager (starting with Oracle Solaris 10 1/13)	Oracle Configuration Manager and the Oracle Auto Service Request utility
System upgrade and BE management	lu and SVR4 package commands	pkg commands beadm utility to manage boot environments Chapter 6, “Managing Software and Boot Environments”
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 User Manager GUI “Commands and Tools for Managing User Accounts” on page 149
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
ZFS root pool disk (SPARC and x86)	Root pool disk requires SMI (VTOC) disk label and a slice 0	“ZFS Root Pool Disk and Boot Administration” on page 46

Feature or Command	Oracle Solaris 10	Oracle Solaris 11
Zones environment	Oracle Solaris 10 branded zones, legacy branded zones	Supported Oracle Solaris 10 and Oracle Solaris 11 zones features and Oracle Solaris Kernel Zones (solaris-kz branded zones), starting with Oracle Solaris 11.2

Removal of Legacy System Management Commands, Files, and Services

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

TABLE 1-2 Legacy System Management Commands, Files, and Services

Legacy Command, File, or Service	Replacement Command, File, or Service	For More Information
bsmconv and bsmunconv	audit	audit(1M)
crypt and des	encrypt	encrypt(1)
/etc/defaultrouter (deprecated)	route	route(1M)
graph and spline	gnuplot	gnuplot(1) Note - Install the image/gnuplot package.
SPARC: installboot x86: installgrub (deprecated and should only be used to install boot blocks on systems that support GRUB Legacy)	bootadm install-bootloader (SPARC and x86)	“ZFS Root Pool Disk and Boot Administration” on page 46
localeadm	nlsadm (starting with Oracle Solaris 11.2) Note that you might need to install the software package prior to using the command.	“Internationalization and Localization Changes” on page 124
Print commands: download, lpfilter, lpforms, lpget, lpset, lpsched, lpshut, lpssystem, 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 122
Print (LP) files and descriptions:	CUPS print files and descriptions:	lpoptions(1)

Legacy Command, File, or Service	Replacement Command, File, or Service	For More Information
<ul style="list-style-type: none"> ■ ~/.printers ■ /etc/printers.conf ■ /etc/lp/printers ■ /var/spool/lp ■ /var/lp/logs 	<ul style="list-style-type: none"> ■ ~/.cups/lpoptions ■ /etc/cups/printers.conf ■ /etc/cups ■ /var/spool/cups ■ /var/log/cups 	
<p>Legacy SMF print services:</p> <ul style="list-style-type: none"> ■ 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 <p>Replacement SMF print services:</p> <ul style="list-style-type: none"> ■ svc:/application/cups/scheduler ■ svc:/application/cups/in-lpd 		<p>“Printer Configuration and Management Changes” on page 122</p>
pmconfig and /etc/power.conf	poweradm	poweradm(1M)
rdist	rsync or scp	rsync(1) and scp(1)
rstart and rstartd	ssh	ssh(1)
listen, nlsadmin, pmadm, sac, sacadm, saf, and ttyadm /usr/include/listen.h, getty, /usr/lib/saf/nlps_server, /var/saf, /etc/saf, ttymon (sac and getty modes <i>only</i>), and ports (sac functionality)	<p>ttymon express mode is still supported by the following SMF services:</p> <ul style="list-style-type: none"> ■ svc:/system/console-login:terma ■ svc:/system/console-login:termb 	<p>“System Console and Terminal Services Changes” on page 110</p>
<p>Network SMF services:</p> <p>svc:/network/physical:default</p> <p>svc:/network/physical:nwam (deprecated in Oracle Solaris 11, but the service is still listed in the output of the svcs -a command)</p>	svc:/network/physical:default	<p>Chapter 7, “Managing Network Configuration”</p>
smosservice and smdiskless	No replacement available	Not applicable
sysidtool, sys-unconfig, and sysidcfg	sysconfig, SCI tool, and SC configuration through profiles	<p>“System Configuration Tools Changes” on page 111</p>
User account management:	useradd, usermod, userdel, groupadd, groupmod, groupdel, roleadd, rolemod, roledel	<p>“Managing User Accounts” on page 150</p>

Legacy Command, File, or Service	Replacement Command, File, or Service	For More Information
Solaris Management Console GUI, smc, smuser, smgroup, and passmgmt	Starting with Oracle Solaris 11.1: User Manager GUI	
vold daemon	volfs and rmvolmgr	Chapter 3, “Managing Devices”

For more information about legacy commands that are no longer supported, see <http://www.oracle.com/technetwork/systems/end-of-notice/index.html>.

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

Keep the following key points in mind when transitioning from Oracle Solaris 10 to an Oracle Solaris 11 release:

- No upgrade methods or tools are available for transitioning from Oracle Solaris 10 to an Oracle Solaris 11 release. You cannot use an installer to upgrade from Oracle Solaris 10 to Oracle Solaris 11. 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. See [“Transitioning From JumpStart to AI” on page 32](#) and [“Tools for Managing Boot Environments” on page 76](#).
- The Oracle Solaris System Archive and Cloning feature provides functionality that is similar to the Oracle Solaris Flash Archive installation method. See [“System Recovery and Cloning With the Oracle Solaris Unified Archives Feature” on page 122](#).
- Oracle Solaris 11 supports 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”](#).

The following table 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 11.2 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”
NFS file sharing and pool migration	Used to access shared files from an Oracle Solaris 10 system on an Oracle Solaris 11 system. Also 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 Tools and Methods

The following installation methods are available:

- **x86: GUI installation with the Live Media** – You can use the GUI installer to install Oracle Solaris 11 on x86 platforms *only*. The GUI installer is capable of functioning with a minimum of 1.5 GB of memory. The exact minimum requirement varies, depending on system specifications. See [“Installing Oracle Solaris by Using Installation Media” on page 30](#) for details.
- **Interactive text installation (from media or over the network)** – You can use the text installer 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 client system 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 33](#).
AI also supports the installation of zones. See [“Oracle Solaris Zone Features” on page 143](#).
- **Customized installation image creation with the Distribution Constructor** – The Distribution Constructor tool creates preconfigured installation images. See [“Creating a Custom Oracle Solaris 11.2 Installation Image ”](#).

The following installation tools and methods are no longer available:

- **Oracle Solaris Flash Archive Installation** – You can use the Oracle Solaris Unified Archives feature to perform system cloning and recovery operations. Oracle Solaris Unified Archives are system archives that can contain one or more archived instances of the OS. Each instance is an independently referenced system. See [“System Recovery and Cloning With the Oracle Solaris Unified Archives Feature” on page 122](#).
- **JumpStart feature of Oracle Solaris** – AI replaces JumpStart in Oracle Solaris 11. See [“Installing Oracle Solaris by Using AI” on page 33](#).
- **Oracle Solaris Live Upgrade feature** – The suite of commands (`lu`) that are part of the Oracle Solaris Live Upgrade feature are not supported. The `beadm` utility provides similar capability. See [“Tools for Managing Boot Environments” on page 76](#).

See [Chapter 2, “Transitioning to an Oracle Solaris 11 Installation Method”](#).

Software 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.

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 Oracle Solaris 11 version to a newer Oracle Solaris 11 version.

See [Chapter 6, “Managing Software and Boot Environments”](#).

Networking Features

The following key feature changes are related to network administration:

- **Generic datalink names** – Oracle Solaris 11 assigns generic names to each datalink on a system by using the `net0`, `net1`, `netN` naming convention. See [Chapter 2, “Administering Datalink Configuration in Oracle Solaris,”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).
- **Naming and directory services configuration** – This configuration is managed through SMF rather than by editing various files within the `/etc` directory, as in Oracle Solaris 10 and previous releases. See [“Configuring Naming and Directory Services”](#) on page 99.
- **Network administration commands** – The following three commands are primarily used to manage persistent network configuration:
 - `dladm` command – Manages datalink configuration, both physical and other types of datalinks. The `dladm` command also replaces the `ndd` command and `drive.conf` file for the configuration of certain network parameters (tunables).
 - `ipadm` command – Creates persistent configuration of IP interfaces and addresses. This command effectively replaces the `ifconfig` command for IP configuration. The `ipadm` command also replaces the `ndd` command for configuring certain network parameters (tunables). [Chapter 5, “Internet Protocol Suite Tunable Parameters,”](#) in [“Oracle Solaris 11.2 Tunable Parameters Reference Manual”](#).
 - `route` command – Configures persistent routes. This command replaces the use of the `/etc/defaultrouter` file for managing a system route configuration. See [“Network Administration Command Changes”](#) on page 86.
- **Network security features** – Oracle Solaris provides several new security features, as well as enhancements to several existing security features. See [“Network Security Features”](#) on page 131.
- **Network virtualization features** – Oracle Solaris 11 provides several network virtualization features that you can use for high availability, managing network resources, and improving overall network performance, for example, aggregations, bridging technologies, virtual local area networks (VLANs), virtual network interface cards (VNICs), and virtual switches. See [“Network Virtualization and Advanced Networking Features”](#) on page 82.

See [Chapter 7, “Managing Network Configuration”](#).

System Configuration and SMF Features

The following system configuration and SMF feature changes have been made:

- **Oracle Auto Service Request utility** – Customers who have a valid My Oracle Support account to automatically log service requests can use this utility. See [“System Registration and Customer Support Changes”](#) on page 112.
- **SMF administrative layers** – Enable you to record the source of properties, property groups, instances, and services. This information enables you to determine which settings

are administrative customizations, which settings were provided in an SMF profile, and which settings were delivered by an SMF manifest. See [“SMF Administrative Changes” on page 107](#).

- **SMF manifest creation tool** – You can use the `svcbundle` command to generate SMF manifests, as well as profiles. See [`svcbundle\(1M\)`](#).
- **System Configuration Interactive (SCI) utility** – Centralizes configuration information through SMF. The `sysconfig` utility replaces the `sys-unconfig` and `sysidtool` utilities that are used in Oracle Solaris 10. See [“System Configuration Tools Changes” on page 111](#).
- **System registration With 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. See [“System Registration and Customer Support Changes” on page 112](#).

See [Chapter 8, “Managing System Configuration”](#).

Storage and File Systems Features

The following key feature changes are related to storage and file systems management:

- **Device management** – New commands are available and existing commands have been updated to help you locate storage devices by their physical locations.
- **Storage solutions** – Oracle's Sun ZFS Storage Appliance provides a low-cost storage solution and simplified administration with a browser-based management and monitoring tool. You use the appliance 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.
- **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** – You can use the ZFS shadow migration to migrate your 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. See [“Transitioning an Oracle Solaris 10 System to an Oracle Solaris 11 Release” on page 19](#).
- **Migrate your Oracle Solaris 10 storage pools** – You can export and disconnect storage devices that contain ZFS storage pools on your Oracle Solaris 10 systems and then import them into your Oracle Solaris 11 systems.

- **Other ways to migrate your UFS data** – You can remotely mount UFS file systems from an Oracle Solaris 10 system onto an Oracle Solaris 11 system. In addition, you can use the `ufsrestore` command to restore UFS data (`ufsdump`) into a ZFS file system.

See [Chapter 4, “Managing Storage Features”](#) and [Chapter 5, “Managing File Systems”](#).

Security Features

Security feature enhancements have been made in the following areas:

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

See [Chapter 9, “Managing Security”](#).

Virtualization Features

The following virtualization features are supported:

- Oracle Solaris Zones
- Oracle VM Server for SPARC
- Oracle VM Server for x86
- Oracle VM Templates
- Oracle VM VirtualBox

See [Chapter 10, “Managing Oracle Solaris Releases in a Virtual Environment”](#).

User Account Management and User Environment Features

User account management and the Oracle Solaris user environment changes include the following:

- Administrative command locations
- Creating and managing user accounts

- Default user shell and path changes
- Development tools locations

See [Chapter 11, “Managing User Accounts and User Environments”](#).

Observability, Debugging, and Tuning Features

Observability, debugging, and tuning feature changes include the following:

- **DTraces feature changes** – DTrace feature changes include the following:
 - **errexit option** – An additional DTrace consumer option that specifies whether a DTrace script should exit upon encountering an error has been added. This enhancement changes the previous DTrace behavior, where an error is reported, but the script does not terminate.
 - **lquantize() action** – Support for a new linear-log quantize aggregating action has been added. This aggregating action enables you to collect data in linear-step buckets (similar to the existing lquantize() action) across multiple magnitudes simultaneously.
 - **Scalability improvements** – Internal processing for DTrace includes some scalability improvements that support better performance on larger systems.
 - **Structure and bitfield enhancements** – The expected behavior of user-defined structures and bitfields has been modified to adhere to the appropriate ABI specification for padding. This change might require you to remove any variables that were previously introduced as a workaround from your DTrace scripts.
 - **tracemem() enhancements** – This action includes an additional argument that specifies the number of bytes to display, which might be less than the number of bytes that was traced.

For additional information, see [“Oracle Solaris 11.2 Dynamic Tracing Guide”](#).

- **Observing users and processes** – Use the -u option with the netstat to observe which users and processes are responsible for network connections. See [netstat\(1M\)](#) and [“Displaying User and Process Information”](#) in [“Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2”](#).
- **System tuning features** – The following system tuning feature enhancements have been made:
 - **NFS-related SMF configuration parameter changes** – The network/nfs/server service includes the nfs-props property group, which provides configurable parameters for controlling the refresh of the NFS authentication cache and the mountd netgroup cache. See [Chapter 4, “NFS Tunable Parameters,”](#) in [“Oracle Solaris 11.2 Tunable Parameters Reference Manual”](#).
 - **Oracle Solaris ZFS tunable parameters flash storage changes** – When using ZFS with Flash storage, refer to [Chapter 3, “Oracle Solaris ZFS Tunable Parameters,”](#) in

[“Oracle Solaris 11.2 Tunable Parameters Reference Manual ”](#) for updated information pertaining to the following:

- F20 PCIe Accelerator Card
- F40 PCIe Accelerator Card
- F80 PCIe Accelerator Card
- F5100 Flash Storage Array
- Flash SSDs

Desktop Features

The default desktop is 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 an Oracle Solaris 11 release, and you previously customized your CDE login, review your display management configuration. You might need to make some changes to the GDM configuration to ensure it works as expected. See [“Troubleshooting Desktop Transition Issues” on page 162](#).

See [Chapter 12, “Managing the Oracle Solaris Desktop”](#).

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.

For detailed instructions on installing Oracle Solaris 11.2, see [“Installing Oracle Solaris 11.2 Systems”](#). For detailed installation instructions on installing another Oracle Solaris 11 release, see the relevant Oracle Solaris 11 installation product documentation for that release.

For information about upgrading your system to Oracle Solaris 11.2, see [“Updating to Oracle Solaris 11.2”](#).

For information about installing an Oracle Solaris virtual image on Oracle VM VirtualBox, see <http://www.oracle.com/technetwork/server-storage/solaris11/downloads/virtual-machines-1355605.html>.

The following topics are covered:

- [“Oracle Solaris Installation Features and Methods”](#) on page 27
- [“Oracle Solaris Installation Requirements”](#) on page 28
- [“Installing Oracle Solaris by Using Installation Media”](#) on page 30
- [“Transitioning From JumpStart to AI”](#) on page 32
- [“Installing Oracle Solaris by Using AI”](#) on page 33
- [“Additional Installation Tasks”](#) on page 37

Oracle Solaris Installation Features and Methods

The following table summarizes the installation features and methods that are available in this release. With the exception of the Automated Installer (AI) method, all of these installation methods are used to install single systems. You can use AI to install single or multiple systems over the network.

TABLE 2-1 Supported Installation Methods

Installation Method	Preparation?	Install Server?	Single or Multiple Systems
Live Media installation (x86 only)	Minimal	No	Single
Text installation	Minimal	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

Note - With the exception of the Live Media (x86 only) and the text installation methods, each of these installation methods installs software packages from an Oracle Solaris Image Packaging System (IPS) repository.

The following installation features are no longer supported:

- **Oracle Solaris Flash Archive Installation** – Use the Oracle Solaris Unified Archives feature. See “[System Recovery and Cloning With the Oracle Solaris Unified Archives Feature](#)” on page 122.
- **JumpStart feature of Oracle Solaris** – Use the AI feature. See “[Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.2 Automated Installer](#)”.

Oracle Solaris Installation Requirements

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

- **Memory** – The minimum memory requirement for installation 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 specific system specifications.
- **Hardware** – Any supported SPARC or x86 platform. See <http://www.oracle.com/webfolder/technetwork/hcl/index.html>.
- **Virtual memory** – If you want to install an Oracle Solaris 11 virtual image on Oracle VM VirtualBox, refer to the memory requirements that are listed at <http://www.oracle.com/technetwork/server-storage/solaris11/downloads/virtual-machines-1355605.html>.

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:

- **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 that are 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 sizes of your own choosing, as long as the new sizes support system operations. See [“Managing Your ZFS Swap and Dump Devices”](#) in [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).
 - **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 11.2 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 System for Installing Multiple Operating Systems”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).
- **Ensure that you have the proper device drivers** – Before installing Oracle Solaris, 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 11.2 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 keeps the Real Time Clock (RTC) in Coordinated Universal time (UTC) format. The behavior on x86 platforms is different than it was in Oracle Solaris 10. AI does not adjust the RTC date and time during an installation. To reconfigure the date and time after an installation, see [“Reconfiguring Date and Time Before and After an Installation”](#) on page 38.

Installing Oracle Solaris by Using Installation Media

You can install Oracle Solaris 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 28.

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 11.2 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 11.2 Systems”](#).

To install the OS by using 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. USB images require the `usbcopy` utility to copy the bootable ISO image to a USB flash drive. Starting with Oracle Solaris 11.2, USB installation media is also available for SPARC platforms. To use the `usbcopy` utility, first install the `pkg:/install/distribution-creator` package. For instructions on creating a persistent device alias for a USB stick on a SPARC based system, see [“How to Create a Persistent Device Alias for a USB Flash Drive on a SPARC System”](#) in [“Installing Oracle Solaris 11.2 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 installer can use the entire disk. If the whole disk option is selected, 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 11.2 Systems”](#). If you use the text installer, you might need to install additional software packages afterwards. See [“Adding Software After a Text Installation”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).

Note - The text installer installs the `solaris-large-server` package set. However, if you use the text installer over the network, the `solaris-autoinstall` package set is installed. After booting the installed system, you should install the `solaris-large-server` package set.

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 Start a Text Installation Over the Network”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).

■ **Automated installations that boot from media**

You can boot an AI image from media or a USB device (x86 only) to initiate a hands-free installation of just that system. An AI manifest provides installation instructions for the system. Starting with Oracle Solaris 11.2, you can use the interactive AI manifest wizard to simply AI manifest creation. See [“Creating a AI Manifest Using the AI Manifest Wizard”](#) in [“Installing Oracle Solaris 11.2 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 11.2 Systems”](#).

You can also create custom Live Media images, text installer images, and AI images. See [“Creating a Custom Oracle Solaris 11.2 Installation Image”](#).

Note - After installing your system, you cannot update it by using a method that is similar to any Oracle Solaris 10 upgrade methods. An Oracle Solaris system is updated based on your desired maintenance schedule and by using the `pkg` command. See [“Adding and Updating Software in Oracle Solaris 11.2”](#) and [“Updating to Oracle Solaris 11.2”](#) for more details.

Before updating to Oracle Solaris 11.2, see [“Issues When Updating to Oracle Solaris 11.2”](#) in [“Oracle Solaris 11.2 Release Notes”](#).

The media paths for the Oracle Solaris 11.2 installers are as follows:

- **x86 only: Live Media** – `Oracle_Solaris-11_2-Live-X86`
- **SPARC: Interactive text installer** – `Oracle_Solaris-11_2-Text-SPARC`
- **x86: Interactive text installer** – `Oracle_Solaris-11_2-Text-X86`
- **SPARC: Automated Installer** – `Oracle_Solaris-11_2-AI-SPARC`
- **x86: Automated Installer** – `Oracle_Solaris-11_2-AI-X86`

Transitioning From JumpStart to AI

AI performs automated (or hands-free) 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 11.2 Automated Installer”](#).

You can use the `js2ai` utility to migrate 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.

To use the `js2ai` utility, you must first install the following software package:

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

You can use the `js2ai` utility to perform several migration tasks, including the following:

- **Replace JumpStart rules and profile files with AI criteria files and AI manifests.**
See [“Using js2ai To Convert JumpStart Rules and Profiles to AI Criteria and Manifests”](#) in [“Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.2 Automated Installer”](#).
- **Convert JumpStart files to an AI configuration files.**
See [“Using js2ai to Convert sysidcfg Files to System Configuration Profiles”](#) in [“Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.2 Automated Installer”](#).
- **Set up an install server.**
See [Chapter 4, “Installing Oracle Solaris 10 by Using JumpStart on an Oracle Solaris 11 Server,”](#) in [“Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11.2 Automated Installer”](#).

Note - Customers with a valid My Oracle Support contract can also set up an Oracle Solaris 10 1/13 system as an AI install server by installing additional software packages. This offering enables you to install the Oracle Solaris 11 11/11 release only. See *Oracle Solaris 11 Provisioning Assistant for Oracle Solaris 10: Installation Guide* (Doc ID 1495735.1) and *Oracle Solaris 11 Provisioning Assistant for Oracle Solaris 10: Release Notes* (Doc ID 1495775.1) at <https://support.oracle.com/>.

- **Dynamically derive an AI provisioning manifest.**
See [“Creating an AI Manifest at Client Installation Time”](#) in [“Installing Oracle Solaris 11.2 Systems”](#)
- **Access a software package repository for AI installations.**
See [Chapter 2, “Copying IPS Package Repositories,”](#) in [“Copying and Creating Package Repositories in Oracle Solaris 11.2”](#).

- **Provide system configuration instructions.**
See [Chapter 11, “Configuring the Client System,”](#) in [“Installing Oracle Solaris 11.2 Systems ”](#).
- **Create an SMF service that runs on first boot and executes a user-defined script.**
See [Chapter 13, “Running a Custom Script During First Boot,”](#) in [“Installing Oracle Solaris 11.2 Systems ”](#).

See the [js2ai\(1M\)](#) man page for more details.

Installing Oracle Solaris by Using AI

You can use the AI installation method to perform a hands-free installation of Oracle Solaris on single or multiple systems. This installation method requires an install server setup. See [Part III, “Installing Using an Install Server,”](#) in [“Installing Oracle Solaris 11.2 Systems ”](#). Also, each system that is to be installed must have network access to retrieve necessary packages during the installation process from a networked IPS repository.

Keep the following key points in mind when using AI:

- 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 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 Software in Oracle Solaris 11.2 ”](#).
- You can optionally customize installation clients with specific installation parameters, for example, disk layout and software selection.
- You can optionally customize clients with specific system configuration parameters, for example, host name, network configuration, and user account information.
- You can make customizations on a client-by-client basis, as well as scale customizations for large enterprise environments.

For additional information about the AI process, see [“Booting an AI Client”](#) in [“Installing Oracle Solaris 11.2 Systems ”](#)

AI Feature Enhancements

The following AI feature enhancements have been made in this release:

- **AI manifest wizard** – Oracle Solaris 11.2 includes a new interactive browser interface that you can use to create AI manifests for use on an AI server. See [“Creating a AI Manifest Using the AI Manifest Wizard”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).
- **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 11.2 Systems”](#).
- **Configuring multiple network interfaces during installation** – This release includes a new svc:/network/install:default SMF service that has two new property group types: ipv4_interface and ipv6_interface for creating SC profiles that contain property groups with the type ipv4_interface and ipv6_interface. See [“Configuring Network Interfaces”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).
- **End-to-end secure provisioning for SPARC platforms** – AI supports a more secure installation for SPARC based systems by using the SPARC WAN boot through process to retrieve installation packages from an IPS repository. This improvement provides more secure communications between the installation server and client systems. See [“Increasing Security for Automated Installations”](#) in [“Installing Oracle Solaris 11.2 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 recently added Oracle Solaris installation screens. See [Appendix A, “Working With Oracle Configuration Manager,”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).
- **Interactive installation on iSCSI targets** – The ability to install to iSCSI target logical unit numbers (LUNs) is included in the Oracle Solaris 11 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 11.2 Systems”](#).
- **Rights profiles and authorizations for managing the AI 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 [“AI Server Requirements”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).

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 Oracle Solaris. 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. The default AI manifest specifies the most recent version of the Oracle Solaris 11 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 AI Server,”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).

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 11.2 Systems”](#)
- [Chapter 10, “Provisioning the Client System,”](#) in [“Installing Oracle Solaris 11.2 Systems”](#)
- [Chapter 11, “Configuring the Client System,”](#) in [“Installing Oracle Solaris 11.2 Systems”](#)

Setting Up an Install Client

When you initially set up your install server, you must create 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 you do not provide adequate system configuration instructions prior to the installation, an interactive tool opens during the first boot after the installation, prompting you for 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_2-sparc` install service as follows:

```
# installadm create-client -n solaris11_2-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 Install Service”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).

For information about setting up an x86 install client, see [“Setting Up an x86 Client”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).

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` OBP 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 11.2 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 key 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 11.2 Systems ”](#).

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

Oracle Solaris 11.2 Text Installer and command line
Oracle Solaris 11.2 Automated Install

See [“Installing an x86 Client”](#) in [“Installing Oracle Solaris 11.2 Systems ”](#) for a list of the events that occur during an x86 client installation.

For examples of a different types of installation scenarios, see [“Automated Installer Use Cases”](#) in [“Installing Oracle Solaris 11.2 Systems ”](#).

Installing and Configuring Zones During the AI Process

Non-global zones are installed and configured on the first reboot after the global zone is installed. With AI, you can install non-global zones 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 installing it. See [Chapter 12, “Installing and Configuring Zones,”](#) in [“Installing Oracle Solaris 11.2 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`
- **SC profiles that are downloaded from the AI server** – `/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.

Reconfiguring 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 this release 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, check or uncheck the Hardware Clock in UTC time setting in the system preferences for the virtual machine.

Switching From Local Time Format to UTC Format

To switch from local time format to UTC format, set the time lag between the kernel and RTC to 0 (zero) as follows:

```
# rtc -z GMT
```

If the date/time requires an adjustment, use the `date` command. See [date\(1\)](#).

Switching From UTC Format to Local Time Format

When the switch from UTC to local time is complete, and each time you reconfigure the time zone setting by using the `sysconfig` command, run the `rtc timezone` command with the `-z` option as follows:

```
# rtc -z timezone
```

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.

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.

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

Starting with Oracle Solaris 11.1, GRUB 2 is the default boot loader on x86 platforms. GRUB 2 uses a different configuration file (`grub.cfg`) than the `menu.lst` file that is used by GRUB Legacy. The `grub.cfg` file contains most of the GRUB configuration, including all Oracle Solaris menu entries. Unlike the `menu.lst` file, you manage the `grub.cfg` file *solely* by using the `bootadm` command. Do *not* directly edit this file.

Also, the `grub.cfg` file does not contain any custom menu entries. For custom menu entries, there is an additional configuration file (`custom.cfg`) that you can use. Before adding custom menu entries to the `custom.cfg` first, you first have to create the file and then store it in the same location as the `grub.cfg` and `menu.conf` files (`/pool-name/boot/grub/`) are stored.

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 then processes any commands within the file as though the contents were textually inserted in the main `grub.cfg` file.

For example, on a system with 64-bit UEFI firmware, `custom.cfg` file entries 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 11.2 Systems”](#).

Additional Installation Troubleshooting Information

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

- Before installing Oracle Solaris, review [“Installation Considerations”](#) in [“Oracle Solaris 11.2 Release Notes”](#).
- For information about troubleshooting issues that you might encounter during an installation, see [Chapter 2, “Installation Issues,”](#) in [“Oracle Solaris 11.2 Release Notes”](#).
- For information about troubleshooting issues that you might encounter when upgrading to Oracle Solaris 11.2, see [Chapter 3, “Update Issues,”](#) in [“Oracle Solaris 11.2 Release Notes”](#).
- If you are installing Oracle Solaris on an x86 based system with Live Media, see [“Initial Root Password Expires After Live Media Installation”](#) in [“Oracle Solaris 11.2 Release Notes”](#).
- If you are installing Oracle Solaris with AI, see [Chapter 15, “Troubleshooting Automated Installations,”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).
- For information about troubleshooting issues with booting a system after an installation, see [“What to Do If Your System Boots in Console Mode”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).

Managing Devices

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

The following topics are covered:

- “Device and Driver Management Changes” on page 41
- “Preparing Disks for ZFS Storage Pools” on page 43
- “Swap and Dump Device Configuration Changes” on page 48

Device and Driver Management Changes

Device and driver identity and configuration has changed as follows:

- Starting with Oracle Solaris 11.2, the Oracle Hardware Management Pack package is included in the Oracle Solaris release. Previously, this package was available as a separate download. These features provide cross platform components to assist you in managing your hardware better. For more information, go to www.oracle.com/goto/ohmp/solaris. See also the detailed documentation at www.oracle.com/goto/ohmp/solarisdocs.
- As in Oracle Solaris 10 releases, all supported devices that are connected to the system when it is installed should be accessible after installation. You can configure devices by using the `cfgadm` command and most devices are hot-pluggable, which means you can add and remove devices while the system is booted.
- The `hotplug` command provides offline and online capabilities, as well as enable and disable operations for PCI Express (PCIe) and PCI SHPC (Standard Hot Plug Controller) devices. Note that you can still use the `cfgadm` command to manage hot pluggable USB and SCSI devices. See Chapter 2, “Dynamically Configuring Devices,” in “Managing Devices in Oracle Solaris 11.2”.
- You can more easily identify devices by using the `crainfo` command to identify physical device location information.

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/HD0` 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 previous 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, as shown in this 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 as follows 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 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 requires 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.

- 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. See [“How to Customize a Driver Configuration”](#) in [“Managing Devices in Oracle Solaris 11.2”](#).

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, as shown in the following 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

- Whenever possible, create non-root pools with whole disks.

Oracle Solaris releases support advanced format disks in addition to traditional 512n disks. See [“Using Advanced Format Disks”](#) in [“Managing Devices in Oracle Solaris 11.2”](#).

ZFS Root Pool Installation Improvements

Review the following installation improvements for root pools:

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

Starting with Oracle Solaris 11.1, SPARC based systems with GPT-aware firmware and most x86 based systems are installed with an EFI (GPT) label on the root pool disk or disks. See [“SPARC: GPT Labeled Disk Support”](#) in [“Oracle Solaris 11.2 Release Notes”](#) for further instructions.

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 enable 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 automated installation. For example, the following 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.

- In 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, as shown in the following example output:

```

# 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

```

- Starting with Oracle Solaris 11.2, an EFI label is applied automatically to the root pool disk or disks during installation on SPARC based systems with GPT enabled firmware (See [“Firmware, Disk Labeling, and EEPROM Changes” on page 114.](#)) 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. See [“Using the fdisk Option”](#) in [“Managing Devices in Oracle Solaris 11.2”](#).

For more general information about managing ZFS root pools, see [Chapter 4, “Managing ZFS Root Pool Components,”](#) in [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).

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 11/11:**
 - **SPARC:** The OpenBoot PROM (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, as follows:

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

- **SPARC and x86:** Attaching a root pool disk with `zpool attach` to create a mirrored root pool requires the following 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 being reattached, as shown in the following example:

```
# 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 [“Managing Devices in Oracle Solaris 11.2”](#).

- **x86:** GRUB Legacy and a root pool disk requires an SMI (VTOC) label.
- **x86:** If replacing a root pool disk with the `zpool replace` command, apply the boot blocks manually as follows:

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

- **x86:** The root pool disk must be less than 2 TBs.

- **Starting with 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, as shown in the following example:

```
# bootadm install-bootloader
```

- **SPARC:** Attaching a root pool disk with `zpool attach` to create a mirrored root pool requires the following 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 as follows:

```
# bootadm install-bootloader
```

- **x86:** Attaching a root pool disk with `zpool attach` to create a mirrored root pool requires the whole disk syntax, as this example shows:

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

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

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

Swap and Dump Device Configuration Changes

Swap space is the reserved area of a disk that the Oracle Solaris OS software and application software can use for temporary storage. Swap space is used as virtual memory storage areas when the system does not have enough physical memory to handle current processes that are running. In Oracle Solaris 10 a UFS root environment provides one disk slice for both swap and dump devices. In Oracle Solaris 11, two separate volumes are created as a swap device and a dump device. In a ZFS root file system, the disk space that is reserved for swap is a ZFS volume. Use the `dumpadm` command as follows to display this information:

```
# 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 as follows:

```
# 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 display swap space sizes in human-readable format, as shown in this 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
```

Managing ZFS swap and dump volumes differs from how you manage 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, as shown in the following example, but it might take some time to reinitialize the volume.

```
# 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 as follows:

```
# 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 -
```

For more information about swap space and dump device configuration, see [“About Swap Space”](#) in [“Managing File Systems in Oracle Solaris 11.2”](#).

Managing Storage Features

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

The following topics are covered:

- [“Comparing Solaris Volume Manager Configurations to ZFS Configurations” on page 51](#)
- [“ZFS Storage Pool Best Practices” on page 52](#)
- [“COMSTAR Replaces iSCSI Target Daemon” on page 55](#)

Comparing Solaris Volume Manager Configurations to ZFS Configurations

In Oracle Solaris 10, you can 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 of the file systems in the pool.

In Oracle Solaris 11, you can 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 are similar to RAID-5 configurations.

ZFS dynamically stripes data across all non-redundant, mirrored, and RAID-Z configurations. Note the following additional information:

- 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, as shown in the following 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
```

ZFS Storage Pool Best 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.

ZFS Storage Pool Creation Best Practices

- **Specific root pool device and boot disk requirements**

See these references:

- [“ZFS Root Pool Device Requirements” on page 45](#)
- [“ZFS Root Pool Disk and Boot Administration” on page 46](#)

- **General root pool creation best practices**

- You must create the root pool must 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. To expand a mirrored virtual device, use the `zpool attach` command.
- The root pool cannot have a separate log device.
- You can set pool properties during an installation with AI 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 result in 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 best 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**

You can create RAID-Z storage pools 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 as follows:

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

ZFS Storage Pool Monitoring Best Practices

Refer to the following best practices for monitoring ZFS storage pools:

- Make sure that pool usage is below 90% of pool capacity for best performance.
Note that the `zpool list` command does not count RAID-Z parity as used space, nor does it subtract it from the pool capacity. RAID-Z pool capacity might be below 90%, while being nearly full. Use the `zfs list pool` command to check. See [“Displaying ZFS File System Information” on page 60](#).
- Use the `zpool scrub` command 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 the `zpool status` command on a weekly basis to monitor pool and pool device status. Also use the `fmddump` or `fmddump -eV` command to determine if any device faults or errors have occurred.

Troubleshooting ZFS Storage Pool Issues

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 “Managing ZFS File Systems in Oracle Solaris 11.2”](#).
- **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. 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.

- **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. 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 these drivers fully support device IDs. However, you might consider exporting the pool before making any hardware changes.

See [Chapter 10, “Oracle Solaris ZFS Troubleshooting and Pool Recovery,”](#) in “[Managing ZFS File Systems in Oracle Solaris 11.2](#)”.

COMSTAR Replaces iSCSI Target Daemon

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

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.

Note - Although the iSCSI target in COMSTAR is a functional replacement for the iSCSI target daemon, no upgrade or update path exists for converting your iSCSI LUNs to COMSTAR LUNs.

Both the iSCSI target daemon and the `shareiscsi` property are not available in Oracle Solaris 11.

Use the following commands to manage iSCSI targets and LUNs:

- `itadm` – Manages SCSI targets.
- `srptadm` – Manages SCSI RDMA Protocol (SRP) target ports.
- `stmfadm` – Manages SCSI LUNs. Rather than setting a special iSCSI property on the ZFS volume, create the volume and use the `stmfadm` command to create the LUN.

See [Chapter 8, “Configuring Storage Devices With COMSTAR,”](#) in “[Managing Devices in Oracle Solaris 11.2](#)”.

Managing File Systems

This chapter provides information about managing file systems in Oracle Solaris 11 releases. The following topics are covered:

- [“File System Changes” on page 57](#)
- [“Managing ZFS File Systems” on page 59](#)
- [“Migrating File System Data to ZFS File Systems” on page 66](#)

File System Changes

The file systems in Oracle Solaris 11 are very similar to Oracle Solaris 10 file systems. The following table describes the file systems that are supported in this release.

TABLE 5-1 File Systems That Are Supported in Oracle Solaris 11

File System Type	Supported 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 Oracle Solaris 11.
- 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

The root file system hierarchy is almost identical to systems that are running Oracle Solaris 10 with 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 you install a system, the root of the Oracle 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.
- A separate `/var` file system is created automatically for a global zone and a non-global zone in Oracle Solaris 11.
- Starting with Oracle Solaris 11.1, a `rpool/VARSHARE` file system is mounted at `/var/share` by default. 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 previously listed for compatibility purposes. This file system generally requires no administration except to ensure that `/var` components do not fill up the root file system. During a system upgrade, 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 a 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. Also, you *cannot* use swap files in a ZFS root environment. See [“Swap and Dump Device Configuration Changes” on page 48](#).

Mounting File Systems

Review the following considerations when mounting file systems:

- 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 Oracle 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 Systems

The following ZFS file system features (not available in Oracle Solaris 10) are available in Oracle Solaris 11:

- **ZFS file system encryption** – You can encrypt a ZFS file system when it is created. 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 65.
- **ZFS file system sharing syntax changes** – Includes both NFS and SMB file system sharing changes. See [“ZFS File System Sharing Changes”](#) on page 63.
- **ZFS man page changes** – The `zfs.1m` man 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\(1M\)](#)
- **ZFS root pool setup simplified** – Support for Unified Archives in Oracle Solaris 11.2 makes root pool recovery setup much easier than in previous releases. See [“Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.2”](#).
- **ZFS send stream monitoring** – You can monitor the progress of a ZFS stream transmission in real time. See [“Monitoring the Progress of ZFS Send Streams”](#) in [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).
- **ZFS temporary pool names** – You can create or import a pool with a temporary pool name in a shared storage or recovery scenario. See [“Importing a Pool With a Temporary Name”](#) in [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).

Displaying ZFS File System Information

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

Display ZFS storage pool information with the `zpool status` command.

Display ZFS file system information with the `zfs list` command.

See [“Reviewing the Initial ZFS BE After an Installation” on page 77](#).

Resolving ZFS File System Space Reporting Issues

The `zpool list` and `zfs list` commands are more improved than the previous `df` and `du` commands for determining 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 descendant 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. 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. See the following examples for more information.

- 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, the `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, as shown in the following 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. See “ZFS File System Sharing Changes” on page 63.
- 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.
- You can export ZFS storage pools from an Oracle Solaris 10 system and then import them to an Oracle Solaris 11 system.

Monitoring File Systems

You can use the `fsstat` command to monitor file systems and report about file system operations. There are several options that report different kinds of activity. For example, you can display information by mount point or by file system type. In the following example, the `fsstat` command displays all of the ZFS file system operations from the time that the ZFS module was initially loaded:

```
$ fsstat zfs
new name name attr attr lookup rmdir read read write write
file remov chng get set ops ops ops bytes ops bytes
268K 145K 93.6K 28.0M 71.1K 186M 2.74M 12.9M 56.2G 1.61M 9.46G zfs
```

See [fsstat\(1M\)](#) for other examples.

Managing Memory Between ZFS and Applications

Starting with Oracle Solaris 11.2, a new `user_reserve_hint_pct` tunable parameter provides a *hint* to the system about application memory usage. This hint is used to limit the growth of ZFS Adaptive Replacement Cache (ARC) cache so that more memory can be made available for applications. For information about using this new parameter, see *Memory Management Between ZFS and Applications in Oracle Solaris 11.2* (Doc ID 1663862.1) at <https://support.oracle.com/>.

NFS nfsmapid Syntax Changes

The syntax for modifying the `nfsmapid` service that maps NFSv4 user and group IDs by using the `passwd` and `group` entries in the `/etc/nsswitch.conf` file has changed.

The `nfsmapid` service is as follows:

```
# svcs mapid
STATE          STIE    FMRI
online         Apr_25  svc:/network/nfs/mapid:default
```

You would modify the service instance as follows:

```
# svccfg -s svc:/network/nfs/mapid:default
svc:/network/nfs/mapid:default> listprop
nfs-props                application
nfs-props/nfsmapid_domain  astring    old.com
general                  framework
general/complete        astring
general/enabled          boolean    false
restarter                framework  NONPERSISTENT
restarter/logfile        astring    /var/svc/log/network-nfs-mapid:default.log
restarter/contract       count      137
restarter/start_pid      count      1325
restarter/start_method_timestamp  time      1366921047.240441000
restarter/start_method_waitstatus  integer   0
restarter/auxiliary_state  astring    dependencies_satisfied
restarter/next_state     astring    none
restarter/state          astring    online
restarter/state_timestamp  time      1366921047.247849000
general_ovr              framework  NONPERSISTENT
general_ovr/enabled      boolean    true
svc:/network/nfs/mapid:default> setprop nfs-props/nfsmapid_domain = new.com
svc:/network/nfs/mapid:default> listprop
nfs-props                application
nfs-props/nfsmapid_domain  astring    new.com
.
.
.
svc:/network/nfs/mapid:default> exit
# svcadm refresh svc:/network/nfs/mapid:default
```

ZFS File System Sharing Changes

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

In Oracle Solaris 11 11/11, the file sharing was enhanced and the command syntax changed. See [“Legacy ZFS Sharing Syntax”](#) in [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).

Starting with Oracle Solaris 11.1, ZFS file sharing includes the following additional enhancements:

- Share syntax is simplified. You share a file system by setting the new `share.nfs` or `share.smb` property as follows:

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

- Support for better inheritance of share properties to descendent file systems. In the previous example, the `share.nfs` property value is inherited to any descendent file systems. For example:

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

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

- You can also specify additional property values or modify existing property values on any existing file system shares as follows:

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

These additional file sharing improvements are associated with pool version 34. See [“New ZFS Sharing Syntax”](#) in [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).

ZFS Sharing Migration Issues

Review the following share transition issues:

- **Upgrading your Oracle Solaris 11 system to a later Oracle Solaris release** – 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 so that you can 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 the `unshareall` command unpublishes a share, but the command 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 message is displayed.

ZFS Data Deduplication Requirements

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 discouraged.

2. Make sure your system has enough memory to support dedup as follows:
 - 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 them 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 suggested data migration best practices when migrating data to systems that are running Oracle Solaris 11.

UFS to ZFS Data Migration Best 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

You can use the ZFS shadow migration tool to migrate data from an existing file system to a new file system. A *shadow* file system is created and that file system then 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 and then does the following:

- Creates an empty ZFS file system.
- Sets 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/ufsddata 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 [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).

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 [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).

Migrating UFS Data to a ZFS File System

You can also use the `ufsrestore` command to restore a previous `ufsdump` dump, as shown in the following example:

```
# mount -F nfs rsystem:/export/ufsddata /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, it is translated into NFSv4 ACLs. See [Chapter 7, “Using ACLs and Attributes to Protect Oracle Solaris ZFS Files,”](#) in [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).

Managing Software and Boot Environments

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

The following topics are covered:

- “Software Package Changes” on page 69
- “Oracle Solaris 10 SVR4 and IPS Package Comparison” on page 70
- “IPS Installation Package Groups” on page 72
- “Displaying Information About Software Packages” on page 72
- “Updating the Software on an Oracle Solaris System” on page 74
- “Managing Boot Environments” on page 76

Software 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 Software in Oracle Solaris 11.2”](#).

IPS includes a suite of pkg commands that enable you to list, search, install, update, and remove software packages. See Chapters 2-4 in [“Adding and Updating Software in Oracle Solaris 11.2”](#). IPS commands also enable you to manage package publishers and copy or create package repositories. See [Chapter 5, “Configuring Installed Images,”](#) in [“Adding and Updating Software in Oracle Solaris 11.2”](#).

Oracle Solaris 10 SVR4 and IPS Package Comparison

Review the following information about software packaging:

- 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 that 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 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 a specific SVR4 package as follows:

```
$ pkg info -g http://pkg.oracle.com/solaris/release/ SUNWcs1
Name: SUNWcs1
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: October 27, 2010 06:35:58 PM
  Size: 0.00 B
  FMRI: pkg://solaris/SUNWcs1@0.5.11,5.11-0.133:20101027T183558Z
```

The previous output shows that the SVR4 `SUNWcs1` package is renamed (Rename) the IPS `system/library` package. Determine if the IPS package is installed as follows:

```
$ pkg list system/library
NAME (PUBLISHER)                                VERSION                                IFO
system/library                                  5.12-5.12.0.0.0.42.1                 i--
```

The previous output indicates that the `system/library` package is already installed. If the package was not installed, you would install the package as follows:

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

- If an SVR4 package is available as an IPS package, install the IPS package rather than the SVR4 package. Installing the IPS package ensures that only versions that are compatible with the rest of the image can be installed and that dependencies are automatically checked and updated. See [“Adding and Updating Software in Oracle Solaris 11.2”](#).

In the previous example, even if you tried to install the SVR4 package, the `system/library` IPS package is automatically installed. However, in this example, since the package is already installed, the command returns the following message:

```
$ pkg install SUNWcsl
```

```
No updates necessary for this image.
```

- 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 package names use a Fault Manager Resource Identifier (FMRI) naming style. Package names are also hierarchical instead of abbreviated. To reiterate, 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 the following:

```
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 Software in Oracle Solaris 11.2”](#).

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 10 packages are split into development, documentation, and runtime components. In Oracle Solaris 11, all of these components are delivered in one package. You can use the `pkg change-facet` command to exclude certain components such as man pages or header files. See [“Controlling Installation of Optional Components”](#) in [“Adding and Updating Software in Oracle Solaris 11.2”](#).
- 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 by using zones and branded zones. See [“Oracle Solaris Zone Features”](#) on page 143.

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
<code>pkgadd</code>	<code>pkg install</code>
<code>patchadd</code>	<code>pkg update</code>
<code>pkgrm</code>	<code>pkg uninstall</code>
<code>pkgadm addcert, pkgadm removecert</code>	<code>pkg set-publisher -k, -c, --approve-ca-cert, --revoke-ca-cert, unset-ca-cert</code>

SVR4 Package Command	IPS Package Command Equivalent
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 four group packages that are appropriate for a large server, a small server or non-global zone, a minimal server, and a graphical desktop environment. Each of these group packages installs different sets of packages on a system. See [“Oracle Solaris 11.2 Package Group Lists”](#).

Display package group information as follows:

```
$ pkg list -as '*group/system/solaris*'
```

Display the contents of package groups as follows:

```
$ pkg contents -ro fmri -t depend -a type=group group-package-name
```

See [“Listing All Installable Packages in a Group Package”](#) in [“Adding and Updating Software in Oracle Solaris 11.2”](#).

Determine which package group is currently installed on your system as follows:

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

IPS also includes other meta and group packages that you can install 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 SUNWCa11 package cluster, consider installing the `group/system/solaris-large-server` package group.

For a complete listing of the packages that are part of each package group, see [“Oracle Solaris 11.2 Package Group Lists”](#).

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
Branch: 0.133
Packaging Date: Wed Oct 27 18:31:05 2010
Size: 0.00 B
FMRI: pkg://solaris/amp@0.5.11-0.133:20101027T183105Z

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: 5.12
Branch: 5.12.0.0.0.48.0
Packaging Date: Mon May 19 05:51:22 2014
Size: 5.46 kB
FMRI: pkg://solaris/group/feature/amp@5.12-5.12.0.0.0.48.0:20140519T055122Z

Name: web/amp
Summary:
State: Not installed (Renamed)
Renamed to: group/feature/amp@0.5.11-0.174.0.0.0.0
             consolidation/ips/ips-incorporation
Publisher: solaris
Version: 0.5.11
Branch: 0.174.0.0.0.0
Packaging Date: Wed Sep 21 19:15:02 2011
Size: 5.45 kB
FMRI: pkg://solaris/web/amp@0.5.11-0.174.0.0.0.0:20110921T191502Z

```

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@24.3-5.12.0.0.0.42.0

$ pkg search file::emacs
INDEX      ACTION VALUE      PACKAGE
basename  file   usr/bin/emacs pkg:/editor/gnu-emacs@24.3-5.12.0.0.0.42.0
```

Updating the Software on an Oracle Solaris 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 package dependencies or image policy. If a package is constrained, a message indicating what constraint is preventing the installation or update is provided. Package constraints generally represent a dependency or a version issue. For some package installation or updates, clone and backup BE creation is separate. If a clone is created, the changes are made in the clone and the current BE is not touched. If a backup BE is created, the changes are made in the current BE. To see the changes, you must reboot the system. If you are not satisfied with the changes, you can reboot to the backup BE. The use of `pkg` options and image policy settings enable you to specify a new or backup BE.

The following options are available:

- **Adding software packages after an installation** – To add packages, use the `pkg install` command. See [Chapter 3, “Installing and Updating Software Packages,”](#) in [“Adding and Updating Software in Oracle Solaris 11.2”](#).

For instructions on adding packages that include the Oracle Solaris Desktop (GNOME 2.30) after an installation, see [“Adding Software After a Text Installation”](#) in [“Installing Oracle Solaris 11.2 Systems”](#).

- **Updating all of the packages on your installed system** – Update all of the packages on your system that have available updates as follows:

```
# pkg update entire
```

Depending on your local package repository or publisher status, your system could be updated automatically from one Oracle Solaris 11 release to a subsequent Oracle Solaris 11 release. For more information about controlling a system upgrade, see [Chapter 4, “Updating or Upgrading an Oracle Solaris Image,”](#) in [“Adding and Updating Software in Oracle Solaris 11.2”](#).

For an example of how to use this command to update a BE, see [“Managing Boot Environments”](#) on page 76.

Display the list of installed packages that have updates available as follows:

```
# pkg list -u
```

- **Install package updates that deliver fixes** – Apply Support Repository Updates (SRUs) as needed. SRUs occur on a regular basis and take the place of maintenance updates or patch bundles that are used in Oracle Solaris 10.

Installing Maintenance Updates on an Oracle Solaris 11 System

If you have an active Oracle support plan, you 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 they occur on a regular basis. SRUs take the place of maintenance updates or patch bundles that are available for Oracle Solaris 10 releases. Subsequent Oracle Solaris 11 releases are made available in the support repository or a release repository that provides the currently available OS. See [“How to Configure the Oracle Solaris Support Repository” on page 75](#).

If you need to access an IPS repository on a system that has zones installed by using `https_proxy` and `http_proxy`, see [“Proxy Configuration on a System That Has Installed Zones” in “Creating and Using Oracle Solaris Zones”](#).

For more information about copying and creating package repositories, see [“Copying and Creating Package Repositories in Oracle Solaris 11.2”](#).

For more information about choosing the best way to update your system images, see [Chapter 4, “Updating or Upgrading an Oracle Solaris Image,” in “Adding and Updating Software in Oracle Solaris 11.2”](#).

▼ How to Configure the Oracle Solaris Support Repository

1. **Log into the following site.**
<http://pkg-register.oracle.com/>
2. **Select Oracle Solaris 11 Support option from the Product list, then click the Submit button to accept the license agreement.**
3. **To download the SSL key and certificate, follow the directions on the download page.**
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 \  
-g https://pkg.oracle.com/solaris/support solaris
```

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

See [Chapter 4, “Updating or Upgrading an Oracle Solaris Image,”](#) in [“Adding and Updating Software in Oracle Solaris 11.2”](#) for instructions.

This operation updates the packages on the system with the latest package versions by creating a new BE.

Managing Boot Environments

Boot environments (BEs) are bootable instances of an image. Previously, you could perform a live upgrade or use the `patchadd` command to update a BE. In Oracle Solaris 11, you use the `pkg update` command to update a BE, or you can use the `beadm` command to create, list, and remove BEs.

Tools for Managing Boot Environments

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. However, the command is not guaranteed to create a new or backup BE in every instance. Use the appropriate `pkg update` command options to specify a desired outcome. Also, new and backup BEs behave differently. For new BEs, updates are made in the new BE. Whereas, if a backup BE is created, updates are made in the current BE.

TABLE 6-2 Comparing Boot Environment Command Syntax

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

See [“Creating and Administering Oracle Solaris 11.2 Boot Environments”](#) and [beadm\(1M\)](#).

In most cases, the `pkg update` command, when used with no operands, performs the following actions:

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 option the next time the system is booted. The current BE remains as an alternate boot option.

Reviewing the Initial ZFS BE After an Installation

After performing a fresh default Oracle Solaris installation, 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/share` file system (starting with Oracle Solaris 11.1). See [“Root File System Requirements” on page 58](#).
- `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. In most cases, a clone or backup BE is created and automatically activated. As a best practice, use the `pkg update -nv` command first to determine whether a backup BE or a new BE will be created and which packages will be updated. New and backup BEs are also activated and updated differently. A

new BE is automatically activated by default, but a backup BE is not automatically activated. Also, a new BE is updated, but a backup BE is not updated.

Before You Begin



Caution - When you update a BE, you most likely will need to upgrade your root pool version. If an upgrade for the current pool version is available for the recent update, you cannot boot back to a previous BE if the previous BE is in a lower pool version. Make sure you have tested all of the features and that you are satisfied with the current update before upgrading your pool version.

For information about upgrading your pool version, see [“Upgrading ZFS Storage Pools” in “Managing ZFS File Systems in Oracle Solaris 11.2”](#).

1. Display the existing BE information for your system.

```
# beadm list
```

2. Update the BE.

```
# pkg update
```

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

3. Reboot the system, then, confirm the BE status.

```
# init 6
.
.
.
# beadm list
```

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

```
# beadm activate previousBE
# init 6
```

If the activated BE does not boot, see [“How to Boot From a Backup BE for Recovery Purposes” on page 117](#).

Managing Network Configuration

This chapter provides basic information about managing network configuration in Oracle Solaris 11 releases.

The following topics are covered:

- “Network Administration Features” on page 79
- “Network Virtualization and Advanced Networking Features” on page 82
- “Comparing the Oracle Solaris 10 Network Protocol Stack to the Oracle Solaris 11 Network Protocol Stack” on page 83
- “Network Administration Command Changes” on page 86
- “Configuring the Network in Oracle Solaris 11” on page 94

Network Administration Features

How you configure the network in Oracle Solaris 11 is not the same as in Oracle Solaris 10. For more detailed information about network administration changes in this release, see [Chapter 1, About Transitioning From Oracle Solaris 10 to an Oracle Solaris 11 Release](#).

The following network administration features (listed alphabetically) are new or have changed:

- **Datalink naming** – Oracle Solaris 11 supports generic naming of datalinks. Generic names are automatically assigned to each datalink on a system by using the `net0`, `net1`, `netN` naming convention, depending on the total number of network devices that are on the system.
- **DHCP support** – In addition to the legacy Sun DHCP product, Oracle Solaris 11 supports the Internet Systems Consortium (ISC) DHCP server. This software is not automatically installed on your system. See “[Administering DHCP](#)” on page 100.
ISC DHCP support includes new SMF services, new administrative commands, and new configuration files. For details, see “[ISC DHCP Server](#)” in “[Working With DHCP in Oracle Solaris 11.2](#)”.
- **IP interface and address configuration** – You use the `ipadm` command to manage network configuration at the IP layer (L3) layer of the network protocol stack. The command configures IP interfaces and addresses, as well as other L3 entities, for example,

IP network multipathing (IPMP). The `ipadm` command replaces the `ifconfig` command that is used in Oracle Solaris 10.

The `ipadm` command provides almost equivalent functionality to the `ifconfig` command for configuring IP interfaces and addresses, with the exception that in Oracle Solaris 11, the `ipadm` command is solely used for IP administration. Also, unlike the `ifconfig` command, changes that you make with the `ipadm` command persist across system reboots. Note that you can still use the `ifconfig` in some cases. See [“Comparing the `ifconfig` Command to the `ipadm` Command” on page 87](#).

- **IPMP changes**– IPMP has a new conceptual model and different commands for managing IPMP configuration. One significant change is that IP interfaces are 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 about these changes, see [“How IPMP Works” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2”](#).

Oracle Solaris 11 also uses different commands for managing IPMP configuration. As a result, some configuration tasks are also performed differently. See [Chapter 3, “Administering IPMP,” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2”](#).

- **IP tunnel administration** – IP tunnel administration has changed to be more consistent with datalink administration in Oracle Solaris 11. You create and configure IP tunnels by using the `dladm` command. Tunnels can also use other datalink features that are supported in this release, for example, the ability to assign tunnels more meaningful names. See [Chapter 4, “About IP Tunnel Administration,” in “Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2”](#).
- **Naming and directory services configuration** – This configuration is managed through SMF and not by editing various files within the `/etc` directory. See [“Configuring Naming and Directory Services” on page 99](#).
- **Network configuration during installation with AI** – Starting with Oracle Solaris 11.2, the `svc:/network/install:default` SMF service includes two new property group types: `ipv4_interface` and `ipv6_interface`. You can create SC profiles that contain property groups with the type `ipv4_interface` and `ipv6_interface`. The `svc:/network/install:default` start method consumes properties of these types and then uses them to configure network interfaces upon the first system boot after an installation. SC profiles can include an unlimited number of property groups of these types, which enable an administrator to configure multiple network interfaces during installation.

Note that the existing `install_ipv4_interface` and `install_ipv6_interface` property groups for this service continue to be supported. For instructions, see [“Configuring Network Interfaces” in “Installing Oracle Solaris 11.2 Systems”](#).

- **Network diagnostics tools** – You can use the Fault Manager (`fmd`) transport module (`network-monitor`) to perform network diagnostics and monitor network resources. The utility reports conditions that might lead to degraded network functionality. See [Chapter 4](#),

[“Performing Network Diagnostics With the network-monitor Transport Module Utility,”](#) in [“Troubleshooting Network Administration Issues in Oracle Solaris 11.2 ”](#).

- **Network modes implementation** – Oracle Solaris 11 supports two modes for network configuration: *fixed* and *reactive*. See [“About Network Configuration Modes”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2 ”](#) for more details.
- **Network monitoring tools** – There are two new commands for observing networking traffic in this release: `tcpstat` and `ipstat`. These commands provide information about the network traffic on a server. See [“Observing Network Traffic With the ipstat and tcpstat Commands”](#) in [“Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”](#).
- **Network packet analysis tools** – Similar to the `snoop` command, you can use the Wireshark GUI or its command line equivalent, TShark, to troubleshoot networking issues and perform packet analysis. See [“Analyzing Network Traffic With the TShark and Wireshark Analysers”](#) in [“Administering TCP/IP Networks, IPMP, and IP Tunnels in Oracle Solaris 11.2 ”](#).
- **Profile-based network configuration** – The use of profiles enables you to define multiple alternative configurations, each identified by a single profile (referred to as a network configuration profile (NCP)). For example, you could create a profile named `office` for a notebook PC that configures the system with static IP addresses and DNS server locations. An alternate home profile might use DHCP to acquire this information. Two additional commands are used for administering profiles in this release: `netcfg` and `netadm`. See [“Network Administration Command Changes”](#) on page 86 for details.
- **Routing configuration** – Use the `route` command to configure a persistent route for a system, default or otherwise. The `route` command replaces the prior method of managing routes through the `/etc/defaultrouter` file. This file is deprecated in Oracle Solaris 11. Furthermore, after an installation you cannot determine a system's default route by checking the `/etc/defaultrouter` file. To determine a system's default route after an installation, use the `route -p show` command or the `netstat -nr` command. See [“Configuring Persistent Routes”](#) on page 98.
- **Tunables (network parameters) configuration** – The `ipadm` and `dladm` commands also replace the `ndd` command for configuring certain network parameters in this release. See [“Comparing the ndd Command to the ipadm Command”](#) on page 91, [“Comparing the ndd Command and driver.conf Configuration to the dladm Command”](#) on page 92, and Chapter 5, [“Internet Protocol Suite Tunable Parameters,”](#) in [“Oracle Solaris 11.2 Tunable Parameters Reference Manual ”](#).

Network Virtualization and Advanced Networking Features

Oracle Solaris 11 supports several network virtualization and advanced networking features. For a detailed description of these newer features, see [“Key Oracle Solaris Network Administration Features”](#) in [“Strategies for Network Administration in Oracle Solaris 11.2”](#).

The following features are new in Oracle Solaris 11.2. Many of these enhancements are for features that were introduced in Oracle Solaris 11 11/11:

- **Communication between VNICs by using an external switch** – By using the Oracle Solaris reflective relay feature, you can force traffic between local Oracle Solaris zones or Oracle VMs that share the same underlying physical NIC to always be sent to the physical network instead of the host virtual switch. See [“Controlling Switching Between VMs Over the Same Physical Port”](#) in [“Managing Network Virtualization and Network Resources in Oracle Solaris 11.2”](#).
- **Displaying multiple MAC addresses that are associated with VNICs** – Multiple MAC addresses are associated with system-created VNICs in Oracle VM Server for SPARC and anet resources in Oracle Solaris Kernel Zones. Starting with Oracle Solaris 11.2, you can use the `dladm show-vnic` command to display multiple MAC addresses that are associated with VNICs. See [“Displaying VNICs With Multiple MAC Addresses”](#) in [“Managing Network Virtualization and Network Resources in Oracle Solaris 11.2”](#) and [Oracle VM Server for SPARC 3.1 Administration Guide \(\[http://docs.oracle.com/cd/E38405_01/html/E38406/index.html\]\(http://docs.oracle.com/cd/E38405_01/html/E38406/index.html\)\)](#).
- **Elastic Virtual Switch (EVS) feature** – EVS is an L2 technology that expands network virtualization capabilities by enabling you to manage virtual switches across multiple hosts. With the Oracle Solaris EVS feature, you can deploy virtual networks that span multiple hosts within either a multi-tenant cloud environment or datacenter. See [Chapter 6, “Administering Elastic Virtual Switches,”](#) in [“Managing Network Virtualization and Network Resources in Oracle Solaris 11.2”](#).
- **Probe-based datalink multipathing (DLMP)** – This feature enhancement for DLMP (introduced in Oracle Solaris 11 11/11) detects the loss of connectivity between DLMP aggregated links and configured targets. This type of failure detection addresses the limitations of the link-based failure detection mechanism, which can only detect failures caused by the loss of direct connection between the datalink and the first-hop switch. See [“Configuring Probe-Based Failure Detection for DLMP Aggregation”](#) in [“Managing Network Datalinks in Oracle Solaris 11.2”](#).
- **Single root I/O virtualization (SR-IOV)** – Oracle Solaris 11.2 includes the capability for managing network devices that support SR-IOV. See [“Using Single Root I/O Virtualization With VNICs”](#) in [“Managing Network Virtualization and Network Resources in Oracle Solaris 11.2”](#).
- **Virtual eXtensible area networks (VXLANS)** – In addition to the VLAN support that was introduced in Oracle Solaris 11 11/11, VXLANS are now also supported. A VXLAN is an L2 and L3 technology that works by overlaying a datalink (L2) network on top of an IP (L3) network. VXLANS address the 4K limitation that is imposed when using VLANs. Typically, VXLANS are used in a cloud infrastructure to isolate multiple virtual networks.

You can manage VXLANs by using the EVS feature. See [Chapter 3, “Configuring Virtual Networks by Using Virtual Extensible Local Area Networks,”](#) in “Managing Network Virtualization and Network Resources in Oracle Solaris 11.2”.

- **Layer 3 Virtual Router Redundancy Protocol (VRRP)** –Oracle Solaris 11 supports both L2 and L3 VRRP. New in Oracle Solaris 11.2, the proprietary L3 VRRP feature provides high availability of IP addresses, such as those that are used for routers and load balancers. L3 VRRP removes the need to configure unique VRRP virtual MAC addresses for VRRP routers, thereby providing better support for VRRP over IPMP, InfiniBand interfaces, and zones. See [Chapter 3, “Using Virtual Router Redundancy Protocol,”](#) in “Configuring an Oracle Solaris 11.2 System as a Router or a Load Balancer”.

See also “What’s New in Managing Network Virtualization and Network Resources in Oracle Solaris 11.2” in “Managing Network Virtualization and Network Resources in Oracle Solaris 11.2”.

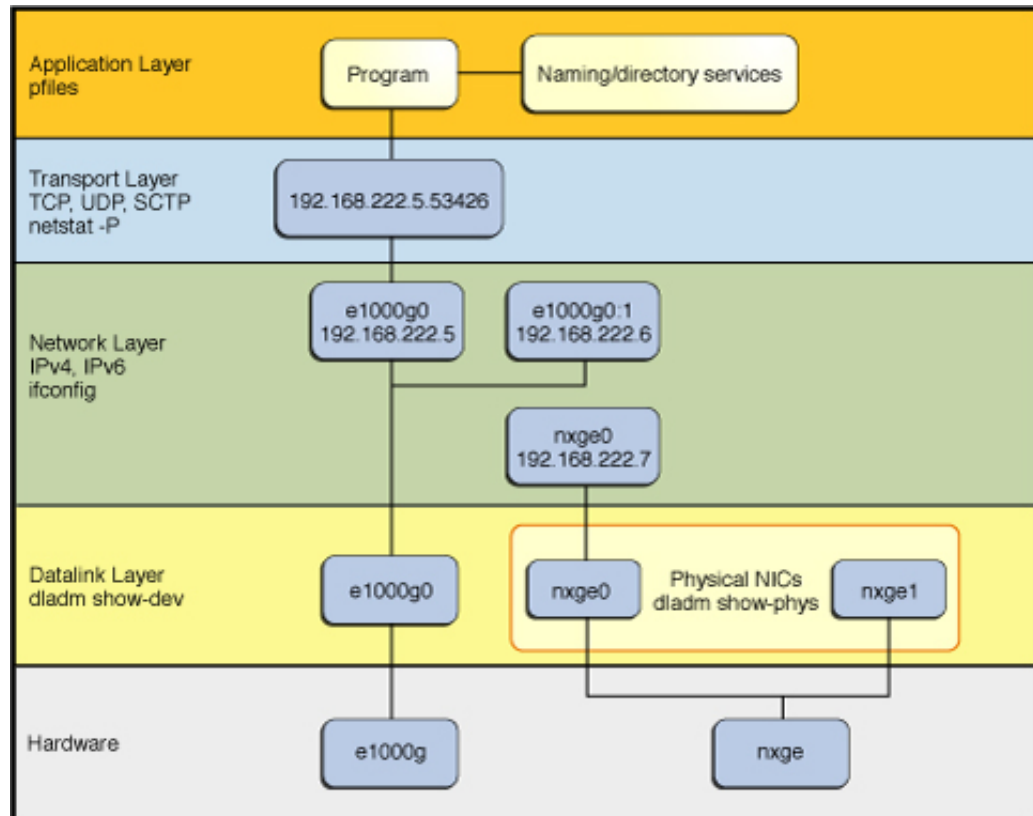
For information about Oracle VM, including Oracle VM Server for x86, Oracle VM Server for SPARC (previously called Sun Logical Domains, or LDoms), and Oracle VM Manager, see the documentation at <http://www.oracle.com/technetwork/documentation/vm-sparc-194287.html>.

Oracle also provides the Oracle Enterprise Manager Ops Center for managing some aspects of network virtualization, for example, the ability to create virtual private networks inside a virtual datacenter. For more information about Oracle Enterprise Manager Ops Center, refer to the *Certified systems Matrix* document at <http://www.oracle.com/pls/topic/lookup?ctx=oc122>.

Comparing the Oracle Solaris 10 Network Protocol Stack to the Oracle Solaris 11 Network Protocol Stack

In previous Oracle Solaris implementations of the network protocol stack, interfaces and links on the software layer were built on the devices in the hardware layer. More specifically, a hardware device instance in the hardware layer had a corresponding link on the datalink layer and a configured interface on the interface layer. This one-to-one relationship amongst the network device, its datalink, and its IP interface is illustrated in the following figure.

FIGURE 7-1 Oracle Solaris 10 Network Protocol Stack With Network Devices, Links, and Interfaces Displayed



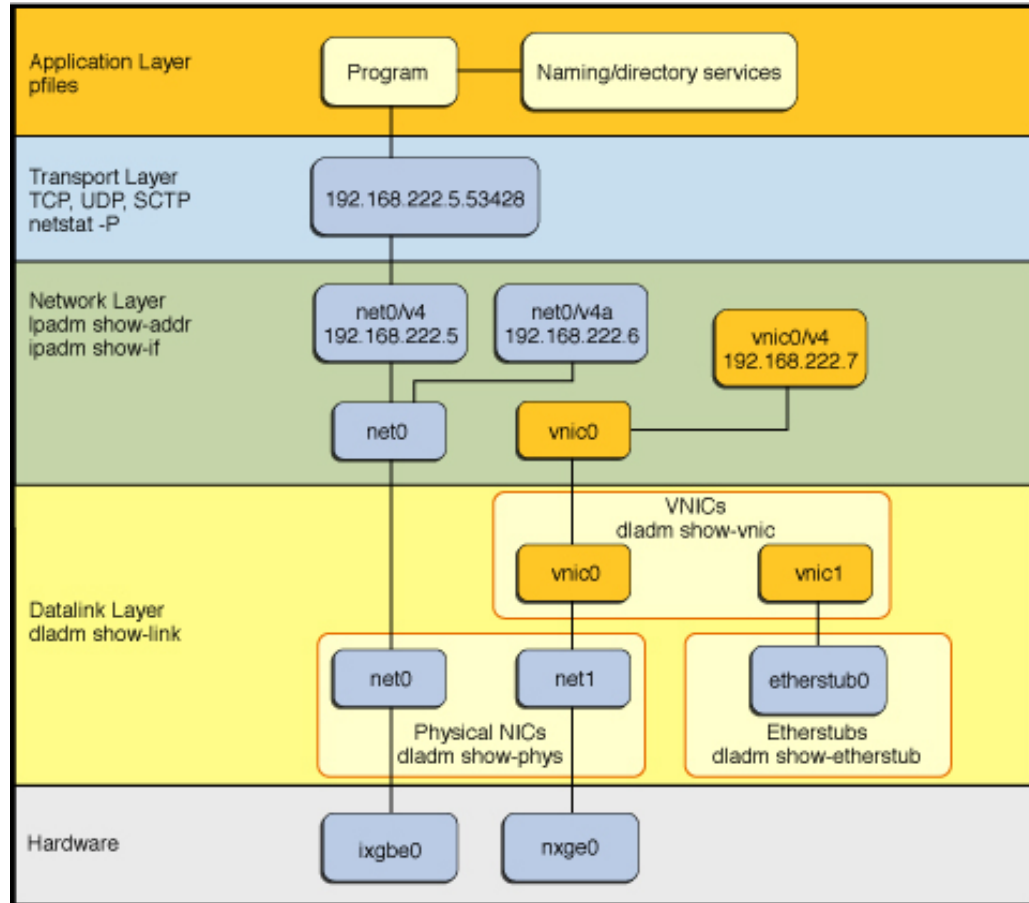
The Oracle Solaris 10 implementation has the following limitations:

- The one-to-one relationship that binds the device, the datalink, and the interface means that the network configuration is dependent on the hardware configuration and also the network topology. Thus, interfaces must be reconfigured if changes are implemented in the hardware layer, such as replacing the NIC or changing the network topology.
- There is limited support for virtual devices at the datalink layer. Only link aggregations are supported in Oracle Solaris 10.
- The `ifconfig` command manages logical interface names, where each logical interface corresponds to an IP address on the interface. It is not always obvious which managed features apply to the interface and which apply to individual addresses.

In Oracle Solaris 11, the one-to-one relationship between the hardware, datalink, and interface layers remains, as shown in the following figure. However, the software layer is decoupled

from the hardware layer. With this separation, network configuration on the software layer is no longer bound to the chipset or the network topology in the hardware layer.

FIGURE 7-2 Oracle Solaris 11 Network Protocol Stack With Devices, Links, and Interfaces Displayed



The changes that are implemented in Oracle Solaris 11 make network administration more flexible in the following ways:

- The network configuration is insulated from any changes that might occur in the hardware layer. Link and interface configurations are preserved even if the underlying hardware is removed. These same configurations can then be reapplied to any replacement NIC, provided that the two NICs are of the same type.
- The separation of the network configuration from the network hardware configuration also enables the use of customized link names in the datalink layer.

- With the abstraction of the datalink layer, multiple networking abstractions or configurations, such as virtual local area networks (VLANs), virtual network interface cards (VNICs), physical devices, link aggregations, and IP tunnels are unified into a common administrative entity, which is the datalink.

For more information about how networking features are administered within the Oracle Solaris network protocol stack, see [“Network Administration Within the Oracle Solaris Network Protocol Stack”](#) in [“Strategies for Network Administration in Oracle Solaris 11.2”](#).

Network Administration Command Changes

In Oracle Solaris 10 and previous releases, the `ifconfig` command is the customary tool that you use to configure network interfaces. However, the command does not implement persistent configuration. Over time, the `ifconfig` command has undergone enhancements to add further network administration capabilities. As a result, the command has become more complex and can sometimes be confusing to use.

Another issue with IP interface configuration and administration is the absence of simple tools to administer TCP/IP properties, also referred to as tunables. The `ndd` command has long been the prescribed customization tool for this purpose, but like the `ifconfig` command, the `ndd` command does not implement persistent configuration. Previously, persistent configuration could be simulated for a network scenario by editing the boot scripts. With the introduction of the service management facility (SMF), using these types of workarounds can be risky because of the complexities of managing the various SMF dependencies, particularly in light of upgrades to an Oracle Solaris installation.

Note the following key points about the network administration commands that you use in this release:

- The `ipadm` and `dladm` commands replace the `ifconfig` command for configuring network interfaces (datalinks and IP interfaces and addresses). Although the `ifconfig` command is still functional, it primarily exists for backward compatibility. Also, the previous method of adding information to the `/etc/hostname*` files is deprecated in Oracle Solaris 11.

You can perform most of the tasks that you previously performed with the `ifconfig` command by using either the `dladm` command (for datalink administration) or the `ipadm` command (for IP administration). Although many `ifconfig` command options have an `ipadm` equivalent, there is not an exact one-to-one mapping between the two commands. For comparable equivalents, see [“Comparing the `ifconfig` Command to the `ipadm` Command”](#) on page 87.

- The `ipadm` and `dladm` commands also replace the `ndd` command as a tool for customizing network parameters (*tunables*). Although the `ndd` command is still functional in Oracle Solaris 11, the `ipadm` and `dladm` commands are preferred.
- In Oracle Solaris 10, you configure drivers through driver-specific mechanisms, for example, the `ndd` command and the `driver.conf` file. However, in Oracle Solaris 11, you

configure common driver features by setting `dladm` properties, as well as some driver-private features through driver-private properties.

Note - Some `ndd` options do not have equivalent `dladm` command options.

Comparing the `ifconfig` Command to the `ipadm` Command

Compared to the `ifconfig` command, the `ipadm` command provides the following advantages:

- Parameter interactions with interfaces and addresses that are clearly represented.
- Configuration commands that manage both the current system state, as well as keep a persistent record of that state synchronized for automatic use upon reboot.
- A committed, parsable output format with many subcommands that exist for easy use by shell scripts.
- User-defined IP address object names that provide a means for management scripts to easily reference individual addresses, including IP addresses that are defined through DHCP or IPv6 address autoconfiguration.

The following table compares selected `ifconfig` command options to `ipadm` command equivalent. The tables does not provide a comprehensive list of all of the available options. See [ipadm\(1M\)](#).

TABLE 7-1 `ifconfig` and `ipadm` Command Comparison

Task Description	<code>ifconfig</code> Command	<code>ipadm</code> Command
List all interfaces and their addresses.	<code>ifconfig -a</code>	<code>ipadm</code>
Create or delete an IP interface.	<code>plumb</code> <code>unplumb</code>	<code>ipadm create-ip</code> <code>ipadm delete-ip</code>
Create or delete a static IP address on an interface.	<code>[address[/prefix-length]</code> <code>[dest-address]] [addif</code> <code>address[/prefix-length]]</code> <code>[removeif address[/prefix-</code> <code>length]][netmask mask]</code> <code>[destination dest-</code> <code>address]</code>	<code>ipadm create-addr -a address</code> <code>ipadm delete-addr</code>
Create or delete a DHCP address on an interface.	<code>{auto-dhcp dhcp} [wait</code> <code>seconds] start release</code>	<code>ipadm create-addr -T dhcp [-w seconds]</code> <code>ipadm delete-addr -r</code>

Task Description	ifconfig Command	ipadm Command
Extend a DHCP lease.	{auto-dhcp dhcp} extend	ipadm refresh-addr
Obtain configuration parameters from DHCP without obtaining a lease.	{auto-dhcp dhcp} inform	ipadm refresh-addr -i
Check if DHCP is in use on an interface.	{auto-dhcp dhcp} ping	ipadm show-addr <i>interface</i>
Display DHCP status.	{auto-dhcp dhcp} status	netstat -D
Create or delete an auto-configured IPv6 address on an existing interface	inet6 plumb up unplumb	ipadm create-addr -T addrconf ipadm delete-addr
View/set address properties.	[deprecated -deprecated] [preferred -preferred] [private -private] [zone zonename -zones -all-zones] [xmit -xmit]	ipadm show-addrprop ipadm set-addrprop
Bring an address up.	up	ipadm up-addr Implicit in create-addr Required for explicit down-addr
Take an address down.	down	ipadm down-addr
View/set interface properties.	[metric <i>n</i>] [mtu <i>n</i>] [nud -nud] [arp -arp] [usesrc [<i>name</i> none] [router router]	ipadm show-ifprop ipadm set-ifprop
Create/delete an IPMP group.	plumb ipmp group [<i>name</i> ""] unplumb	ipadm create-ipmp ipadm delete-ipmp
Add an interface to an IPMP group.	group [<i>name</i>]	ipadm add-ipmp -i <i>ifname</i>
Turn on/off standby flag.	standby -standby	ipadm set-ifprop -p standby=on ipadm set-ifprop -p standby=off
Configure an IP tunnel link.	[<i>tdstunnel-dest-addr</i>] [<i>tsrc tunnel-srsc-addr</i>] [encaplimit <i>n</i> -encaplimit] [thoplimit <i>n</i>]	dladm *-iptun set of commands.

Task Description	ifconfig Command	ipadm Command
View/set the hardware address of a link.	[ether <i>address</i>]	dladm show-linkprop -p mac-address dladm set-linkprop -p mac-address= <i>addr</i>
View/set modules to be autopushed on a link.	[modlist] [modinsert <i>mod_name@pos</i>] [modremove <i>mod_name@pos</i>]	dladm show-linkprop -p autopush dladm set-linkprop -p autopush= <i>modlist</i>
Set subnet, netmask, broadcast domain.	subnet <i>subnet-address</i>] [broadcast <i>broadcast-address</i>]	ipadm set-addrprop -p prefixlen= <i>len</i>
Set IPsec policy for a tunnel link.	[auth_algs <i>authentication-algorithm</i>] [encr_algs <i>encryption-algorithm</i>] [encr_auth_algs <i>encryption-authentication-algorithm</i>]	ipseccconf See ipseccconf(1M)
Miscellaneous networking commands that have no ipadm command equivalent.	[auth_revarp] [<i>index if-index</i>] [<i>token address/prefix-length</i>] DHCP 'drop' option E	Not applicable

ifconfig Replacement Commands

In Oracle Solaris 11, there is no single command that replaces the information that is displayed in the output of the `ifconfig -a` command. However, in most cases, using the `ipadm` command without any options provides very similar information.

To determine which command to use as a replacement for the `ifconfig` command, refer to the following information:

- Use the `ipadm` command without any options to display basic information about a system's interfaces:

```
# ipadm
NAME          CLASS/TYPE STATE   UNDER  ADDR
lo0           loopback  ok      --      --
  lo0/v4      static    ok      --      127.0.0.1/8
  lo0/v6      static    ok      --      ::1/128
net0          ip        ok      --      --
  net0/v4     dhcp     ok      --      10.134.64.65/24
  net0/v6     addrconf ok      --      fe80::214:4fff:fefb:bbf0/10
```

- For MAC address information, use the `dladm` command with the following options:

- ```
dladm show-linkprop -p mac-address -o link,effective
```

  - Display detailed IP interface state or property information as follows:
- ```
# ipadm show-if -o ifname,class,state,current,over
# ipadm show-ifprop -o ifname,property,proto,current
```

 - Display detailed IP address state or property information as follows:
- ```
ipadm show-addr -o addrobj,type,state,current,addr
ipadm show-addrprop -o addrobj,property,current
```

  - Display IP tunnel configuration details as follows:
- ```
# dladm show-iptun
```

 - The following are situations in which you might still opt to use the `ifconfig` command:
 - To display the logical interface number for any given address or a link index number. The `ipadm` does not display this information and some applications still use these numbers.
 - As a diagnostic tool, the `ifconfig` command can provide additional information that you might not be able to obtain by using the `dladm` and `ipadm` commands.

The following two examples compare differences between the output of the `ifconfig` and the output of the `ipadm` command when used to obtain similar information about a system's datalink (`net0`).

```
# ifconfig net0
net0: flags=100001000942<BROADCAST,RUNNING,PROMISC,MULTICAST,IPv4,PHYSRUNNING> mtu 1500
index 4
    inet 0.0.0.0 netmask 0
    ether 0:d0:b7:b9:a5:8c
# ifconfig net0 inet6
net0: flags=120002000940<RUNNING,PROMISC,MULTICAST,IPv6,PHYSRUNNING> mtu 1500 index 4
    inet6 ::/10
```

```
# ipadm show-if -o ifname,class,state,current,over net0
IFNAME    CLASS    STATE    CURRENT    OVER
net0      ip       down    bm46----- --
sekon# ipadm show-ifprop -o ifname,property,proto,current net0
IFNAME    PROPERTY    PROTO    CURRENT
net0      arp         ipv4     on
net0      forwarding  ipv4     off
net0      metric      ipv4     0
net0      mtu         ipv4     1500
net0      exchange_routes  ipv4     on
net0      usesrc      ipv4     none
net0      forwarding  ipv6     off
net0      metric      ipv6     0
```

```

net0      mtu          ipv6 1500
net0      nud           ipv6 on
net0      exchange_routes ipv6 on
net0      usesrc      ipv6 none
net0      group       ip    --
net0      standby    ip    off

```

Comparing the ndd Command to the ipadm Command

Compared to the ndd command, the ipadm command provides the following advantages:

- Provides information about each TCP/IP property, such as a property's current and default value, as well as the range of possible values. Thus, debugging information is more easily obtained.
- Follows a consistent command syntax and is therefore easier to use.
- Persistent configuration of routing and transport layer tunables that is made possible by using an ipadm subcommand rather than the previously required uncommitted ndd commands that required the use of custom SMF scripts or /etc/rc*.d scripts.

The following table compares selected ndd command options to equivalent ipadm command options. See the [ipadm\(1M\)](#) man page for a more comprehensive list of command options.

TABLE 7-2 ndd Command Compared to the ipadm Command

ndd Command	ipadm Command
<pre> bash-3.2# ndd -get /dev/ ip ? ip_def_ttl (read and write) ip6_def_hops (read and write) ip_forward_directed_ broadcasts (read and write) ip_forwarding (read and write) </pre>	<pre> bash-3.2# ipadm show-prop ip PROTO PROPERTY PERM CURRENT PERSISTENT DEFAULT POSSIBLE ipv4 forwarding rw off -- off on, off ipv4 ttl rw 255 -- 255 1- 255 ipv6 forwarding rw off -- off on, off ipv6 hoplimit rw 255 -- 255 1- 255 ... </pre>
<pre> bash-3.2# ndd -get /dev/ip \ ip_def_ttl 100 </pre>	<pre> bash-3.2# ipadm show-prop -p ttl,hoplimit ip PROTO PROPERTY PERM CURRENT PERSISTENT DEFAULT POSSIBLE ipv4 ttl rw 255 -- 255 1- 255 </pre>

ndd Command	ipadm Command
<pre>bash-3.2# ndd -get /dev/ip \ ip6_def_hops 255 bash-3.2# ndd -get /dev/ tcp ? tcp_cwnd_max (read and write) tcp_strong_iss (read and write) tcp_time_wait_interval (read and write) tcp_tstamp_always (read and write) tcp_tstamp_if_wscale (read and write) ... bash-3.2# ndd -get /dev/tcp ecn 1 bash-3.2# ndd -get /dev/tcp sack 2</pre>	<pre>ipv6 hoplimit rw 255 -- 255 1- 255 bash-3.2# ipadm show-prop tcp PROTO PROPERTY PERM CURRENT PERSISTENT DEFAULT POSSIBLE tcp ecn rw passive -- passive never,passive, active tcp extra_ rw 2049 2049,4045 2049,4045 1- 65535 priv_ports tcp largest_ rw 65535 -- 65535 1024-65535 anon_port tcp recv_ rw 128000 -- 128000 2048-1073741824 maxbuf tcp sack rw active -- active never,passive, active tcp send_ rw 49152 -- 49152 4096-1073741824 maxbuf tcp smallest_ rw 32768 -- 32768 1024-65535 anon_port tcp smallest_ rw 1024 -- 1024 1024-32768 nonpriv_port ... bash-3.2# ipadm show-prop -p ecn,sack tcp PROTO PROPERTY PERM CURRENT PERSISTENT DEFAULT POSSIBLE tcp ecn rw passive -- passive never, passive,active tcp sack rw active -- active never, passive,active</pre>

Comparing the ndd Command and driver.conf Configuration to the dladm Command

In Oracle Solaris 10, you use the ndd command to customize network parameters (tunables) and some driver-specific properties. Although the ndd command is still functional in Oracle Solaris 11, the dladm command is the preferred command for managing these properties.

The `driver.conf` file is also used in Oracle Solaris 10 to configure some driver-specific properties. In Oracle Solaris 11, you can configure some common driver features by setting `dladm` properties, as well as certain driver-private features through driver-private properties.

The following three classes of tunables are configurable:

- **Common generic properties** – The majority of these properties have a straightforward mapping to a `dladm` command equivalent.

While `ndd` command parameters are queried and set with the `-get` and `-set` subcommands, `dladm` properties are queried and set with the `show-linkprop` and `set-linkprop` subcommands. You can also reset `dladm` properties by using the `reset-linkprop` subcommand. The following examples illustrate some of the differences between these two commands.

In the following example, the `ndd` command is used with the `-get` subcommand to retrieve the link speed of the datalink `net0`:

```
# ndd -get /dev/net/net0 link_speed
```

The following example shows the equivalent `dladm` command that you would use to retrieve that information from the `speed` property:

```
# dladm show-linkprop -p speed net0
```

LINK	PROPERTY	PERM	VALUE	EFFECTIVE	DEFAULT	POSSIBLE
net0	speed	r-	0	0	0	--

Another example is how the automatic negotiation of the link speed and duplex setting is enabled. In the following example, the `ndd` command is used to set the `adv_autoneg_cap` parameter:

```
# mdd -set /dev/net/net0 adv_autoneg_cap 1
```

Note that the `ndd` command does not configure settings that persist between reboots.

The following example shows how to enable the automatic negotiation of the link speed and duplex setting by using the `dladm` command to set the `adv_autoneg_cap` parameter:

```
# dladm set-linkprop -p adv_autoneg_cap=1
```

When you use the `dladm` command, the changes take place immediately and are persistent between system reboots.

- **Capability related tunables** – Many of these properties have an equivalent `dladm` command option in Oracle Solaris 11. The list of properties is extensive. See the "Ethernet Link Properties" section of the [dladm\(1M\)](#) man page.

You can display these properties by using the `dladm` command without any options, or you can use the `dladm show-ether` command. If you do not specify any options with the `dladm show-ether` command, only the current Ethernet property values for the datalink

are displayed. To obtain information beyond what is provided by default, use the `-x` option, as shown in the following example:

```
# dladm show-ether -x net1
LINK   PTYPE      STATE    AUTO  SPEED-DUPLEX          PAUSE
net1   current    up       yes   1G-f                  both
--     capable    --       yes   1G-fh,100M-fh,10M-fh both
--     adv        --       yes   100M-fh,10M-fh      both
--     peeradv   --       yes   100M-f,10M-f        both
```

With the `-x` option, the command also displays the built-in capabilities of the specified link, as well as the capabilities that are currently advertised between the host and the link partner.

- **Driver-specific properties** – In Oracle Solaris 11, how you configure properties that were previously stored in the `driver.conf` file depends on the specific driver. The main property that was previously configured in this file is the maximum transmission unit (MTU) property. You manage this property by using the `dladm` command. See [“Setting the MTU Property”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

For more information about the various properties that you can customize by using the `dladm` command, see [“Obtaining Status Information for Datalink Properties”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

For information about configuring other private driver properties, refer to the manufacturer's documentation for that driver.

Configuring the Network in Oracle Solaris 11

Refer to the following information when transitioning from network administration in Oracle Solaris 10 the network administration model that is used in Oracle Solaris 11.

How the Network Is Configured During an Installation

During an installation, the network is configured as follows:

- For a GUI installation, the system-generated `Automatic` profile is activated on the system and the network is automatically configured, based on the current network conditions.
- For a text installation, you must choose one of the following: `Automatic`, `Manual`, or `None`.

- If you choose `Automatic`, the `Automatic` profile is activated on the system and the network is automatically configured upon reboot. See [“Managing Network Configuration in Reactive Mode” on page 101](#).
- If you choose `Manual`, the system's only fixed profile (`DefaultFixed`) 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` is activated on the system, 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 persistent network configuration after the installation. See [“Network Administration Task Comparison” on page 95](#).
- 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 at that time. See [“Installing Oracle Solaris 11.2 Systems”](#).

Starting with Oracle Solaris 11.2, the `svc:/network/install:default` SMF service includes two new property group types: `ipv4_interface` and `ipv6_interface` that enable you to configure multiple network interfaces during an installation. You can create SC profiles that contain property groups with the type `ipv4_interface` and, or `ipv6_interface`. The existing `install_ipv4_interface` and `install_ipv6_interface` property groups for this service continue to be supported in this release. See [“Installing Oracle Solaris 11.2 Systems”](#).

Because the commands that you use to manage network configuration vary, depending on which network mode your system defaults to after an installation, make sure you know which network profile is currently active on your system by using the `netadm list` command. See [“Enabling and Disabling Profiles” in “Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

Network Administration Task Comparison

The following table compares Oracle Solaris 10 network administration tasks to Oracle Solaris 11 network administration tasks. For more information administering network configuration in Oracle Solaris 11, see the [`dladm\(1M\)`](#) and [`ipadm\(1M\)`](#) man pages.

TABLE 7-3 Comparing Oracle Solaris 10 Network Administration With Oracle Solaris 11 Network Administration

Task Information	Oracle Solaris 10	Oracle Solaris 11
Datalink configuration	<code>dladm</code> command	<code>dladm</code> command

Task Information	Oracle Solaris 10	Oracle Solaris 11
IP interface and IP address configuration	ifconfig command Edit the /etc/hostname* file	ipadm command
DHCP server configuration	SMF service: svc:/network/dhcp-server: and default DHCP management commands: dhcpcmgr, dhtadm, and pntadm	(Sun Legacy) SMF service: svc:/network/dhcp-server: SMF service and default DHCP management commands: dhcpcmgr, dhtadm, and pntadm (ISC) SMF services: svc:/network/dhcp/server:ipv4 svc:/network/dhcp/server:ipv6 svc:/network/dhcp/relay:ipv4 svc:/network/dhcp/relay:ipv6 Configuration files: /etc/inet/dhcpd4.conf /etc/inet/dhcpd6.conf
DHCP client configuration	ifconfig command Edit the /etc/dhcp* and /etc/default/dhcpagent files	ipadm command Edit the /etc/default/dhcpagent file
Name services switch configuration	SMF service: svc:/system/name-service/switch:default and /etc/nsswitch.conf	SMF service: svc:/system/name-service/switch:default. Modify with svccfg; view with svcprop -p config svc:/system/name-service/switch:default
System host name configuration	Edit the /etc/nodename file	hostname command
TCP/IP host name configuration	Edit the /etc/inet/hosts file	Edit the /etc/inet/hosts file
Network parameter administration (tunables)	ndd command	ipadm command
Wireless network configuration	wificonfig command	dladm command

Administering Datalink Configuration

When you perform a fresh installation, all datalinks are automatically assigned generic names by using the net0, net1, and netN naming convention, depending on the total number of network devices on a system. After the installation, you can use different datalink names. See [Chapter 2, “Administering Datalink Configuration in Oracle Solaris,”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

Note that 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
```

Based on the criteria, Ethernet devices on a lower motherboard or IO board, host bridge, PCIe root complex, bus, device, and function are ranked ahead of the other devices. You can display the correspondences of link names, devices, and locations as follows:

```
# dladm show-phys -L
LINK      DEVICE      LOCATION
net0      e1000g0     MB
net1      e1000g1     MB
net2      e1000g2     MB
net3      e1000g3     MB
net4      ibp0        MB/RISER0/PCIE0/PORT1
net5      ibp1        MB/RISER0/PCIE0/PORT2
net6      eoib2       MB/RISER0/PCIE0/PORT1/cloud-nm2gw-2/1A-ETH-2
net7      eoib4       MB/RISER0/PCIE0/PORT2/cloud-nm2gw-2/1A-ETH-2
```

In Oracle Solaris 10, you can use the `/etc/path_to_inst` file 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.

See [Chapter 2, “Administering Datalink Configuration in Oracle Solaris,”](#) in “[Configuring and Administering Network Components in Oracle Solaris 11.2](#)”.

Configuring IP Interfaces and Addresses

You use the `ipadm` command to configure IP interfaces and addresses in Oracle Solaris 11. 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
```

You can use the `-T` option 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, as shown in the following example:

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

See [Chapter 3, “Configuring and Administering IP Interfaces and Addresses in Oracle Solaris,”](#) in “[Configuring and Administering Network Components in Oracle Solaris 11.2](#)” and [ipadm\(1M\)](#).

Configuring Persistent Routes

Because the `/etc/defaultrouter` file is deprecated in Oracle Solaris 11, you can no longer manage routes (default or otherwise) by using this file. Using the `route` command is the only way that you can manually add a route to a system. To make the changes persistent across reboots, use the `-p` option with the `route` command.

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

For example, you would add a route to network `10.0.5.0`, which has its gateway as the border router, as follows:

```
# route -p add -net 10.0.5.0/24 -gateway 10.0.5.150
add net 10.0.5.0: gateway 10.0.5.150
```

View routes that were created by the using the previous command as follows:

```
# route -p show
```

Also, note that after an installation, you can no longer determine a system's default route by checking the `/etc/defaultrouter` file. To display the currently active routes on a system, use the `netstat` command with the following options:

```
# netstat -rn
```

See the [netstat\(1M\)](#) and [route\(1M\)](#) man pages.

For instructions, see “[Creating Persistent \(Static\) Routes](#)” in “[Configuring and Administering Network Components in Oracle Solaris 11.2](#)”.

Configuring Naming and Directory Services

In this release, the SMF repository is the primary repository for all naming services configuration. The previous behavior of modifying a particular file to manage naming services configuration no longer applies. For a list of the naming services that has migrated to SMF, see [Table 8-2](#).

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. If necessary, you can use the `nscfg` command to import or export legacy name service configuration files into or out of the SMF repository. When a valid SMF configuration and corresponding Fault Management Resource Identifier (FMRI) is supplied, the `nscfg` command regenerates legacy naming service configuration files, for example, `nsswitch.conf`, `resolv.conf`, `nscd.conf`, into their legacy locations. See “[Importing Naming Services Configuration](#)” in “[Configuring and Administering Network Components in Oracle Solaris 11.2](#)” and [nscfg\(1M\)](#).

Note - The persistent configuration of naming services through SMF applies to the fixed network configuration mode *only* and only when the `DefaultFixed` profile is active on the system. If you are using the reactive mode and the `Automatic` or another reactive profile is active on the system, you configure naming services in a `Location` profile by using the `netcfg` command and not through SMF. See “[Creating Locations](#)” in “[Configuring and Administering Network Components in Oracle Solaris 11.2](#)”.

The following example shows how you would configure the Domain Name Service (DNS) by using the `svccfg`. After you set the various properties, you must enable and refresh the SMF service.

```
# svccfg -s dns/client setprop config/nameserver=net_address: 192.168.1.1
# svccfg -s dns/client setprop config/domain = astring: "myhost.org"
```

```
# svccfg -s name-service/switch setprop config/host = astring: "files dns"
# svcadm refresh name-service/switch
# svcadm refresh dns/client
```

You can also configure naming and directory SMF service properties interactively. For an example, see [“Configuring a DNS Client”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

See [Chapter 4, “Administering Naming and Directory Services on an Oracle Solaris Client,”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

Administering DHCP

Note the following information about DHCP administration in this release:

- The ISC DHCP server software is available for installation in this release. You can add the server package to your system as follows:

```
# pkg install pkg:/service/network/dhcp/isc-dhcp
```

For more information about administering ISC DHCP, including configuring the ISC DHCP server administering the ISC DHCP service, see [Chapter 2, “Administering the ISC DHCP Service,”](#) in [“Working With DHCP in Oracle Solaris 11.2”](#).

- The legacy Sun DHCP server software is still part of the Oracle Solaris release, but the feature has been marked as obsolete. See [“Legacy Sun DHCP Server”](#) in [“Working With DHCP in Oracle Solaris 11.2”](#).
- The term DHCP client refers to a software entity. The DHCP client is a daemon (dhcpcagent) that runs on systems that are configured to request their network configuration from the DHCP service. Both the legacy Sun DHCP server and the ISC DHCP server work with the DHCP client. See [Chapter 3, “Configuring and Administering the DHCP Client,”](#) in [“Working With DHCP in Oracle Solaris 11.2”](#) for details.

Setting a System's Host Name

The primary interface's TCP/IP host name is a distinct entity from the system host name that you set with the `hostname` command. Although not required by Oracle Solaris, the same name is normally used for both. Some network applications depend on this convention. See [hostname\(1\)](#).

Permanently set a system's host name as follows:

```
# hostname name-of-host
```

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 you

use the `hostname` command, then the `hostname` is the value that is specified in the `config/nodename` file. 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 the [hostname\(1\)](#) man page.

Managing Network Configuration in Reactive Mode

When you are using the reactive network configuration mode, the system handles network connectivity and network configuration based on the current network conditions. The type of network configuration uses different profiles to specify the various parameters that define a system's network configuration. These profiles are automatically enabled on the system in response to changes in network conditions. Or, you can manually enable profiles on a system, as needed.

Reactive network configuration is most suitable for notebook PCs and in situations where cables are regularly plugged or unplugged, cards are added or removed, etc. 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. For a detailed overview of profile-based network configuration, see [“About Profile-Based Network Configuration”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

For the reactive network configuration mode, you use the `netcfg` command to configure system-specific network configuration (datalinks and configure IP interfaces and addresses), as well as system-wide network configuration, for example, naming services. There is a second command, `netadm`, that you use to administering profiles on a system. These commands create network configuration that is applied to both the active and non-active profiles on the system.

For information about profile-based network configuration, see [Chapter 6, “Administering Profile-Based Network Configuration in Oracle Solaris,”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

You can also manage network configuration from the desktop by using the network administration GUI (formerly NWAM). This tool is similar to using the `netcfg` and `netadm` commands to manage reactive network configuration. Reactive network configuration is more suitable for notebook PC use and in situations where the network environment changes often, for example when switching from a wired connection to a wireless connection or when switching from a work location to a home office location. In these situations, you will want to activate the system-defined Automatic network profile or a user-defined reactive network profile. See [“Administering Network Configuration From the Desktop”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

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:

- “System Configuration Changes” on page 103
- “Service Management Facility Changes” on page 106
- “System Console and Terminal Services Changes” on page 110
- “Power Management Configuration Changes” on page 111
- “System Configuration Tools Changes” on page 111
- “System Registration and Customer Support Changes” on page 112
- “Boot, Recovery, Platform, Hardware, and Disk Labeling Changes” on page 113
- “System Recovery and Cloning With the Oracle Solaris Unified Archives Feature” on page 122
- “Printer Configuration and Management Changes” on page 122
- “Internationalization and Localization Changes” on page 124

System Configuration Changes

The following is a summary of Oracle Solaris 11 system configuration changes:

- `/etc/default/init` **file is read-only** – Locale and time zone configuration have migrated to the Service Management Facility (SMF). All changes to environment variables should be managed through the new `svc:/system/environment:init` SMF service. See “Internationalization and Localization Changes” on page 124.
- `/etc/dfs/dfstab` **configuration** – Publishing and unpublishing a file system share is 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 “Configuring the Network in Oracle Solaris 11” on page 94.

- **/etc/system.d implementation** – This directory provides an easier way to package Oracle Solaris kernel configuration than the traditional method of editing the `/etc/system` file. Because you can use IPS to deliver fragments (one line or many) into files within the `/etc/system.d/` directory, rather than editing the `/etc/system` file through first boot SMF services or other scripting, you can deliver any Oracle Solaris kernel customizations much more easily. See [system\(4\)](#).

Note - The `/etc/system` file remains fully supported in this release. However, for third-party software, using files within the `/etc/system.d/` directory rather than editing the `/etc/system` file is encouraged.

Also, as part of this change, the `cryptoadm` and `dttrace` commands have been updated to write to files within the `/etc/system.d/` directory rather than the `/etc/system` file as in previous releases. See [cryptoadm\(1M\)](#) and [dttrace\(1M\)](#).

- **Mapping a host name to a system's primary interface** – A system's 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 111](#).
- **Setting a system's host name** – Use the `hostname` command to permanently set a 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 and Terminal Services Changes” on page 110](#).
- **System logging services** – The `rsyslog` daemon 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.

Display the status of the `system-log` services as follows:

```
# 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 in Oracle Solaris 11.

- **System recovery and cloning** – The Oracle Unified Archives feature provides support for boot environments (BEs), IPS, and the various virtualization technologies that are available in Oracle Solaris 11. The Unified Archives feature is more robust and flexible than the flash archive installation method that is used in Oracle Solaris 10. See [“System Recovery and Cloning With the Oracle Solaris Unified Archives Feature”](#) on page 122.
- **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, Timezone, and Console Keymap Configuration Changes”](#) on page 127.

Oracle Solaris 10 and Oracle Solaris 11 System Configuration Feature Comparison

The following table compares Oracle Solaris 10 and Oracle Solaris 11 system configuration features.

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
System configuration (network, and naming service configuration)	Configured in various files within the <code>/etc</code> directory	Configured through properties of the appropriate SMF service. See “Configuring Naming and Directory Services” on page 99
System console service (serial port monitor) configuration	<code>getty, pmadm, ttyadm, ttymon</code>	Configured through properties of the appropriate SMF service See “System Console and Terminal Services Changes” on page 110
System configuration (host name)	Edit <code>/etc/nodename</code>	Use the <code>hostname</code> command. See hostname(1)
System configuration (Oracle Solaris kernel customizations)	Edit <code>/etc/system</code>	Edit <code>/etc/system</code> Add configuration to files within the <code>/etc/system.d</code> directory

System Configuration Feature, Tool, or Function	Oracle Solaris 10	Oracle Solaris 11
System logging	syslog	syslog (default) and rsyslog See “Naming and Directory Services Migration to SMF” on page 106
Power management	Edit /etc/power.conf or use the pmconfig command	poweradm See “Power Management Configuration Changes” on page 111
System unconfiguration and reconfiguration	By using the sysidtool.sys-unconfig, sysidconfig, and sysidcfg commands	sysconfig or the SCI tool See “System Configuration Tools Changes” on page 111
System registration	Auto Registration feature Starting with Oracle Solaris 10 1/13: Oracle Configuration Manager	Oracle Configuration Manager See “System Registration and Customer Support Changes” on page 112
System recovery and cloning	Oracle Solaris Flash Archive features	Oracle Solaris Unified Archives “System Recovery and Cloning With the Oracle Solaris Unified Archives Feature” on page 122
Printer configuration and administration	LP print commands, Solaris Print Manager	CUPS command-line, CUPS Print Manager, and CUPS web browser interface See “Printer Configuration and Management Changes” on page 122
Locale and time zone configuration	Edit /etc/default/init	Configured through properties of the appropriate SMF service See “Locale, Timezone, and Console Keymap Configuration Changes” on page 127

Service Management Facility Changes

The following information pertains to Service Management Facility (SMF) changes in Oracle Solaris 11.

Naming and Directory Services Migration to SMF

Naming and directory services configuration are managed through SMF in this release. The following table describes the various configuration files that have migrated to SMF control. For

information about importing legacy naming services configuration to SMF after an installation, see [“Importing Naming Services Configuration”](#) in [“Configuring and Administering Network Components in Oracle Solaris 11.2”](#).

TABLE 8-2 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
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.yupdated)
svc:/system/name-service/upgrade:default	Not applicable	Naming legacy file to SMF upgrade service

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 which are delivered by a service manifest or profile.

The different settings by administrator, profile, or manifest are captured in *layers*. Use the `svccfg listprop` command with the `-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, as shown in the following 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  adminv    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 you use the `-l` option.

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, you can use the `svccfg listcust` command to list customizations *only*.

Services and instances that are delivered in standard locations (`/lib/svc/manifest` and `/etc/svc/profile`) are 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, you can use the `svccfg delete` command. 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 you made with the `svccfg delete` command, and return to the configuration that is provided by the service manifest, use the `svccfg delcust` 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
```

See [“Managing System Services in Oracle Solaris 11.2”](#) and [svccfg\(1M\)](#).

SMF Manifest Creation Tool

You can use the `svcbundle` command to generate SMF manifests. You can also use the command to generate profiles by specifying the `bundle-type` option. The generated bundle is defined by multiple `-s name=value` options. Some examples of name arguments include `bundle-type`, `instance-name`, `service-name`, and `start-method`. To generate a manifest, you must specify a `service-name` and `start-method`. The `svcbundle` command uses default values for some service characteristics. You can edit the generated manifest. Follow the DTD that is specified at the top of the manifest. For detailed instructions, see [Chapter 5, “Using SMF to Control Your Application,”](#) in [“Managing System Services in Oracle Solaris 11.2”](#) and [svcbundle\(1M\)](#).

System Process Summary Information

Both Oracle Solaris 10 and Oracle Solaris 11 include some system processes that perform a specific task, but typically do not require any administration, such as those that are listed in the following table.

TABLE 8-3 System Processes That Require No Administration

System 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- <i>pool-name</i>	System process for each ZFS storage pool containing the I/O taskq threads for the associated pool

System Console and Terminal Services Changes

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 `svc:/system/console-login` SMF service. 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. See [Chapter 5, “Managing the System Console, Terminal Devices, and Power Services,”](#) in [“Managing System Information, Processes, and Performance in Oracle Solaris 11.2”](#).

Power Management Configuration Changes

In Oracle Solaris 10, you administer power management 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 in Oracle Solaris 11 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. See [poweradm\(1M\)](#).

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 11.2”](#).

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.

The `sysconfig configure` command produces similar results to the `sys-unconfig` command that is used 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 you would 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, as shown in this example:

```
# 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 [sysconfig\(1M\)](#) and [Chapter 6, “Unconfiguring or Reconfiguring an Oracle Solaris Instance,”](#) in “Installing Oracle Solaris 11.2 Systems”.

System Registration and Customer 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 and problem avoidance, as well as 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 (ASR) 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 supported 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 11.2 Systems](#)”.

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>.

Boot, Recovery, Platform, Hardware, and Disk Labeling Changes

An Oracle Solaris 11 system boots from a ZFS root file system by default and the ZFS root file system is contained within a ZFS root pool named `rpool`. Creating a UFS file system is still supported in Oracle Solaris 11, but you can no longer boot from a UFS or a Solaris Volume Manager root file system.

Review the following information, as it 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 OBP ok 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 this release, you can create and deploy Oracle Solaris Unified Archives to perform system recovery and cloning operations. Oracle Solaris Unified Archives are system archives that can contain one or more archived instances of the OS. Each instance is an independently referenced system. An instance is defined as a boot environment in either a global or a non-global zone. Each system archive can contain any number of global and non-global zones. See “[Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.2](#)” for details.

x86: GRand Unified Bootloader Changes

GRUB 2 is the default boot loader, starting with Oracle Solaris 11.1. GRUB 2 replaces the original GRUB 0.97-based boot loader (GRUB Legacy) that is used in Oracle Solaris 10 and Oracle Solaris 11 11/11. 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 that is used in Oracle Solaris 11.

If you are transitioning from Oracle Solaris 10 to Oracle Solaris 11, note the following key differences between the two GRUB versions:

- **GRUB menu changes** – Unlike the editable `menu.lst` file that is used by GRUB Legacy, GRUB 2 stores its configuration in the `grub.cfg` file. This file is syntactically different than the legacy `menu.lst` file and is also not meant to be edited. 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 includes several new subcommands, as well as a new `-P` option for administering the GRUB configuration for multiple root pools.

Note - Because any GRUB configuration changes can automatically overwrite changes that you might make to the `grub.cfg` file, *do not* manually edit this file. Instead, use the `bootadm` command to update the GRUB configuration file. See [Chapter 2, “Administering the GRand Unified Bootloader \(Tasks\)”](#), in “[Booting and Shutting Down Oracle Solaris 11.2 Systems](#)” and [bootadm\(1M\)](#).

- **Managing non-Solaris boot entries** – GRUB 2 includes an additional configuration file named `custom.cfg`. You use this file 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 store it in the same location as the `grub.cfg` file (`/pool-name/boot/grub/`). During the boot process, GRUB checks for the existence of the `custom.cfg` file in the top-level dataset of the root pool (`boot/grub`). If the file exists, GRUB sources the file and then 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 11.2 Systems](#)”.

If you are running an Oracle Solaris 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 11.2 Systems](#)”.

Firmware, Disk Labeling, and EEPROM Changes

If you are transitioning from Oracle Solaris 10, note the following feature changes:

- **64-Bit UEFI firmware support** – Oracle Solaris 11 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 11.2 Systems](#)” for more details.

- **Support for GPT labeled disks**— GPT labeled disks are supported on both SPARC and x86 platforms. Installing 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. For SPARC based systems that support a GPT labeled boot disk, see [“SPARC: GPT Labeled Disk Support”](#) in [“Oracle Solaris 11.2 Release Notes ”](#) [“Firmware Issues”](#) in [“Oracle Solaris 11.2 Release Notes ”](#) for information about how to apply the GPT-aware firmware update. Otherwise, installing Oracle Solaris 11.2 on a SPARC based system applies an SMI (VTOC) label to the root pool disk with a single slice 0.
- **Setting EEPROM variables on UEFI enabled systems** – For UEFI enabled systems, the parameters are stored in two places: Oracle Solaris specific variables are stored in the `bootenv.rc` file and UEFI specific variables are set in the NVRAM store. Unlike SPARC based systems with the OpenBoot PROM (OBP), Oracle Solaris variables are not consumed by UEFI firmware. To enable the availability of UEFI specific variables available, you can use the `eeprom` command with the `-u` option. Most UEFI variables are in a binary format and are translated to a readable format. When translation is not possible, a hexdump is printed. See [`eeprom\(1M\)`](#) for details of this change.
- **Installing boot blocks** — You use the `bootadm install-bootloader` command to install or reinstall the boot loader on both SPARC and x86 based systems. This command replaces the `installboot` command on SPARC platforms and the `installgrub` command x86 platforms. See [`bootadm\(1M\)`](#).

Additional Boot, Platform, and Hardware Changes

Note the following boot, platform, and hardware feature changes:

- **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 - 32-bit applications are not impacted by this change.

- **Bitmapped console support** – Support for high resolution and color depth consoles has been added to Oracle Solaris. 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. You can control the console type (and also the older VGA TEXT 640x480 console) 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 11.2 Systems ”](#).

- Fast Reboot support on x86 and SPARC platforms** – On x86 platforms, a 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 you can enable or disable as needed. See [“Accelerating the Reboot Process”](#) in [“Booting and Shutting Down Oracle Solaris 11.2 Systems”](#).

- 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!
```

Booting a System for Recovery

If an Oracle Solaris 11 system becomes unbootable, you most likely will need to perform a recovery boot. You can boot from installation media, or you can boot a backup BE.

If you need to perform complete system (bare metal) recovery, see [“How to Create a Recovery Archive”](#) in [“Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.2”](#).

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 117 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.

Note - 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 11.2 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 11.2 SPARC
2 solaris-backup
Select environment to boot: [ 1 - 2 ]: 2
```

In the previous output, the active BE is Oracle Solaris 11.2 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 a System For Recovery Purposes” on page 118](#).

- **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.**
 - 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 a System For Recovery Purposes” on page 118](#).

▼ How to Boot a System For Recovery Purposes

1. Select the appropriate boot method:

- **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** – Booting from an install server on the network requires 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 an x86 based system, edit the selected boot entry in the GRUB menu, then add the `-s` kernel argument to the end of the `$kernel` line.

On a SPARC based system, shut down the system and boot to single-user mode. After you log in as the root user, edit the `/etc/passwd` file, and fix the root shell entry.

```
# init 0
ok boot -s
Boot device: /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@0,0:a ...
SunOS Release 5.11 Version 11.2 64-bit
Copyright (c) 1983, 2013, Oracle and/or its affiliates. All rights reserved.
Booting to milestone "milestone/single-user:default".
Hostname: systema.domain
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): xxxxxxxx
single-user privilege assigned to root on /dev/console.
Entering System Maintenance Mode
```

```
Aug 3 15:46:21 su: 'su root' succeeded for root on /dev/console
Oracle Corporation SunOS 5.11 11.2 July 2013
su: No shell /usr/bin/mybash. Trying fallback shell /sbin/sh.
root@systema.domain:~# TERM =vt100; export TERM
root@systema.domain:~# vi /etc/passwd
```

```
root@systema.domian:~# <Press control-d>
logout
svc.startd: Returning to milestone all.
```

- Resolve a corrupt boot loader problem.

First, boot from media or the network by using one of the boot methods that are listed in Step 1. Then, import the root pool.

```
# zpool import -f rpool
```

Note - Do not use 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 11.2 Systems”](#).

Then, reinstall the boot loader as follows:

```
# bootadm install-bootloader -f -P rpool
```

where `-f` forces the installation of the boot loader and bypasses any checks related to not downgrading the version of the boot loader on the system. The `-P` option specifies the root pool.

Exit and reboot the system.

```
# exit
1 Install Oracle Solaris
2 Install Additional Drivers
3 Shell
4 Terminal type (currently sun-color)
5 Reboot
```

```
Please enter a number [1]: 5
```

Confirm that the system boots successfully.

- Resolve an unknown root password that prevents you from logging into the system.

First, you must boot from media or the network by using one of the boot methods that are listed in Step 1. 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.

```
# zpool import -f rpool
# beadm list
be_find_current_be: failed to find current BE name
be_find_current_be: failed to find current BE name
BE           Active Mountpoint Space Policy Created
--           -
solaris     -           -           11.45M static 2011-10-22 00:30
```



```

solaris-2      R      -      12.69G static 2011-10-21 21:04
# mkdir /a
# beadm mount solaris-2 /a
# TERM=vt100
# export TERM
# cd /a/etc
# vi shadow
<Carefully remove the unknown password>
# cd /
# beadm umount solaris-2
# halt

```

3. Set the root password by booting to single-user mode and setting the password.

This step assumes that you have removed an unknown root password in the previous step.

- On an x86 based system, edit the selected boot entry in the GRUB menu, then add the `-s` option to the `$kernel` line.
- On a SPARC based system, 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 11.2 64-bit
Copyright (c) 1983, 2012, Oracle and/or its affiliates. All rights
reserved.
Booting to milestone "milestone/single-user:default".
Hostname: systema.domain
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
.
.
.
root@sysadma.domain:~# passwd -r files root
New Password: xxxxxx
Re-enter new Password: xxxxxx
passwd: password successfully changed for root
root@systema.central:~# <Press control-d>
logout
svc.startd: Returning to milestone all.

```

System Recovery and Cloning With the Oracle Solaris Unified Archives Feature

The Oracle Solaris Unified Archives feature supports multiple system archives that consist of one or more point-in-time system archive images in a single file format. Unified Archives can contain one or more archived instances of Solaris from a single host. You can select individual installed zones to include during archive creation, while the host itself is optional. Unified archives provide similar functionality to the Oracle Solaris Flash Archive installation method that is supported in Oracle Solaris 10.

You can deploy a unified archive to perform system recovery, cloning, or migration by using any of the following methods

- AI installation method
- Oracle Solaris Zones utilities
- Unified Archive bootable media

In Oracle Solaris 10, the Oracle Solaris Flash Archive installation method is used. Introduced before the wide adoption of virtual systems, flash archives are designed to create and deploy OS instances for bare metal systems. Flash archives capture file system data from a running system and any system-related metadata. However, to support boot environments (BEs), the Image Packaging System (IPS), and the various virtualized technologies that are used in Oracle Solaris 11, a more flexible and robust archive solution was required. Unified Archives provide support for virtualized environments, such as zones, as well as cross platform portability within the same hardware architecture.

The `archiveadm` command enables you to create system archive images of a running Oracle Solaris system for the purposes of system cloning and recovery. You can also use the command to obtain information about existing archives and create bootable media from an archive. See [archiveadm\(1M\)](#).

For additional information, see “[Using Unified Archives for System Recovery and Cloning in Oracle Solaris 11.2](#)”.

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 [“Configuring and Managing Printing in Oracle Solaris 11.2”](#).
- Several LP print commands, files, and services are no longer available. Some LP print commands, for example `lp`, `lpadmin`, `lpc`, `lpr` are still available. In Oracle Solaris 11, these commands are 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, Files, and Services”](#) on page 17.
- 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 11.2”](#).
- 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. Starting with 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 an Installation”](#) on page 124.
- Printers that are configured on a per-user basis in the `~/.printers` file no longer work. Printer configuration is 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 11.2”](#).

- The `lp` entry in the `/etc/passwd` file is 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 11.2”](#).

▼ How to Set Up Your Printing Environment After an Installation

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 [“Configuring and Managing Printing in Oracle Solaris 11.2”](#).

Internationalization and Localization Changes

Note 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)

Other notable core locale changes include the addition of the Portuguese – Brazilian locale and the removal of the Swedish locale.

- **Other locale changes** – Starting with Oracle Solaris 11.1, the following locale changes are implemented:
 - 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.
 - Locale data for Simplified Chinese, Traditional Chinese, Korean, and Thai UTF-8 locales has been updated to support Unicode 6.0.
- **Language and locale packaging** – In Oracle Solaris 10, optional package components, such as documentation, localization, and debug files are split into separate packages. However, 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
```

Starting with Oracle Solaris 11.2, you can use the `nlsadm` command to administer locales in place of the `localeadm` command that is used in Oracle Solaris 10. The `nlsadm` command provides a consolidated and convenient way to administer national language properties.

For example, you would use the following command to install the Danish locale and any available translations:

```
# nlsadm install-locale da_DK.UTF-8
```

Note - You might need to install the `nls-administration` software package prior to using the `nlsadm` command on your Oracle Solaris 11.2 system:

```
# pkg install nls-administration
```

Although the `nlsadm` command is the suggested method for installing and uninstalling locales in Oracle Solaris 11.2, you can still install and remove locales by modifying the locale facets directly, as shown in the following example:

```
# 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. If you are using the `nlsadm` command, the command automatically installs all of the necessary packages. If you are not using the `nlsadm` command, you must install the `system/locale/extra` package to enable these locales. See “[Controlling Installation of Optional Components](#)” in “[Adding and Updating Software in Oracle Solaris 11.2](#)”.

- **Setting a system's default locale** – In Oracle Solaris 10, the default system locale is configured in the `/etc/default/init` file. Starting with 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, Timezone, and Console Keymap Configuration Changes](#)” on page 127.
- **Short form locales** – Oracle 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. Starting with Oracle Solaris 11, you should use fully qualified locale names. Or, if possible, use UTF-8 locales. See the end-of-feature announcements at <http://www.oracle.com/technetwork/systems/end-of-notice/eonsolaris11-392732.html>.
- **Locale aliasing** – Locale aliases are new. Locale name aliases are accepted and mapped to the corresponding canonical locale names. For example, the `de` locale is mapped to the canonical `de_DE.ISO8859-1` locale. For all of the locale name mappings, see the [locale_alias\(5\)](#) man page.

Locale, Timezone, and Console Keymap Configuration Changes

In Oracle Solaris 10, locale, timezone, and console keymap configuration is set in the `/etc/default/init` file. In Oracle Solaris 11, you manage this configuration through the following SMF services:

- System locale – `svc:/system/environment:init`
- Timezone – `svc:/system/timezone:default`
- Console keymap – `svc:/system/keymap:default`

Starting with Oracle Solaris 11.2, you can use the `nlsadm` command to display and set these national language properties. The following examples show how to set these properties by using the `nlsadm` command.

Note - You might need to run the following command prior to using the `nlsadm` command:

```
# pkg install nls-administration
```

Change the default locale to `fr_FR.UTF-8` as follows:

```
# nlsadm set-system-locale fr_FR.UTF-8
```

Set the timezone to `Europe/Paris` as follows:

```
# nlsadm set-timezone Europe/Paris
```

Set the console keymap to `US-English` as follows:

```
# nlsadm set-console-keymap US-English
```

For other date and time configuration changes, see [“Reconfiguring Date and Time Before and After an Installation” on page 38](#).

Managing Security

This chapter describes security feature changes in Oracle Solaris 11 releases.

The following topics are covered:

- “Security Feature Changes” on page 129
- “Roles, Rights, Privileges, and Authorizations” on page 133
- “File and File System Security Changes” on page 137

Security Feature Changes

Note 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
```

In this release, auditing is on by default. For a secure system, use the interfaces that are always audited when auditing of administrative actions is turned on. Because `pfedit` use is always audited, it is the preferred command for editing system files. See [pfedit\(1M\)](#) and [Chapter 3, “Controlling Access to Systems,” in “Securing Systems and Attached Devices in Oracle Solaris 11.2”](#).

- **Auditing** – Auditing is a service in Oracle Solaris 11 and is enabled by default. No reboot is required when disabling or enabling this service. You use the `auditconfig` command 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 [“Managing Auditing in Oracle Solaris 11.2”](#).

- **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` plugin. 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 `-set remote` option in [`auditconfig\(1M\)`](#).
- **Compliance assessment** – Use the `compliance` command (new in Oracle Solaris 11.2) to automate compliance assessment, not remediation. You can use the command to list, generate, and delete assessments and reports. See [“Oracle Solaris 11.2 Security Compliance Guide”](#) and [`compliance\(1M\)`](#).
- **Basic Audit Reporting Tool (BART)** – The default hash that is used by BART is SHA256, not MD5. In addition to SHA256 being the default, you can also select the hash algorithm. See [Chapter 2, “Verifying File Integrity by Using BART,”](#) in [“Securing Files and Verifying File Integrity in Oracle Solaris 11.2”](#).
- **cryptoadm command changes** – As part of the implementation of the `/etc/system.d` directory for easier packaging of Oracle Solaris kernel configuration, the `cryptoadm` command has also been updated to write to files within the this directory rather than the `/etc/system` file as in previous releases. See [`cryptoadm\(1M\)`](#).
- **Cryptographic Framework** – This feature includes more algorithms, mechanisms, plugins, and support for Intel and SPARC T4 hardware acceleration. Also, Oracle Solaris 11 provides better alignment with the NSA Suite B cryptography. Many of the algorithms in the framework are optimized for x86 platforms with the SSE2 instruction set. For more information about T-Series optimizations, see [“Cryptographic Framework and SPARC T-Series Servers”](#) in [“Managing Encryption and Certificates in Oracle Solaris 11.2”](#).
- **dttrace command changes** – As part of the implementation of the `/etc/system.d` directory for easier packaging of Oracle Solaris kernel configuration, the `dttrace` command has also been updated to write to files within the this directory rather than the `/etc/system` file as in previous releases. See [`dttrace\(1M\)`](#).
- **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 RFC 4120.
- **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** – The `lofi` command supports the encryption of block devices in this release. See [`lofi\(7D\)`](#).
- **profiles command changes** – In Oracle Solaris 10, this command is only used to list profiles for a specific user or role, or a user’s privileges for specific commands. Starting

with Oracle Solaris 11, you can 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 139.
- **rstchown property** – The `rstchown` tunable parameter that is used in previous releases to restrict `chown` operations is a ZFS file system property, `rstchown`, and is also a general file system mount option. See “[Managing ZFS File Systems in Oracle Solaris 11.2](#)” 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)** – IKE Version 2 (IKEv2) provides automatic key management for IPsec using the latest version of the IKE protocol. IKEv2 and IPsec use cryptographic algorithms from the Cryptographic Framework feature of Oracle Solaris. IKEv2 includes more Diffie-Hellman groups and can also use Elliptic Curve Cryptography (ECC) groups. See [Chapter 8, “About Internet Key Exchange,”](#) in “[Securing the Network in Oracle Solaris 11.2](#)”.
- **IP Security Architecture (IPsec)** – IPsec includes AES-CCM and AES-GCM modes and is capable of protecting network traffic for the Trusted Extensions feature of Oracle Solaris (Trusted Extensions). See [Chapter 6, “About IP Security Architecture,”](#) in “[Securing the Network in Oracle Solaris 11.2](#)”.
- **IP Filter Firewall** – IP Filter Firewall, which is similar to the open source IP Filter feature, is compatible, manageable, and highly integrated with SMF. This feature enables selective access to ports, based on IP address.
- **Kerberos** – Kerberos is capable of mutual authentication of clients and servers in this release. Also, support for initial authentication by using X.509 certificates with the PKINIT protocol has been introduced. See “[OpenSSL Support in Oracle Solaris](#)” in “[Managing Encryption and Certificates in Oracle Solaris 11.2](#)”.
- **OpenSSL 1.0.1** – Starting with Oracle Solaris 11.2, OpenSSL 1.0.1 is supported. This version of OpenSSL provides you with a choice between performance or FIPS-140 compliance. See https://blogs.oracle.com/observatory/entry/openssl_on_solaris_11_2.

- **Secure by Default** – You use the Secure by Default feature to disable and protect several network services from attack, which provides minimization of network exposure. In Oracle Solaris 10, this feature was introduced, but was turned off by default and had to be enabled during the OS installation or by running the `net services limited` command. Starting with Oracle Solaris 11, this feature is enabled by default, and only SSH is enabled for remote access to the system. To enable remote access for other services, see instructions in the man page for each specific network service.
- **SSH** – Support for host and user authentication by using X.509 certificates is supported.

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 `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, as shown in this 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 the correct 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 that you made. See [pam.conf\(4\)](#).

- **definitive flag added to pam.conf** – The `pam.conf` file includes the definitive `control_flag` in this release. See [pam.conf\(4\)](#).

Removed Security Features

The following security features are excluded:

- **Automated Security Enhancement Tool (ASET)** – The ASET functionality is replaced by a combination of IP Filter, which includes `svc.ipfd`, BART, SMF, and other security features that are supported in this release.
- **Smartcard** – Smartcard support is no longer available.

Roles, Rights, Privileges, and Authorizations

The following information describes how roles, rights, privileges, and authorizations work 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. Starting with 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 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. 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. You can determine which password has been assigned to the `root` role as follows:

```
# userattr roleauth root
```

No output means the root account has not been customized, meaning the password is the Oracle Solaris default and not the user's password.

See [user_attr\(4\)](#).

- **root as a Role** – root is a role by default in Oracle Solaris 11, 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 “[Securing Users and Processes in Oracle Solaris 11.2](#)”.

- **Oracle Solaris basic privileges include the following:**
 - file_read
 - file_write
 - net_access

- **Profile shell versions of regular shells** – Every regular shell has its own profile version in Oracle Solaris 11. The following profile shells are available:
 - pfbash
 - pfcsh
 - pfksh
 - pfksh93
 - pfrksh93
 - pfish
 - pftcsh
 - pfzsh

See [pfexec\(1\)](#).

- **Rights profiles** – The `user_attr`, `prof_attr`, and `exec_attr` databases are 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 rights 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 135.

- **Stop rights profile** – This profile enables administrators to create restricted accounts. See “[More About Rights Profiles](#)” in “[Securing Users and Processes in Oracle Solaris 11.2](#)”.

- **pfsh script command** – This command runs the same as the `pfsh -c script` command. Previously, commands within a script would not be able to take advantage of using rights profiles, unless the script specified a profile shell as its first line. This rule required you to modify any scripts to use rights profiles, 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 no longer setuid root. The new PF_PFEEXEC 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 searches for an entry in the Forced Privilege rights 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 execution rights 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. See [Chapter 8, “Reference for Oracle Solaris Rights,”](#) in [“Securing Users and Processes in Oracle Solaris 11.2”](#).

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
```

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
P: file_link_any,...,proc_clock_highres,proc_session
L: cpc_cpu,dtrace_kernel,dtrace_proc,dtrace_user,...,sys_time
$ ppriv -vI 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 changes pertain to file and file system security.

acemode Property Reintroduced

The `acemode` property that determines how the ACL permissions on a file are modified during a `chmod` operation is reintroduced in this release. The `acemode` 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 `acemode` 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 `acemode` 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 acemode=mask pond/whoville
# zfs set aclinherit=restricted pond/whoville
```

```
# ls -lV file.1
-rwxrwx---+ 1 root      root      206695 Aug 30 16:03 file.1
user:amy:r-----a-R-c---:-----:allow
user:roxy: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 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:roxy: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:roxy: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:roxy: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. However, 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
```

See [“Encrypting ZFS File Systems”](#) in [“Managing ZFS File Systems in Oracle Solaris 11.2”](#).

Immutable Zones

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 [“Creating and Using Oracle Solaris Zones”](#).

Managing Oracle Solaris Releases in a Virtual Environment

This chapter describes the virtualization environments that are supported in Oracle Solaris 11 releases.

The following topics are covered:

- “Oracle Solaris Virtualization Features” on page 141
- “Consolidating Legacy Oracle Solaris Systems With Oracle VM Server” on page 142
- “Oracle Solaris Zone Features” on page 143
- “Transitioning an Oracle Solaris 10 Instance to a Non-Global Zone on an Oracle Solaris 11 System” on page 146

Oracle Solaris Virtualization Features

The following table provides a brief description of the various virtualization features that are supported in Oracle Solaris 11. All of these features are also supported in Oracle Solaris 10.

TABLE 10-1 Virtualization Features Supported in Oracle Solaris 11

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	Supported	“Administering Resource Management in Oracle Solaris 11.2”
Oracle VM Server for SPARC	Hypervisor-based virtualization for SPARC servers	Supported	http://www.oracle.com/technetwork/documentation/vm-sparc-194287.html
Oracle VM Server for x86 (Xen)	Hypervisor-based virtualization for x86 based servers	Supported	http://www.oracle.com/technetwork/documentation/vm-096300.html

Oracle Solaris 11 Feature	Description	Oracle Solaris 10 Support	For More Information
Oracle VM VirtualBox	Hosted workstation and server virtualization for x86 based systems	Supported	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	Supported	“Introduction to Oracle Solaris Zones ”
Oracle VM Templates	The following types of Oracle Solaris VM templates a available: Oracle VM Templates for Oracle Solaris Zones, Oracle VM Templates for SPARC, Oracle VM Templates for x86, and Oracle VM Templates for Oracle VM VirtualBox.	Supported in some Oracle Solaris 10 releases	http://www.oracle.com/technetwork/server-storage/solaris11/downloads/virtual-machines-1355605.html

For an introduction to Oracle Solaris virtualization environments, see “[Introduction to Oracle Solaris 11.2 Virtualization Environments](#) ”.

Consolidating Legacy Oracle Solaris Systems With Oracle VM Server

Oracle VM Server for SPARC 3.1 provides several virtual networking performance improvements, including the addition of following:

- Dynamic Single-Root I/O Virtualization (SR-IOV)
- Extension of Direct I/O and SR-IOV features to non-Primary root domains
- InfiniBand device support for SR-IOV features, as well as Ethernet devices
- Fault Management Architecture (FMA) blacklisting feature
- Recovery mode feature that automatically recovers domain configurations that cannot be booted due to faulty or missing resources
- `ldmpower` command to show per-domain, power-consumption information
- Support for VNICs on virtual networks

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 Oracle Solaris 10 in a logical domain on a chip multithreading (CMT) system.

Run the `ldmp2v` command from a control domain that is running Oracle Solaris 10 or Oracle Solaris 11 to convert one of the following source systems to a logical domain:

- Any `sun4u` SPARC based system that is running at least Solaris 8, Solaris 9, or Oracle Solaris 10
- Any `sun4v` system that is running Oracle Solaris 10, but does not run in a logical domain

Note - The `ldmp2v` command does not support any SPARC based system that runs Oracle Solaris 10 with a ZFS root or Oracle Solaris 11.

See [Chapter 14, “Oracle VM Server for SPARC Physical-to-Virtual Conversion Tool,”](#) in [“Oracle VM Server for SPARC 3.1 Administration Guide”](#).

Oracle Solaris 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 a Non-Global Zone on an Oracle Solaris 11 System”](#) on page 146.
- 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 `zonpath` is moved to the target host, where it is attached. See [“About Detaching and Attaching the solaris10 Zone”](#) in [“Creating and using Oracle Solaris 10 Zones”](#).
- You can create and manage multiple boot environments (BEs) for a Solaris 10 branded zone, as well as modify the currently active BE or any inactive BE, all while the production workload continues to run. See [“About Multiple Boot Environments On solaris10 Zones”](#) in [“Creating and using Oracle Solaris 10 Zones”](#).
- **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 11.2 Systems”](#).
- **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 link is automatically selected at boot time. See [“Introduction to Oracle Solaris Zones”](#).

- **Legacy branded zones** – The following legacy branded zone features are supported in Oracle Solaris 10 only:
 - Linux brand (`lx`)
 - Oracle Solaris 8 Containers (`solaris8`)
 - Oracle Solaris 9 Containers (`solaris9`)
- **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 [“Creating and Using Oracle Solaris Zones”](#).
- **iSCSI support in non-global zones** – Neither iSCSI target nor initiator services are currently supported in non-global zones.
- **Network virtualization for zones** – Most Oracle Solaris network virtualization features can be applied to a zone by creating a virtual NIC (VNIC) for the zone and then applying bandwidth limits and traffic flows to the zone's assigned VNIC. The VNIC is created when the zone boots, deleted when the zone halts, and is created within the non-global zone's datalink name space. This feature enables you to provision a zone without knowing the details of the 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.
- **NFS server and CIFS support in non-global zones** – Any Oracle Solaris 11 non-global zone brand type can be an NFS server or an NFS client. However, an Oracle Solaris10 branded non-global zone cannot be an NFS server. Any Oracle Solaris 11 non-global zone can be a CIFS client, but no non-global zone of any brand type can be a CIFS server. Also, an Oracle Solaris10 branded non-global zone cannot be a CIFS client, except when using the non-native Solaris open source Samba package.
- **Oracle Solaris Kernel Zones** – Kernel zones, also known as `solaris-kz` branded zones, are new in Oracle Solaris 11.2. Before using this feature, review the information in [“Hardware and Software Requirements for Oracle Solaris Kernel Zones”](#) in [“Creating and Using Oracle Solaris Kernel Zones”](#).
- **Whole root zones only** – Oracle Solaris Zones are whole-root type only. However, you can configure zones in a more flexible way, for example, 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 that individual software stacks for each non-global zone are independent of the global zone.

- **Zone migration through ZFS Archives** – You can migrate an existing non-global zone from one system to another system by creating an archive of a zone and then attaching that archive to another system. See [“How to Migrate A Non-Global Zone Using ZFS Archives”](#) in [“Creating and Using Oracle Solaris Zones”](#) for instructions.
- **Zone monitoring** – System resources that are consumed by non-global zones can be monitored by using the `zonestat` command.

Oracle Solaris Zones Feature Enhancements

The following zones enhancements have been made:

- **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.
- **Multiple boot environments supported on Oracle Solaris 10 branded zones** – Multiple boot environments are supported on Oracle Solaris 10 branded zones. This change provides a greater degree of flexibility and safety for performing patching operations within an Oracle Solaris 10 environment that is running Oracle Solaris 11.
- **Parallel zone updates** – A system that has multiple Oracle Solaris zones is updated in parallel. The increase in speed for updating 20 zones is in the 4x range.
- **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** – You can simplify zones deployment, administration and migration by running the zones on arbitrary storage objects, for example, Fibre Channel devices or iSCSI targets. The shared storage feature enables you to transparently access and manage shared storage resources in zones. You can describe the corresponding shared storage resources in a host-independent format in the zone configuration.

For iSCSI targets, a URI type is used to describe the various storage devices that can be accessed through the iSCSI network-based storage protocol. See [“About Shared Storage Resources Using Storage URIs”](#) in [“Creating and Using Oracle Solaris Zones”](#).

Zones installations that use the shared storage feature are encapsulated into dedicated ZFS storage pools that are hosted on shared storage devices. You can configure a device path directly with the `zonecfg` command. The zone is automatically encapsulated into its own ZFS storage pool. See [Chapter 14, “Getting Started With Oracle Solaris Zones on Shared Storage,”](#) in [“Creating and Using Oracle Solaris Zones”](#).

See [“Introduction to Oracle Solaris Zones”](#).

Oracle Solaris 10 Branded Zones Preparation

Prepare for migrating a Oracle Solaris 10 OS instance or zone to your Oracle Solaris 11 system as follows:

- Confirm that your Oracle Solaris 10 instance or zone is running at least the Oracle Solaris 10 9/10 release, which is the minimum OS requirement. The root file system can be UFS or ZFS.

- 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 the 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:

<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)

Note - 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 a Non-Global Zone on an Oracle Solaris 11 System

To transition your Oracle Solaris 10 environment to a non-global zone on an Oracle Solaris 11 system, you can create a zone archive, then migrate the zone archive to the Oracle Solaris 11 system.

The steps are as follows:

1. Install the Oracle Solaris 10 zone package on your Oracle Solaris 11 system.

```
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: systema
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.

```
s11sysB# zfs create pond/s10archive
s11sysB# zfs set share.nfs.sec.default.root=s10sysA=on pond/s10archive
```

4. Select an Oracle Solaris 10 instance, which could be a virtual environment or a global zone on an Oracle Solaris 10 system. Note the Oracle Solaris 10 system's host id.

```
s10sysA# hostid
8439b629
```

5. Create an archive of the Oracle Solaris 10 instance that you plan 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
```

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” on page 149](#)
- [“Managing User Accounts” on page 150](#)
- [“User Environment Feature Changes” on page 153](#)
- [“Oracle Solaris Man Page Changes” on page 155](#)

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 “Assigning Rights to Users” in “Securing Users and Processes in Oracle Solaris 11.2”
usermod, groupmod, rolemod	Commands for modifying users, groups, and roles	“How to Modify a User Account” in “Managing User Accounts and User Environments in Oracle Solaris 11.2”
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 11.2” and userdel(1M)

Command/Tool Name	Description	For More Information
User Manager graphical user interface (GUI)	GUI for creating and managing users	groupdel(1M) , roledel(1M) Chapter 3, “Managing User Accounts by Using the User Manager GUI,” in “Managing User Accounts and User Environments in Oracle Solaris 11.2”

Managing User Accounts

In this release, you create and manage user accounts from the command line or by using the User Manager GUI. The GUI replaces some of the functionality of the Solaris Management Console and its associated command-line. See [“Managing User Accounts and User Environments in Oracle Solaris 11.2”](#).

User Account Management Changes

The following features are new or have changed in this release:

- **Creating user accounts** – User account creation has changed in the following ways:
 - 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.
 - Starting with Oracle Solaris 11.2, user names and group names can be up to 32 characters. The 8 character limitation no longer exists.
 - 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 optional server name specifies the host on which the home directory resides. Entries in this form depend on the automounter and are maintained in the `auto_home` map. The path `/home/username` is maintained in the `passwd` database. When a user subsequently references `/home/username`, the automounter mounts the specified directory on `/home/username`. You can disable the `autofs` service if you do not specify home directory path names that include a server name or `localhost`.
- **Modifying user accounts** – 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** – You can create roles 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 “Assigning Rights to Users” in “Securing Users and Processes in Oracle Solaris 11.2”.
- **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,” in “Managing User Accounts and User Environments in Oracle Solaris 11.2”.

User Password and Login Changes

User password management and login information have changed in the following ways:

- **Assuming a role** – All role assumption requires a password. In this release, at administrative discretion, the password that you supply to assume a role can be your own password.
- **Expanded login options during shutdown** – During a system shutdown, an `/etc/nologin` file is created. This file displays a message that the system is being shut down and that logins are not possible. However, this type of shutdown does not prevent superuser from logging into the system. In this release, users who are assigned the `root` role and users who are assigned the `solaris.system.maintenance` authorization are also not blocked when the `nologin` file is present on the system.
- **Failed login count notification** – The system 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.

- **Monitoring and restricting root access** – In a default system configuration, a user cannot remotely log in to as root. When logging in remotely, users must log in with their user name and then become root by using the su command. You can monitor who has been using the su command, as well as restrict root access to a system. See “[Monitoring and Restricting root Access](#)” in “[Securing Systems and Attached Devices in Oracle Solaris 11.2](#)”
- **Password hashing algorithm** – The default password hashing algorithm in this release is SHA256. This password hash is similar to the following:

```
$5$cgQk2iUy$AhHtVGx5Qd0.W3NCKj i kb8.Kh0iA4Dpxsw55sP0UnYD
```

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 be much longer than eight characters. See [policy.conf\(4\)](#).

- **root password changes** – It is no longer possible to use a system without assigning the root role a password of the requisite length that also meets other password complexity requirements.
- **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 in the following ways:

LK	The account is locked for UNIX authentication. The <code>passwd -l</code> 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 <code>passwd -N</code> command was run.
----	--

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.

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 11.2”](#).

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 11.2”](#).

User Environment Feature Changes

Note the following user environment feature and command 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.
- **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.

- **Default login and other shell changes** - In Oracle Solaris 10, the default scripting shell (/bin/sh) is the Bourne shell. Starting with 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.

Note the following additional changes:

- 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 user path and PATH environment variable** – The default user path is /usr/bin. The default path for the root role is /usr/bin:/usr/sbin. The default PATH environment variable for bash is /usr/bin:/usr/sbin
- **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.
- **Editor changes** – The vi family of editors, including /usr/bin/vi, /usr/bin/view, and /usr/bin/ex, are links to the vim open source implementation of the vi editor. The traditional SunOS versions of these commands are available in /usr/sunos/bin/.
- **File locations**– Files that were previously in the /usr/sfw directory are now in /usr/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>.

To change the default version to Java 7, run the following command:

```
# pkg set-mediator -V 1.7 java
```

Note - If you install Java 8, it becomes your default Java version, *unless* you run the pkg set-mediator command that is shown in the previous example. For additional information, see “[Java Recommendations](#)” in “[Oracle Solaris 11.2 Release Notes](#)”.

- **MANPATH variable** – The MANPATH environment variable is no longer required. The man command determines the appropriate MANPATH, based on the PATH environment variable setting.

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 previously contained the Oracle Solaris man pages has changed to the smaller `system/manual` package. The bulk of the man pages are 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** – By default, man pages are installed on your Oracle Solaris system. If man pages on not displaying on your system, check whether the default value is set to `True` as follows:

```
$ pkg facet -a facet.doc.man
FACET VALUE SRC
facet.doc.man True system
```

Change the setting to `True` as follows

```
$ pkg change-facet facet.doc.man=True
```

If you do not want man pages to display on your system, you can toggle the default setting to `False` as follows:

```
$ pkg change-facet facet.doc.man=False
```

Note - Changing the default setting from `True` to `False` removes all of the man pages from the system and creates a backup BE. The backup BE still has man pages, but the newly created BE does not.

Managing the Oracle Solaris Desktop

This chapter describes the desktop features that are supported in Oracle Solaris 11 releases. The following topics are covered:

- [“Oracle Solaris Desktop Features” on page 157](#)
- [“Desktop Features That Have Been Removed” on page 160](#)
- [“Xorg Family of Servers” on page 161](#)
- [“Troubleshooting Desktop Transition Issues” on page 162](#)

Oracle Solaris Desktop Features

The default desktop environment in this release is 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. See [“Installing the Oracle Solaris Desktop Software Package After an Installation” on page 162](#).

Other supported 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 are key desktop features in this release:

- **Accessibility enhancements** – Users with disabilities can use a wide range of accessibility features, including Orca, espeak, and brltty. These features replace gnomercus 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 in Oracle Solaris 10 is no longer available. Use the Dasher application 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. If necessary, install the package as follows:

```
# pkg install cmdassist
```

- **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. To learn more, consult the `gdm` and `console-kit-daemon` man pages. ConsoleKit configuration features are used to manage multi-seat environments in this release. To troubleshoot transition issues, see [“GNOME Desktop Manager Issues” on page 163](#).

- **Multimedia support:**
 - **Brasero CD/DVD burner** – You can use the Brasero CD/DVD burner to create a data disc project, drag and drop files in place, and then burn them.
 - **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** – The PulseAudio sound server supports improved audio mixing. The `/usr/bin/gnome-volume-control` Device combo-box displays additional PulseAudio devices. For desktop and notebook PC use, 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 plugins 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. Go to <http://www.freedesktop.org/wiki/Software/PulseAudio>.

- **Other media tools** – Rhythmbox media player, Cheese photo/video tool, the Ekiga video conference tool are also included in this release.
- **Network administration GUI** – You use the network administration GUI (formerly NWAM) to manage network connections from the desktop. See “[Administering Network Configuration From the Desktop](#)” in “[Configuring and Administering Network Components in Oracle Solaris 11.2](#)”.
- **Print management** – Starting with Oracle Solaris 11, CUPS is the default print service, replacing the LP print service that is used in Oracle Solaris 10. Solaris Print Manager is also no longer available in the desktop. CUPS has a print manager that you can start from the desktop by choosing System → Administration → Print Manager. See “[Configuring and Managing Printing in Oracle Solaris 11.2](#)”.
- **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 `vol` daemon and the `volcheck` command are 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 supported in this release. 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** – In this release, the Trusted Extensions feature of Oracle Solaris is 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 rights profile improvements. For example, the `txzonemgr` GUI has significantly improved. You can use this tool 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.

- **Time Slider** – Manages ZFS snapshots. Use the tool to regularly back up data by taking timed ZFS snapshots.
- **Virtual console terminals**– You can 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** – The Firefox and Thunderbird applications are supported.

Desktop Features That Have Been Removed

The following desktop features have been replaced or removed. Note that some removed features were introduced later than Oracle Solaris 10:

- **Adobe Flash Player** – This feature was present in Oracle Solaris 11 11/11, 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** – You can use the Dasher application as a replacement in some instances.

- GNOME System tools (Introduced in a previous Oracle Solaris 11 release):
 - `network-admin` – NWAM replaces this feature. Starting with Oracle Solaris 11.1, this tool is renamed to the Network Administration GUI.
 - `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 149](#).

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, starting with Oracle Solaris 11.1. See [“Commands and Tools for Managing User Accounts” on page 149](#).
- Solaris Print Manager – This tool is replaced by CUPS Print Manager. See [“Printer Configuration and Management Changes” on page 122](#).
- Xsun family of servers – The Xorg family of servers is still supported. See [“Xorg Family of Servers” on page 161](#).

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
<code>/usr/openwin/bin/Xsun</code>	<code>/usr/bin/Xorg</code> See Xorg(1)
<code>/usr/openwin/bin/Xnest</code>	<code>/usr/bin/Xephyr</code> See Xephyr(1)
<code>/usr/openwin/bin/Xvfb</code>	<code>/usr/bin/Xvfb</code> See Xvfb(1)

▼ How to Update Custom Hot Key Configurations or Enable Legacy Mappings

Oracle Solaris 11 has moved to more common Xorg key mappings. For example, the **Copy** key is mapped to XF86Copy.

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 a Text Installation”](#) in [“Installing Oracle Solaris 11.2 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 in Oracle Solaris 10. The 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 GDM login GUI by modifying the `/usr/share/gdm/gdm-greeter-login-window.ui` file, as desired.

