

Transitioning From Oracle® Solaris 10 to Oracle Solaris 11

Copyright © 2012, 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 RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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.

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 RIGHTS. Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

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

Preface	9
1 Transitioning From Oracle Solaris 10 to Oracle Solaris 11 (Overview)	13
Welcome to Oracle Solaris 11	13
Oracle Solaris 10 Compared to Oracle Solaris 11	14
Removal of Legacy System Management Commands, Tools, Services, and Files	17
Transitioning Your Oracle Solaris 10 System to Oracle Solaris 11	19
Installation Features	20
Software Management Features	21
Network Configuration Features	22
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	25
Desktop Features	25
2 Transitioning to an Oracle Solaris 11 Installation Method	27
Oracle Solaris 11 Installation Methods	27
Oracle Solaris 11 Installation Requirements	28
ZFS Root Pool Installation Requirements	28
Oracle Solaris 11 Preinstallation Tasks	29
Installing Oracle Solaris 11 by Using Installation Media	30
Migrating From JumpStart to AI	31
Using the JumpStart Migration Utility	32
JumpStart to AI Conversion Tasks	32
Installing Oracle Solaris 11 by Using AI	33

Setting Up an Install Server	33
Customizing an AI Installation	34
Provisioning a Client System	35
Configuring a Client System	35
Creating a Script That Runs at First Boot	36
Installing Client Systems by Using AI	36
Additional Installation Tasks	38
Configuring Date and Time Before and After an Installation	38
Adding a Linux Entry to the GRUB Menu After an Installation	39
Troubleshooting an Oracle Solaris 11 Installation	40
Monitoring the Live Media Startup Process	40
Troubleshooting Login and User Account Issues After an Installation	40
3 Managing Devices	43
Device Identity and Configuration Changes	43
Changes to Device Driver Customization	45
Preparing Disks for ZFS Storage Pools	45
ZFS Root Pool Installation Improvements	45
ZFS Root Pool Device Requirements	46
Swap and Dump Device Configuration Changes	47
4 Managing Storage Features	49
Comparison of Solaris Volume Manager Configurations to ZFS Configurations	49
Recommended ZFS Storage Pool Practices	50
ZFS Storage Pool Creation Practices	50
ZFS Storage Pool Monitoring Practices	52
ZFS Storage Pool Troubleshooting Practices	52
COMSTAR Replaces iSCSI Target Daemon	53
5 Managing File Systems	55
Oracle Solaris 11 File System Changes	55
Root File System Requirements and Changes	56
Managing ZFS File System Changes	56
Displaying ZFS File System Information	57

Making ZFS File Systems Available	59
ZFS File System Sharing Changes	59
ZFS Data Deduplication Requirements	61
Considering ZFS Backup Features	62
Migrating File System Data to ZFS File Systems	62
Recommended Data Migration Practices	62
Migrating Data With ZFS Shadow Migration	63
Migrating UFS Data to a ZFS File System (ufsdump and ufsrestore)	63
6 Managing Software	65
Oracle Solaris 11 Package Changes	65
Oracle Solaris 10 SVR4 and IPS Package Comparison	66
IPS Installation Package Groups	68
Displaying Information About Software Packages	68
Updating the Software on Your Oracle Solaris 11 System	70
Installing Maintenance Updates on an Oracle Solaris 11 System	71
Managing Boot Environments	72
Tools for Managing Boot Environments	72
▼ How to Update Your ZFS Boot Environment	74
7 Managing Network Configuration	75
How the Network Is Configured in Oracle Solaris 11	75
How the Network Is Configured During an Installation	76
Managing Network Configuration When in Manual Mode	78
Configuring Network Interfaces When in Manual Mode	79
Displaying and Configuring Link Interfaces When in Manual Mode	80
Configuring Naming Services When in Manual Mode	81
resolv.conf Error-Checking Capabilities	83
Temporarily Resetting SMF Naming Services	83
Importing Network Configuration	84
Configuring LDAP When in Manual Mode	85
Managing Network Configuration When in Automatic Mode	85
Configuring Naming Services When in Automatic Mode	86
Configuring LDAP When in Automatic Mode	87
Managing Automatic Network Configuration From the Desktop	87

Commands That Are Used for Network Configuration (Quick Reference)	88
8 Managing System Configuration	91
Comparing Oracle Solaris 10 and Oracle Solaris 11 System Configuration Tools	91
System Configuration Changes and Migration of System Configuration to SMF	93
SMF Administrative Changes	94
System Process Changes	95
System Console, Terminal Services, and Power Management Changes	96
Power Management Configuration	96
System Configuration Tools Changes	97
System Registration Changes	98
System Boot, Recovery, and Platform Changes	98
Booting for System Recovery Changes	99
Boot, Platform, and Hardware Changes	102
Printer Configuration and Management Changes	103
Removal of the LP Print Service	103
▼ How to Set Up Your Printing Environment After Installing Oracle Solaris 11	104
Internationalization and Localization Changes	105
Locale and Time Zone Configuration Changes	107
9 Managing Security	109
Security Feature Changes	109
Network Security Features	110
Removed Security Features	111
Roles, Rights, Privileges, and Authorizations	111
Viewing Your Privileges	113
File and File System Security Changes	114
acLmode Property Is Reintroduced	114
Encrypting ZFS File Systems	115
Immutable Zones	116
10 Managing Oracle Solaris Releases in a Virtual Environment	117
Installing and Managing Oracle Solaris 11 Virtualization Features	117
Oracle Solaris 11 Zone Features	118

Oracle Solaris 10 Branded Zones Preparation	120
Transitioning an Oracle Solaris 10 Instance to an Oracle Solaris 11 System	120
11 User Account Management and User Environment Changes	123
Tools for Managing User Accounts, Groups, and Roles	123
Creating and Managing User Accounts, Groups, and Roles	124
User Passwords	124
Creating User Accounts	125
Modifying User Accounts	125
Creating Roles	125
Sharing Home Directories That Are Created as ZFS File Systems	126
How Home Directories Are Mounted in Oracle Solaris 11	126
User Environment Feature Changes	127
Default Login Shell and PATH Environment Variable	127
Oracle Solaris 11 Man Page Changes	127
12 Using Oracle Solaris Desktop Features	129
Oracle Solaris 11 Desktop Feature Summary	129
Key Desktop Features	130
Desktop Features That Have Been Removed	132
Xorg Family of Servers	133
X Server Key Maps	133
Troubleshooting Desktop Transition Issues	134
Installing the Oracle Solaris Desktop Software Package After an Installation	134
GNOME Desktop Manager Issues	134
A Transitioning From Previous Oracle Solaris 11 Releases to Oracle Solaris 11	137
Solaris 11 Express to Solaris 11 Transition Issues	137
Network Configuration Changes	137
Naming Service Configuration Changes	138
Printing Setup Changes	138
Device Driver Customization Changes	139
Root File System Changes	139
File System Sharing Changes	139

Default Path Changes 140

Preface

Transitioning From Oracle Solaris 10 to Oracle Solaris 11 covers topics for transitioning from Oracle Solaris 10 to Oracle Solaris 11, as well as provides some information about transitioning from Oracle Solaris 11 Express and earlier Oracle Solaris 11 releases.

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

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

Who Should Use This Book

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

How the System Administration Guides Are Organized

Here is a list of the topics that are covered by the System Administration Guides.

Book Title	Topics
<i>Booting and Shutting Down Oracle Solaris on SPARC Platforms</i>	Booting and shutting down a system, managing boot services, modifying boot behavior, booting from ZFS, managing the boot archive, and troubleshooting booting on SPARC platforms
<i>Booting and Shutting Down Oracle Solaris on x86 Platforms</i>	Booting and shutting down a system, managing boot services, modifying boot behavior, booting from ZFS, managing the boot archive, and troubleshooting booting on x86 platforms

Book Title	Topics
<i>Oracle Solaris Administration: Common Tasks</i>	Using Oracle Solaris commands, booting and shutting down a system, managing user accounts and groups, managing services, hardware faults, system information, system resources, and system performance, managing software, printing, the console and terminals, and troubleshooting system and software problems
<i>Oracle Solaris Administration: Devices and File Systems</i>	Removable media, disks and devices, file systems, and backing up and restoring data
<i>Oracle Solaris Administration: IP Services</i>	TCP/IP network administration, IPv4 and IPv6 address administration, DHCP, IPsec, IKE, IP Filter, and IPQoS
<i>Oracle Solaris Administration: Naming and Directory Services</i>	DNS, NIS, and LDAP naming and directory services, including transitioning from NIS to LDAP
<i>Oracle Solaris Administration: Network Interfaces and Network Virtualization</i>	Automatic and manual IP interface configuration including WiFi wireless; administration of bridges, VLANs, aggregations, LLDP, and IPMP; virtual NICs and resource management.
<i>Oracle Solaris Administration: Network Services</i>	Web cache servers, time-related services, network file systems (NFS and autofs), mail, SLP, and PPP
<i>Oracle Solaris Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management</i>	Resource management features, which enable you to control how applications use available system resources; Oracle Solaris Zones software partitioning technology, which virtualizes operating system services to create an isolated environment for running applications; and Oracle Solaris 10 Zones, which host Oracle Solaris 10 environments running on the Oracle Solaris 11 kernel
<i>Oracle Solaris Administration: Security Services</i>	Auditing, device management, file security, BART, Kerberos services, PAM, Cryptographic Framework, Key Management Framework, privileges, RBAC, SASL, Secure Shell and virus scanning.
<i>Oracle Solaris Administration: SMB and Windows Interoperability</i>	SMB service, which enables you to configure an Oracle Solaris system to make SMB shares available to SMB clients; SMB client, which enables you to access SMB shares; and native identity mapping service, which enables you to map user and group identities between Oracle Solaris systems and Windows systems
<i>Oracle Solaris Administration: ZFS File Systems</i>	ZFS storage pool and file system creation and management, snapshots, clones, backups, using access control lists (ACLs) to protect ZFS files, using ZFS on an Oracle Solaris system with zones installed, emulated volumes, and troubleshooting and data recovery
<i>Trusted Extensions Configuration and Administration</i>	System installation, configuration, and administration that is specific to Trusted Extensions

Book Title	Topics
<i>Oracle Solaris 11 Security Guidelines</i>	Securing an Oracle Solaris system, as well as usage scenarios for its security features, such as zones, ZFS, and Trusted Extensions
<i>Transitioning From Oracle Solaris 10 to Oracle Solaris 11</i>	Provides system administration information and examples for transitioning from Oracle Solaris 10 to Oracle Solaris 11 in the areas of installation, device, disk, and file system management, software management, networking, system management, security, virtualization, desktop features, user account management, and user environments

Access to Oracle Support

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

Typographic Conventions

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

TABLE P-1 Typographic Conventions

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

Shell Prompts in Command Examples

The following table shows the default UNIX system prompt and superuser prompt for shells that are included in the Oracle Solaris OS. Note that the default system prompt that is displayed in command examples varies, depending on the Oracle Solaris release.

TABLE P-2 Shell Prompts

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

General Conventions

Be aware of the following conventions used in this book.

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

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

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

This is a list of the information that is in this chapter:

- “Welcome to Oracle Solaris 11” on page 13
- “Oracle Solaris 10 Compared to Oracle Solaris 11” on page 14
- “Transitioning Your Oracle Solaris 10 System to Oracle Solaris 11” on page 19
- “Installation Features” on page 20
- “Software Management Features” on page 21
- “Network Configuration Features” on page 22
- “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 25
- “Desktop Features” on page 25

Welcome to Oracle Solaris 11

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

For information about transitioning from Oracle Solaris 11 Express to Oracle Solaris 11, see [Appendix A, “Transitioning From Previous Oracle Solaris 11 Releases to Oracle Solaris 11.”](#) For information about a particular feature, refer to the product documentation.

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

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

See also the documentation at:

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

Note that this guide does not provide information about every new feature in Oracle Solaris 11, nor does it mention every feature that is excluded from Oracle Solaris 11. Also, the guide does not provide detailed information about every feature that has changed since Oracle Solaris 11 Express.

- For information about new features, see <http://www.oracle.com/technetwork/server-storage/solaris11/documentation/solaris11-whatsnew-201111-392603.pdf>.
- For information about excluded features, see <http://www.oracle.com/technetwork/systems/end-of-notices/eonsolaris11-392732.html>.
- For information about upgrading from Oracle Solaris 11 Express or an Oracle Solaris 11 Early Adopter release, see [Oracle Solaris 11 Release Notes](#).
- For information about Oracle's Sun hardware platforms and the corresponding Oracle Solaris operating system requirements, go to <http://www.oracle.com/technetwork/systems/software-stacks/stacks/index.html>.

Oracle Solaris 10 Compared to Oracle Solaris 11

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

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

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Additional Information
Booting from a root device	From a UFS or Solaris Volume Manager root device	From an Oracle Solaris ZFS root file system	“System Boot, Recovery, and Platform Changes” on page 98

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

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Additional Information
Booting from the network	From SPARC ok PROM prompt: boot net[:dhcp] or boot net[:rarp] x86: Requires a DHCP server that supports a Preboot Execution Environment (PXE) boot from the network.	SPARC: boot net:dhcp x86: No change since Oracle Solaris 10	“Boot, Platform, and Hardware Changes” on page 102
Booting (recovery)	SPARC: Boot in failsafe mode by typing boot -F failsafe at the ok PROM prompt. x86: Boot in failsafe mode by selecting the failsafe entry in the GRUB menu at boot time.	Failsafe mode is not supported on SPARC or x86 platforms. Depending on the error condition, boot in single-user mode or perform system recovery steps.	“System Boot, Recovery, and Platform Changes” on page 98
Desktop environment	Common Desktop Environment (CDE)	Oracle Solaris Desktop (GNOME 2.30)	“User Account Management and User Environment Features” on page 25
File systems (default)	UFS and Solaris Volume Manager root file systems	ZFS root file system (default)	Chapter 5, “Managing File Systems”
Installation (graphical user interface (GUI))	GUI installation program on DVD or CD	Live Media (x86 only)	“Installing Oracle Solaris 11 by Using Installation Media” on page 30
Installation (interactive text)	Interactive text installation and interactive text installer for ZFS root pools	Text installer (stand-alone and network installation)	“Installing Oracle Solaris 11 by Using Installation Media” on page 30
Installation (automated)	JumpStart feature of Oracle Solaris 10	Automated Installer (AI) feature of Oracle Solaris 11	“Installing Oracle Solaris 11 by Using AI” on page 33
Installation (other)	Oracle Solaris Flash Archive installation	See “System Boot, Recovery, and Platform Changes” on page 98.	<i>Oracle Solaris 11 Release Notes</i>
Network configuration (manual and automatic)	ifconfig Edit /etc/hostname.* ndd for configuring protocols	manual mode: dladm and ipadm automatic mode: netcfg and netadm	Chapter 7, “Managing Network Configuration”

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

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Additional Information
Network configuration (DHCP)	Oracle Solaris DHCP and other naming services	Internet Systems Consortium (ISC) DHCP and legacy Sun DHCP	Chapter 10, “About DHCP (Overview),” in <i>Oracle Solaris Administration: IP Services</i>
Network configuration (wireless)	wificonfig	manual: dladm and ipadm automatic: netcfg	Chapter 7, “Managing Network Configuration”
Packaging (software management)	SVR4 package and patch commands	Image Packaging System (IPS) pkg(1) commands, Package Manager and Update Manager GUIs	Chapter 6, “Managing Software”
Print service (default), printer configuration and administration	LP print service, lp print commands, Solaris Print Manager GUI	Common UNIX Print System (CUPS), CUPS command-line tools, CUPS Print Manager GUI	“Printer Configuration and Management Changes” on page 103
Security management	root as a user account	root as a role	Chapter 9, “Managing Security”
System configuration and reconfiguration	sysidtool, sys-unconfig, sysidconfig, and sysidcfg	sysconfig, System Configuration Interactive (SCI) tool, SC profiles	Chapter 8, “Managing System Configuration”
System configuration (naming services configuration)	Configured in files within /etc and /var	Managed through the Service Management Facility (SMF) feature of Oracle Solaris	“System Configuration Changes and Migration of System Configuration to SMF” on page 93
System configuration (nodename or host name)	Edit /etc/nodename	svccfg -s sets the config/nodename property of the svc:system/identity:node service to the desired name. Note – If the system is configured to use DHCP, which is always the case when the Automatic NCP is enabled, the SMF service property can only be set <i>if</i> the DHCP server does not provide a value for the nodename/hostname option. See <code>nodename(4)</code> .	Chapter 8, “Managing System Configuration”

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

Feature or Command	Oracle Solaris 10	Oracle Solaris 11	Additional Information
System registration	Auto Registration	Oracle Configuration Manager	“System Registration Changes” on page 98
System upgrade and boot environment management	lu and SVR4 package commands	pkg commands, Package Manager, Update Manager beadm utility for managing boot environments	Chapter 6, “Managing Software”
User account management	Solaris Management Console GUI and associated command-line tools	useradd, usermod, userdel, groupadd, groupmod, groupdel, roleadd, rolemod, and roledel	“Creating and Managing User Accounts, Groups, and Roles” on page 124
User environment management	Korn shell (ksh) MANPATH variable is 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 is no longer required	“User Environment Feature Changes” on page 127

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

The following table describes legacy commands, tools, services, files that are no longer supported in Oracle Solaris 11.

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

Legacy Command, Tool, Service, or File	Replacement Command, Tool, Service, or File	For More Information
bsmconv and bsmunconv	audit	audit(1M)
crypt and des	encrypt	encrypt(1)
graph and spline	gnuplot	gnuplot(1) Note – Install the image/gnuplot package.

TABLE 1-2 System Management Commands, Tools, Services, and Files (Continued)

Legacy Command, Tool, Service, or File	Replacement Command, Tool, Service, or File	For More Information
localeadm	Locale facet mechanism	“Internationalization and Localization Changes” on page 105
download, lpfilter, lpforms, lpget, lpset, lpsched, lpshut, lpsystem, lpusers, printmgr, print-service (launches Solaris Print Manager), 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 103
LP print files and descriptions in: <ul style="list-style-type: none"> ■ ~/.printers ■ /etc/printers.conf ■ /usr/lib/lp/postscript ■ /usr/lib/print 	~/.lpoptions	lpoptions(1)
Legacy SMF print services: <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 Replacement SMF print services: <ul style="list-style-type: none"> ■ svc:/application/cups/scheduler ■ svc:/application/cups/in-lpd 		“Printer Configuration and Management Changes” on page 103
rdist	rsync or scp	rsync(1) and scp(1)
rstart and rstartd	ssh	ssh(1)
saf, sac, sacadm, nlsadmin, pmadm, ttyadm, and listen /usr/include/listen.h, getty, /usr/lib/saf/nlps_server, /var/saf, /etc/saf, ttymon (sac and getty modes <i>only</i>), and ports (sac functionality)	ttymon express mode is still supported by the following SMF services: <ul style="list-style-type: none"> ■ svc:/system/console-login:terma ■ svc:/system/console-login:termb 	“System Console, Terminal Services, and Power Management Changes” on page 96
pmconfig and /etc/power.conf	poweradm	poweradm(1M)
srosservice and smdiskless	Not available	
Solaris Management Console GUI, smc, smuser, smgroup, and passmgmt	useradd, usermod, userdel, groupadd, groupmod, groupdel, roleadd, rolemod, roledel	“Creating and Managing User Accounts, Groups, and Roles” on page 124

TABLE 1-2 System Management Commands, Tools, Services, and Files (Continued)

Legacy Command, Tool, Service, or File	Replacement Command, Tool, Service, or File	For More Information
sysidtool, sys-unconfig, and sysidcfg	sysconfig, SCI tool, and SC configuration through profiles	“System Configuration Tools Changes” on page 97
vold daemon	vofls and rmvolmgr	Chapter 3, “Managing Devices”

For more information about legacy commands that are no longer supported, see *Oracle Solaris 11 Release Notes*.

Transitioning Your Oracle Solaris 10 System to Oracle Solaris 11

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

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

However, you can migrate your Oracle Solaris 10 OS instances or zones and your data to Oracle Solaris 11 systems. For more information, see [Table 1-3](#).

- The following Oracle Solaris 10 installation features are not available in Oracle Solaris 11: the Oracle Solaris installation upgrade option, the Oracle Solaris Flash Archive installation method, JumpStart, and the Oracle Solaris Live Upgrade feature (`lu` suite of commands).

The Automated Installer (AI) replaces JumpStart, and the `beadm` utility provides similar functionality to the `lu` commands. For more information, see “[Migrating From JumpStart to AI](#)” on page 31 and “[Tools for Managing Boot Environments](#)” on page 72.

- Oracle Solaris 11 introduces the Image Packaging System (IPS), which is a different mechanism than the legacy SVR4 package commands that are used in Oracle Solaris 10 and previous releases. See [Chapter 6, “Managing Software.”](#)

The following table describes the tools and features that are available for transitioning to Oracle Solaris 11.

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 and profiles to a format that is compatible with AI manifest entries.	Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11 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. Used to import a ZFS storage pool from an Oracle Solaris 10 system into an Oracle Solaris 11 system.	Chapter 5, “Managing File Systems”

Installation Features

The following installation methods are available:

- **x86: GUI installation with the Live Media** – The GUI installer can be used to install Oracle Solaris 11 on x86 platforms *only*. The GUI installer is capable of functioning with a minimum of 1536 MB of memory. See “[Installing Oracle Solaris 11 by Using Installation Media](#)” on page 30.
- **Interactive text installation (From media or over the network)** – The text installer enables you to install Oracle Solaris on SPARC and x86 based systems from media or over a network.
- **Automated installation on single or multiple systems** – The Automated Installer (AI) installs Oracle Solaris 11 on a single or multiple client systems. Similar to JumpStart, AI provides a hands-free installation. You can also perform automated installations that boot from media. See “[Installing Oracle Solaris 11 by Using AI](#)” on page 33.
AI also supports the installation of zones. See “[Oracle Solaris 11 Zone Features](#)” on page 118.
- **Customized installation image creation with the Distribution Constructor** – The Distribution Constructor tool creates preconfigured installation images. See “[Oracle Solaris 11 Installation Methods](#)” on page 27.

These installation tools and methods are no longer available:

- **Oracle Solaris Flash Archive Installation** – For information about recovering from a system failure, see [“System Boot, Recovery, and Platform Changes”](#) on page 98.
- **JumpStart feature of Oracle Solaris** – AI replaces JumpStart in this release. See [“Installing Oracle Solaris 11 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 also no longer supported. The `beadm` utility provides similar capability. See [“Tools for Managing Boot Environments”](#) on page 72.

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 download new or updated software packages and install them on your system. With IPS commands, you can list, search, install, update, and remove software packages.

Software management in Oracle Solaris 11 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.
- **Package Manager and Update Manager GUIs** – Package Manager and Update Manager are GUI versions of the IPS command-line tools that can be accessed from the desktop. Both tools can be used to install and update software packages.
- **IPS repositories** – An *IPS repository* is like a package archive, where clients can publish and retrieve packages. Several Oracle package repositories are available. See [Copying and Creating Oracle Solaris 11 Package Repositories](#).
- **Boot environment management** – The `beadm` utility is used to create and manage boot environments.

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

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

Network Configuration Features

Network configuration in Oracle Solaris 11 includes the following key features:

- **Profile-based network configuration** – Network configuration is profile-based. There are two configuration modes: manual (fixed) and automatic (reactive). The switch between network configuration modes no longer takes place at the service level (as in Oracle Solaris 11 Express), but at the profile level.
- **Single network SMF service** – The `svc:/network/physical:default` SMF service manages network configuration for both the manual and automatic network configuration modes. The output of the `svcs -a` command shows this service as disabled.
- **Naming services configuration through SMF** – The primary repository for the configuration of all naming services is now the SMF repository. Previous behavior that required you to modify a configuration file to make specific changes to a naming service configuration no longer works.
- **Generic datalink name assignment** – Generic names are automatically assigned to datalinks using the `net0`, `net1`, `netN` naming convention, depending on the total number of network devices that are on the system.
- **Support for virtual network interface cards (VNICs)** – VNICs are new, pseudo interfaces that you create on top of datalinks. Along with virtual switches, VNICs are the building blocks of a virtual network. You can create and modify VNICs in a system or in a zones environment. See [Part III, “Network Virtualization and Resource Management,” in *Oracle Solaris Administration: Network Interfaces and Network Virtualization*](#).

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

System Configuration and SMF Features

The following system configuration and SMF features are supported in Oracle Solaris 11:

- **System, network, and naming services configuration migration to SMF** – Several aspects of system and network configuration, including configuration that was previously stored in the `/etc` directory, is now stored in an SMF repository. Moving configuration data to SMF service properties enables the delivery of a uniform, extensible architecture for system configuration that provides customers with a more complete capability for managing system configuration. See [“System Configuration Changes and Migration of System Configuration to SMF” on page 93](#).
- **System console and terminal device management** – The system console and locally connected terminal devices are now managed by SMF. The `sac` and `saf` programs for managing console services and terminals are no longer available.

- **SMF administrative layers** – Information for recording the source of properties, property groups, instances, and services has been added to the SMF repository. This information enables you to determine which settings are administrative customizations and which settings were delivered with Oracle Solaris by a manifest. See [“SMF Administrative Changes” on page 94](#).
- **System Configuration Utility** – Uses SMF to centralize configuration information. The `sysconfig` utility replaces the `sys-unconfig` and `sysidtool` utilities that are used in Oracle Solaris 10 to unconfigure and reconfigure an Oracle Solaris 11 instance. The `sysconfig` utility can be run interactively by using the System Configuration Interactive (SCI) tool or in an automated manner by creating an SC configuration profile. See [“System Configuration Tools Changes” on page 97](#).
- **System registration through Oracle Configuration Manager** – Oracle Configuration Manager collects configuration information and then uploads it to the Oracle repository during the first reboot of a system after an installation. This information is used by Oracle to provide better service to customers. In Oracle Solaris 10, the Auto Registration feature performs a similar function. See [“System Registration Changes” on page 98](#).

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

Storage and File Systems Features

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

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

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

- **Migrate your UFS data with ZFS shadow migration** – The ZFS shadow migration feature is used to migrate data from an existing file system to a new file system. You can either migrate a local file system to a new file system or migrate an NFS file system to a new local file system. For more information, see [“Transitioning Your Oracle Solaris 10 System to Oracle Solaris 11”](#) on page 19.
- **Migrate your Oracle Solaris 10 storage pools** – Storage devices that contain ZFS storage pools on your Oracle Solaris 10 systems can be exported, disconnected, if necessary, and imported into your Oracle Solaris 11 systems.
- **Migrate your UFS data** – You can remotely mount UFS file systems from an Oracle Solaris 10 system on an Oracle Solaris 11 system. In addition, a `ufsdump` of your UFS data can be restored into a ZFS file system by using the `ufsrestore` command.

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

Security Features

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

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

For more information about these changes, see [Chapter 9, “Managing Security.”](#)

Virtualization Features

Oracle Solaris 11 introduces support for the following virtualization features:

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

For more information about these changes, see [Chapter 10, “Managing Oracle Solaris Releases in a Virtual Environment.”](#)

User Account Management and User Environment Features

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

Some of the key changes include the following:

- **Tools for creating and managing user accounts** – The Solaris Management Console GUI and all of its related command-line tools, for example, `smc` and `smuser`, are no longer available. User accounts are managed solely by using command-line tools, such as the `useradd`, `usermod`, and `userdel` commands.
- **Default user shell and path** – Oracle Solaris 11 provides more familiarity with Linux and Berkeley Software Distribution (BSD) operating systems. As a result, the default user shell and path have been changed, See [“Default Login Shell and PATH Environment Variable” on page 127](#).
- **Where administration commands are located** – Administration commands have moved from `/sbin` to `/usr/sbin`.
- **Where development tools are located** – Development tools that were previously in `/usr/ccs/bin` have moved to `/usr/bin`.

See [Chapter 11, “User Account Management and User Environment Changes.”](#)

Desktop Features

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

Note – In Oracle Solaris 11 the login manager has changed from CDE to the GNOME Desktop Manager (GDM). If you are transitioning from Oracle Solaris 10 to Oracle Solaris 11, and you previously customized your CDE login, review your display management configuration, as you might need to make some changes to the GDM configuration to ensure it works as expected. For more information, see [“Troubleshooting Desktop Transition Issues” on page 134](#).

See [Chapter 12, “Using Oracle Solaris Desktop Features.”](#)

Transitioning to an Oracle Solaris 11 Installation Method

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

If you are upgrading your system from Oracle Solaris 11 Express, see [Oracle Solaris 11 Release Notes](#).

This is a list of the information that is in this chapter:

- “Oracle Solaris 11 Installation Methods” on page 27
- “Oracle Solaris 11 Installation Requirements” on page 28
- “Installing Oracle Solaris 11 by Using Installation Media” on page 30
- “Migrating From JumpStart to AI” on page 31
- “Installing Oracle Solaris 11 by Using AI” on page 33
- “Additional Installation Tasks” on page 38
- “Troubleshooting an Oracle Solaris 11 Installation” on page 40

Oracle Solaris 11 Installation Methods

The following table summarizes the installation methods that are available in Oracle Solaris 11. With the exception of the Live Media, which is for x86 platforms only, all methods apply to both SPARC and x86 platforms.

TABLE 2-1 Available Installation Methods in Oracle Solaris 11

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

TABLE 2-1 Available Installation Methods in Oracle Solaris 11 (Continued)

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

Oracle Solaris 11 Installation Requirements

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

ZFS Root Pool Installation Requirements

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

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

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

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

- Disk space** – At least 13 GB of disk space is recommended. The space is consumed as follows:
 - Swap area and dump device** – The default sizes of the swap and dump volumes created by the Oracle Solaris installation programs vary, based on the amount of memory that is on the system and other variables.

After installation, you can adjust the sizes of your swap and dump volumes to the sizes of your choosing, as long as the new sizes support system operation. See “[Managing Your ZFS Swap and Dump Devices](#)” in *Oracle Solaris Administration: ZFS File Systems*.
 - Boot environment (BE)** – A ZFS BE is approximately 4–6 GB in size. Each ZFS BE that is cloned from another ZFS BE does not need additional disk space. Consider that the BE size will increase when it is updated, depending on the amount of updates. 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 Systems*.

Oracle Solaris 11 Preinstallation Tasks

Before installing Oracle Solaris 11, review the following information:

- **x86: Prepare the boot environment (applies to x86 based systems that will run multiple operating systems)** – See [“Preparing a Boot Environment for Installing Multiple Operating Systems”](#) in *Installing Oracle Solaris 11 Systems*.
- **Partition your OS disk** – If necessary, you can partition the disk that will contain the Oracle Solaris 11 OS before or during an installation. See [“Partitioning Your System”](#) in *Installing Oracle Solaris 11 Systems*.
- **Ensure that you have the proper device drivers** – Before installing Oracle Solaris 11, determine whether the devices on your system are supported. You can use either the Device Driver Utility or the Oracle Device Detection Tool 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 Systems*. See also the Hardware Compatibility Lists (HCL) at <http://www.oracle.com/webfolder/technetwork/hcl/index.html>.
- **x86: Configure the system's date and time (applies to x86 platforms that are installed with AI only)** – Oracle Solaris 11 keeps the Real Time Clock (RTC) in Coordinated Universal time (UTC) format. The behavior on x86 platforms is different than in Oracle Solaris 10 and Oracle Solaris 11 Express. AI does not adjust the RTC date and time during an installation. To configure the date and time in the BIOS, see [“Configuring Date and Time Before and After an Installation”](#) on page 38.
- **Decide whether to disable the default automatic registration of your system immediately after an installation (Optional)** – Oracle Configuration Manager automatically uploads anonymous system configuration information to My Oracle Support during the first reboot after an installation by using the Live Media or the text installer. If you choose, you can disable this feature before an installation and then re-enabled it later. See [“Using Oracle Configuration Manager”](#) in *Installing Oracle Solaris 11 Systems*.

Installing Oracle Solaris 11 by Using Installation Media

Oracle Solaris 11 can be installed from installation media by using any of the following installers or methods:

- **x86: Live Media**

The installer on the Live Media ISO image is for x86 platforms *only*. The Live Media contains software that is more appropriate for a desktop or laptop. Also, the Live Media requires more memory than the text installer. The exact memory requirements vary for each system. See “[Oracle Solaris 11 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 Systems*.

The GUI installer cannot upgrade your operating system. The default GUI installer settings are described in “[Default Settings With GUI Installer](#)” in *Installing Oracle Solaris 11 Systems*.

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

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

You can copy the downloaded image to removable media, such as a USB drive or burn it to a DVD by using the `usbcopy` utility. To use the `usbcopy` utility, you must first install the `pkg:/install/distribution-creator` package. See “[How to Perform a GUI Installation](#)” in *Installing Oracle Solaris 11 Systems*.

- **Interactive text installer**

The text installation media contains a set of software that is more appropriate for a general-purpose server. The text installer can perform an installation on an existing Oracle Solaris x86 partition or on a SPARC slice. Or, the text installation can use the entire disk. If the whole disk option is chosen, a partition or slice is created to cover the targeted device. In either case, the installation overwrites everything on the targeted partition or slice. See “[How to Perform a Text Installation](#)” in *Installing Oracle Solaris 11 Systems*. If you use the text installer, you might have to install additional software packages afterwards. See “[Adding Software After Text Installation](#)” in *Installing Oracle Solaris 11 Systems*.

If you are set up to perform an automated installation over the network, you can also perform an interactive text installation over the network. When using this method, you can only install a single system at a time. However, you can modify installation specifications by using the interactive selections. See “[How to Perform a Text Installation Over the Network](#)” in *Installing Oracle Solaris 11 Systems*.

- **Automated installations that boot from media**

You can boot an AI image from media or a USB device to initiate a hands-free installation of just that system. An AI manifest provides installation instructions for the system. See “[Creating a Custom AI Manifest](#)” in *Installing Oracle Solaris 11 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 Systems*.

For more information, see “Installing Oracle Solaris 11 by Using AI” on page 33.

You can also create custom Live Media images, text installer images, and AI images. See *Creating a Custom Oracle Solaris 11 Installation Image*.

Note – You cannot upgrade your system with any of the Oracle Solaris installers, but after installing the OS, you can update all of the packages on your system by using the pkg utility. See “Installing and Updating Packages” in *Oracle Solaris Administration: Common Tasks*.

The following installation features are no longer supported:

- **Oracle Solaris Flash Archive Installation** – You can restore root pool snapshots to recover from a failed system or devices. See “System Boot, Recovery, and Platform Changes” on page 98.
- **JumpStart feature of Oracle Solaris** – The Automated Installer replaces this feature. See *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11 Automated Installer*.

Migrating From JumpStart to AI

AI performs automated installations of single or multiple systems. This tool replaces JumpStart in this release. The following table describes the similarities and differences between AI and JumpStart. See also *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11 Automated Installer*.

TABLE 2-2 JumpStart Tasks Compared to AI Tasks

Task	JumpStart Command or Method	AI Command or Method
Set up an install server.	setup_install_server	installadm create-service
Add clients to the installation.	add_install_client	installadm create-client
Create derived profiles.	Use a begin script.	Use the derived manifest mechanism.
Specify installation instructions.	Use profile files.	Use AI manifests.
Provision the client.	Use DVDs.	Use a package repository on the Internet or a local network.

TABLE 2-2 JumpStart Tasks Compared to AI Tasks (Continued)

Task	JumpStart Command or Method	AI Command or Method
Specify client customizations.	Use rules files to associate clients with profile files.	Use <code>installadm</code> with the appropriate subcommands to associate clients with AI manifests and system configuration profiles.
Specify client configurations.	Use finish scripts and <code>sysidcfg</code> files.	Use an SMF system configuration profile file or scripts that are executed by an SMF service that runs once during the first boot.

Using the JumpStart Migration Utility

The `js2ai` utility converts Oracle Solaris 10 JumpStart rules and profiles to a format that is compatible with AI manifest entries. To use the `js2ai` utility, first install the `pkg:/install/js2ai` package. See the `js2ai(1M)` man page and [Chapter 2, “Converting Rules and Profile Files,”](#) in *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11 Automated Installer*.

JumpStart to AI Conversion Tasks

To migrate from JumpStart to AI, you must complete the following tasks. Some of these tasks can be performed by using the `js2ai` utility.

- **Specify AI client criteria.**

You can specify AI client criteria as follows:

- By using the `installadm add-manifest` or `installadm create-profile` command with the `-c` option, or by using the `installadm set-criteria` command with the `-c` or `-a` options.
- By using the `installadm add-manifest` command to specify criteria files, by using the `installadm create-profile` command, or by using the `installadm set-criteria` command with the `-C` option.

For example, the `karch i86pc` JumpStart keyword is specified as an AI criteria directive in command-line format as follows:

```
# installadm add-manifest -c arch=i86pc
```

The same directive is specified in a criteria file as follows:

```
<ai_criteria name="arch">
  <value>i86pc</value>
</ai_criteria>
```


See Chapter 9, “Customizing Installations,” in *Installing Oracle Solaris 11 Systems*.

- **Convert a JumpStart profile to an AI manifest.**
AI uses XML manifest files to define client installation information. Use the `js2ai` utility to convert both JumpStart rules and their associated profiles to AI criteria and manifests. See “Customizing an AI Installation” on page 34.
- **Convert the `sysidcfg` configuration to system configuration (SC) profile specifications.**
The `sysidcfg` file keywords must be converted to SC profile specifications. AI uses SMF XML profile files to specify this configuration. See Chapter 10, “Provisioning the Client System,” in *Installing Oracle Solaris 11 Systems* and `ai_manifest(4)`. For information about migrating from JumpStart to AI, see *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11 Automated Installer*.

Installing Oracle Solaris 11 by Using AI

If you are transitioning to an AI installation method, refer to the following information, which provides a basic overview of each process and task that is required to set up and install systems by using AI.

Setting Up an Install Server

An Oracle Solaris 11 server can be used to install Oracle Solaris 11 by using AI or to install Oracle Solaris 10 by using JumpStart. You can set up a JumpStart install server on an Oracle Solaris 11 system. However, you can *only* use JumpStart to install Oracle Solaris 10. See “Setting Up an Oracle Solaris 11 System as an Oracle Solaris 10 JumpStart Server” in *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11 Automated Installer*.

You can use the same Oracle Solaris 11 system as both a JumpStart install server and an AI install server. See Chapter 4, “Installing Oracle Solaris 10 Using JumpStart on an Oracle Solaris 11 Server,” in *Transitioning From Oracle Solaris 10 JumpStart to Oracle Solaris 11 Automated Installer*.

The following tasks are required to set up an AI install server:

- **Check system requirements** – See “Install Server Requirements” in *Installing Oracle Solaris 11 Systems*.
- **Install the AI installation tools package** – To install the tools package, you must have access to a networked IPS package repository. Determine whether the `installadm` package is already installed by typing the following command:

```
$ pkg list installadm
```

Next, ensure that the IPS package repository contains the `installadm` package:

```
$ pkg list -a installadm
```

- **Configure the install server** – The configuration that you can perform on the install server to prepare for AI client installations is described in [“Configure the Install Server” in *Installing Oracle Solaris 11 Systems*](#).
- **Create an AI installation instance** – An install server can have more than one install service. A separate install service is created for each client hardware architecture and for each version of Oracle Solaris 11 that you want to install. See [“Create an AI Install Service” in *Installing Oracle Solaris 11 Systems*](#). See also [“Automated Installer Use Cases” in *Installing Oracle Solaris 11 Systems*](#).

Customizing an AI Installation

The following tasks make up an AI customization:

- **Match clients with installation and configuration instructions** – When using AI, you must set up a DHCP server and an install server. The install server must have at least one AI boot image and an AI install service that is associated with that boot image. When the client boots, DHCP directs the client to the install server. The client uses the default install service for that client architecture. Or, the client uses an assigned install service. The install service uses the methods that are described in [Chapter 9, “Customizing Installations,” in *Installing Oracle Solaris 11 Systems*](#) to match the client with the correct installation and configuration instructions.
- **Select the AI manifest to be used by the client system** – Each client uses exactly one AI manifest to complete its installation. The AI manifest is selected for a client according to a particular algorithm. See [“Selecting the AI Manifest” in *Installing Oracle Solaris 11 Systems*](#).
- **Select SC profiles** – The criteria keywords that are used for selecting SC profiles for a client are the same as those that are used for selecting an AI manifest. More than one SC profile can be selected for any particular client. See [“Selecting System Configuration Profiles” in *Installing Oracle Solaris 11 Systems*](#).

The Oracle Configuration Manager settings can also be configured in an SC profile. To opt out of using Oracle Configuration Manager for any clients, see [“Opting Out of Oracle Configuration Manager” in *Installing Oracle Solaris 11 Systems*](#).

- **Specify selection criteria** – The following keywords and values can be used with the `create-manifest`, `create-profile`, and `set-criteria` subcommands of the `installadm` command:

```
-c criteria=value|list|range
-c mac="aa:bb:cc:dd:ee:ff"
-c zonename="zone1 zone2"
-c mem="2048-unbounded"
```

See [“Selection Criteria” in *Installing Oracle Solaris 11 Systems*](#).

Provisioning a Client System

AI manifests contain instructions for how the client is to be installed. See [Chapter 10, “Provisioning the Client System,”](#) in *Installing Oracle Solaris 11 Systems*.

AI manifests can be created and used in the following ways:

- **Use the default AI manifest** – When you create an AI install service, you get a default AI manifest that specifies how to provision the clients. The AI manifest is an XML file that specifies where to install the operating system and what software packages to install. See [“Default AI Manifest”](#) in *Installing Oracle Solaris 11 Systems*.
- **Create a new, custom AI manifest** – You can create a custom XML AI manifest file. This method is best suited to an environment where fewer systems require custom provisioning. Most systems that are to be installed have identical or similar hardware and are provisioned identically.
- **Create a new AI manifest at client installation time** – You can write a script that dynamically creates an AI manifest for each client at installation time. Use this method to create a custom installation for each client, based on client characteristics that are discovered at installation time.

Configuring a Client System

You can configure a client by providing configuration profiles, by specifying the configuration in an SC profile, or by using scripts that are executed by an SMF service that runs once at the first boot of a system after the installation. These methods replace the JumpStart method that uses finish scripts and `sysidcfg`. For example, a client might be assigned one profile that provides only a static network configuration or naming services configuration. The same client, and other clients, might be assigned other profiles that set more broadly applicable property values. See [“Sample System Configuration Profile”](#) in *Installing Oracle Solaris 11 Systems*.

To create an SC profile, you can run the interactive configuration tool and save the output to a file, as shown in this example, where a valid profile is created in the `sc.xml` file from responses that are provided:

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

Or, you can create an SC profile by using the property specifications that are described in [“Specifying Configuration in a System Configuration Profile”](#) in *Installing Oracle Solaris 11 Systems*. For more information about using the `sysconfig` utility to unconfigure and reconfigure a system, see [“System Configuration Tools Changes”](#) on page 97.

EXAMPLE 2-1 Specifying a Client Configuration in an SC Profile

The following example shows how to specify a client configuration in an SC profile by using SMF properties:

EXAMPLE 2-1 Specifying a Client Configuration in an SC Profile *(Continued)*

```
# svccfg -s name-service/switch describe config
config application
Name service switch configuration data as described in nsswitch.conf(4).
config/value_authorization astring solaris.smf.value.name-service.switch
config/default astring files
Default configuration database entry.
config/host astring "files dns mdns"
config application
Name service switch configuration data as described in nsswitch.conf(4).
config/value_authorization astring solaris.smf.value.name-service.switch
config/default astring files
Default configuration database entry.
config/host astring "files dns mdns"
```

See “Specifying Configuration in a System Configuration Profile” in *Installing Oracle Solaris 11 Systems*.

Creating a Script That Runs at First Boot

To perform an installation or create a configuration that cannot be specified in an AI manifest or an SC profile, you can create a script that runs at first boot by using a once-run SMF service, as follows:

1. Create a first-boot script. See “Creating a Script To Run at First Boot” in *Installing Oracle Solaris 11 Systems*.
2. Create a manifest for an SMF service that runs one time at first boot and executes that script. See “Creating an SMF Manifest File” in *Installing Oracle Solaris 11 Systems*.
3. Create an IPS package that contains the service manifest and the script, then add the IPS package to the repository. See “Creating an IPS Package For the Script and Service” in *Installing Oracle Solaris 11 Systems*.
4. Install the first boot package on the AI client. See “Installing the First Boot Package on the AI Client” in *Installing Oracle Solaris 11 Systems*.

Installing Client Systems by Using AI

When you initially set up your install server, you created at least one install service for each client architecture and each version of Oracle Solaris that you plan to install. When you created each install service, you created customized installation instructions and system configuration instructions for different clients. The installation begins when you boot the client and follows this sequence:

1. The client is booted.
2. The client system contacts the DHCP server and retrieves the client IP address, the boot file, and the IP address of the installer, if required.

3. The client system loads the net image.
4. The client system completes its installation by using the appropriate AI manifest, as described in “Selecting the AI Manifest” in *Installing Oracle Solaris 11 Systems*.
5. The client system reboots, if the `auto_reboot` value is set in the AI manifest or the client, or if rebooted by the system administrator.
6. During a reboot, the client system is configured by using an SC profile, as described in “Selecting System Configuration Profiles” in *Installing Oracle Solaris 11 Systems*, or by the administrator's responses in the interactive system configuration tool.

When the `svc:/application/auto-installer` SMF service on that client reaches an online state, the installation is complete.

EXAMPLE 2-2 Setting Up an Installation Client

To set up an installation client on an install server, use the `installadm create-client` command. This command associates a particular client with a particular install service. The `installadm create-client` command requires the MAC address for the client and the name of the install service that the client is to use for installation.

For x86 clients, you can optionally specify boot properties, as shown in the following example:

```
# installadm create-client -n s11-x86 -e 0:e0:81:5d:bf:e0
No local DHCP configuration found. If not already configured, the
following should be added to the DHCP configuration:
Boot server IP : 10.80.239.5
Boot file : 01001122334455
```

Display the results of the `installadm create-client` command in the `/etc/netboot` directory:

```
# cd /etc/netboot
# ls -l
lrwxrwxrwx 13:23 0100E0815DBFE0 -> pxegrub.I86PC.Solaris-1
drwxr-xr-x 13:26 I86PC.Solaris-1
-rw-r--r-- 13:23 menu.lst.0100E0815DBFE0
-rwxr-xr-x 13:23 pxegrub.I86PC.Solaris-1
-rw-r--r-- 13:23 rm.0100E0815DBFE0

# cat menu.lst.0100E0815DBFE0
default=0
timeout=30
min_mem64=1000
title Oracle Solaris boot image
kernel$ /I86PC.Solaris-1/platform/i86pc/kernel/$ISADIR/unix
-B install_media=http://135.134.0.10:5555/export/aiserver/s11-ai-x86,
install_service=s11-ai-x86,install_svc_address=135.134.0.10:46501
module$ /I86PC.Solaris-1/platform/i86pc/$ISADIR/boot_archive
title Oracle Solaris Automated Install
kernel$ /I86PC.Solaris-1/platform/i86pc/kernel/$ISADIR/unix
-B install=true,install_media=http://135.134.0.10:5555/export/aiserver/s11-ai-x86,
install_service=osol-1003-ai-x86,install_svc_address=135.134.0.10:46501
```

EXAMPLE 2-2 Setting Up an Installation Client (Continued)

```
module$ /I86PC.Solaris-1/platform/i86pc/$ISADIR/boot_archive
```

To install a SPARC client, bring the system to the ok PROM prompt, then boot the system:

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

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

For details, see “Installing a SPARC Client” in *Installing Oracle Solaris 11 Systems* and “Installing an x86 Client” in *Installing Oracle Solaris 11 Systems*.

To install and configure zones by using AI, see Chapter 12, “Installing and Configuring Zones,” in *Installing Oracle Solaris 11 Systems*. For complete instructions on installing client systems by using AI, see Chapter 15, “Installing Client Systems,” in *Installing Oracle Solaris 11 Systems*.

Additional Installation Tasks

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

Configuring Date and Time Before and After an Installation

Oracle Solaris 11 keeps the Real Time Clock (RTC) in Coordinated Universal time (UTC) format. The behavior on x86 platforms is different in Oracle Solaris 11 than in Oracle Solaris 10 and Oracle Solaris 11 Express. The interactive installers enable you to configure the date and time during the installation. As part of that process, the RTC is updated with the time in UTC format. However, AI does *not* adjust the RTC date and time during an installation. To ensure that the time stamp of installed files are correct, configure the time in the BIOS in UTC format *before* beginning the installation. On x86 platforms, when using the `pkg update` command, the OS continues to keep time in RTC in the local time format. This method is used to avoid time inconsistencies between Oracle Solaris 11 BEs and BEs from previous releases.

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

▼ How to Switch From Local Time Format to UTC Format

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

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

▼ How to Switch From UTC Format to Local Time Format

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

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

```
# rtc -z timezone
```

For example:

```
# rtc -z US/Pacific
```
- 2 If the date/time requires an adjustment, use the `date` command.

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

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

- Switch from local time to UTC format in the OS that keeps RTC time in local time format.
 For example, if you are dual-booting Windows 7, set the registry key as follows:

```
[HKEY_LOCAL_MACHINESYSTEM\CurrentControlSet\Control\TimeZoneInformation] \
"RealTimeIsUniversal"=dword:00000001
```
- Switch from the UTC format to local time on a freshly installed Oracle Solaris 11 system.
- Enable the Network Time Protocol (NTP) in operating systems that assume that the RTC format is running in local time. In this case, the time is synchronized automatically.

Adding a Linux Entry to the GRUB Menu After an Installation

If you are setting up a boot environment in such a way that you install Linux on one partition first and Oracle Solaris on another partition afterwards, you will need to follow certain instructions to ensure that the GRUB menu information from the new installation does not

erase the GRUB menu information from a previous installation. For instructions, see “[How to Add a Linux Entry to the GRUB Menu After Installing Oracle Solaris](#)” in *Booting and Shutting Down Oracle Solaris on x86 Platforms*.

Note – Some Linux distributions now run on GRUB2, for example, Ubuntu and Mint Linux. You cannot boot GRUB2 partitions on the version of GRUB that is included in Oracle Solaris 11. In these instances, an alternate workaround is suggested.

Troubleshooting an Oracle Solaris 11 Installation

You might encounter some of the following issues during or after installing Oracle Solaris 11.

See also these references:

- “[What to Do If Your System Boots in Console Mode](#)” in *Installing Oracle Solaris 11 Systems*
- Chapter 16, “[Troubleshooting Automated Installations](#),” in *Installing Oracle Solaris 11 Systems*

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, there is no way to switch back to the GUI boot screen.

Troubleshooting Login and User Account Issues After an Installation

▼ x86: How to Troubleshoot Your Login

- 1 Boot the system in single-user mode.
 - a. When the boot sequence starts, and the GRUB menu is displayed, type **e** to edit the GRUB menu entries.

- b. Select the `kernel$` line, then type `e` to edit the entry.
 - c. Type `-s` at the end of the line to boot the system in single-user mode.
 - d. Press Return to go back to the previous screen, then type `b` to boot the system.
- 2 When prompted, log in as root with the root password.
After the system has booted, you can do any of the following:
 - Display the existing user accounts and roles.
`# cat /etc/user_attr`
 - Delete a user account.
`# userdel username`
 - Create a new user account.
`# useradd username`
 - a. Assign a password for the user name.
`# passwd username`
You will need to type the password twice.
 - b. Assign the root role to that user.
`# usermod -R root username`
- 3 To return to the installed system, type `exit`.

Managing Devices

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

This is a list of the information that is in this chapter:

- “Device Identity and Configuration Changes” on page 43
- “Preparing Disks for ZFS Storage Pools” on page 45
- “Swap and Dump Device Configuration Changes” on page 47

Device Identity and Configuration Changes

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

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

For more information, see [Chapter 6, “Dynamically Configuring Devices \(Tasks\),”](#) in *Oracle Solaris Administration: Devices and File Systems*.

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

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

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

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

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

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

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

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

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

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

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

```
% croinfo -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 `croinfo` example shows how to display a specific disk's physical location.

```
$ croinfo -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 `croinfo` 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.

Changes to Device Driver Customization

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

For more information, see “[How to Customize a Driver Configuration](#)” in *Oracle Solaris Administration: Devices and File Systems*.

Preparing Disks for ZFS Storage Pools

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

Review the following general pool device configuration recommendations:

- Create non-root pools by using whole disks, which are easier to manage than disk slices. For example, you can easily create a mirrored storage pool with 4 devices as follows:

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

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

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

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

- We recommend that you create non-root pools with whole disks but keep in mind that root pool disks need an SMI label from which to boot.

ZFS Root Pool Installation Improvements

Review the following installation improvements for root pools:

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

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

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

You can use AI manifest keyword syntax to create a mirrored root pool during an Oracle Solaris 11 automatic installation. For example:

```
<!DOCTYPE auto_install SYSTEM "file:///usr/share/install/ai.dtd.1">
<auto_install>
  <ai_instance name="default">
    <target>
      <disk whole_disk="true" in_zpool="rpool" in_vdev="root_mirror">
        <disk_name name="c8t0d0" name_type="ctd"/>
      </disk>
      <disk whole_disk="true" in_zpool="rpool" in_vdev="root_mirror">
        <disk_name name="c8t1d0" name_type="ctd"/>
      </disk>
      <zpool name="rpool" is_root="true">
        <vdev name="root_mirror" redundancy="mirror"/>
        <be name="solaris"/>
      </zpool>
    </target>
  </ai_instance>
</auto_install>
```

In the above example, the installer will automatically create slice 0 on each disk, `c8t0d0` and `c8t1d0`, which will be used to create a mirrored root pool. In addition, because an SMI label is created on disks for the root pool, the installer will also create a slice 2 and a slice 8 on x86 systems.

ZFS Root Pool Device Requirements

In general, root pool devices are relabeled and the root pool is created when the system is installed. Be aware of the following device requirements if you create a mirrored root pool after installation by using the `zpool attach` command or if you replace a disk in a root pool.

- If you attach a root pool disk with the `zpool attach` command after the system is installed to create a mirrored root pool and the intended disk contains an EFI label, you will need to relabel the disk with an SMI label.

In the Oracle Solaris 11 release, you can use the following shortcut command to force an SMI label to a disk with an EFI label that can be attached to an existing root pool. For 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 Oracle Solaris Administration: Devices and File Systems](#).

Boot blocks are applied automatically during a `zpool attach` operation of a root pool disk.

- If you use the `zpool replace` command to replace the root pool disk, you must first apply an SMI label and also apply the boot blocks. For more information, see [“How to Replace a Disk in a ZFS Root Pool” in Oracle Solaris Administration: ZFS File Systems](#).
- The disk that is intended for the root pool must be less than 2 TBs in size so that the Oracle Solaris OS can boot successfully.
- The pool must exist either on a disk slice or on disk slices that are mirrored. If you attempt to use an unsupported pool configuration during an `beadm` operation, you will see a message similar to the following:

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

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

For more information about creating ZFS root pools, see [“ZFS Storage Pool Creation Practices” on page 50](#).

Swap and Dump Device Configuration Changes

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

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

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

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

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

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

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

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

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

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

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

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

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

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


Managing Storage Features

This chapter describes storage management changes in Oracle Solaris 11.

This is a list of the information that is in this chapter:

- [“Comparison of Solaris Volume Manager Configurations to ZFS Configurations” on page 49](#)
- [“Recommended ZFS Storage Pool Practices” on page 50](#)
- [“COMSTAR Replaces iSCSI Target Daemon” on page 53](#)

Comparison of Solaris Volume Manager Configurations to ZFS Configurations

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

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

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

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

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

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


```
# zpool create tank mirror c1t0d0 c2t0d0 mirror c1t1d0 c2t1d0
```
- Solaris Volume Manager RAID-5 (distributed parity) is available as a ZFS RAID-Z (raidz1) configuration. For example:


```
# zpool create rzpool raidz1 c1t0d0 c2t0d0 c1t1d0 c2t1d0
```
- Solaris Volume Manager does not offer a RAID-6 but ZFS provides both RAIDZ-2 and RAIDZ-3 parity configurations, which means a RAIDZ-2 configuration can withstand the failure of two disks and a RAIDZ-3 configuration can withstand the failure of 3 disks. For example:


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

Recommended ZFS Storage Pool Practices

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

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

ZFS Storage Pool Creation Practices

- **Root pool** – Create pools with slices by using the s* identifier. Do not use the p* identifier. In general, a system's ZFS root pool is created when the system is installed. If you are creating a second root pool or re-creating a root pool, use syntax similar to the following:

```
# zpool create rpool c0t1d0s0
```

Or, create a mirrored root pool. For example:

```
# zpool create rpool mirror c0t1d0s0 c0t2d0s0
```

- Consider keeping root pool components separate from non-root pool data.
- The root pool must be created as a mirrored configuration or as a single-disk configuration. A RAID-Z or a striped configuration is not supported. You cannot add additional disks to create multiple mirrored top-level virtual devices by using the `zpool add` command, but you can expand a mirrored virtual device by using the `zpool attach` command.
- The root pool cannot have a separate log device.
- You can set pool properties during an AI installation by using the `pool_options` keyword syntax. The `gzip` compression algorithm is not supported on root pools.

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

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

Mirrored storage pools are also more flexible in that you can detach, attach, and replace existing devices in the pool.

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

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

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

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

ZFS Storage Pool Monitoring Practices

- Make sure that pool capacity is below 80% for best performance. Monitor pool and file system space to make sure that they are not full. Consider using ZFS quotas and reservations to make sure file system space does not exceed 80% pool capacity.
- Run `zpool scrub` on a regular basis to identify data integrity problems:
 - If you have consumer-quality drives, consider a weekly scrubbing schedule.
 - If you have datacenter-quality drives, consider a monthly scrubbing schedule.
 - You should also run a scrub prior to replacing devices to ensure that all devices are currently operational.
- Use `zpool status` on a weekly basis to monitor pool and pool device status. Also use `fmddump` or `fmddump -eV` to see if any device faults or errors have occurred.

ZFS Storage Pool Troubleshooting Practices

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

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

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

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

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

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

For a full description of troubleshooting pool problems, see [Chapter 11, “Oracle Solaris ZFS Troubleshooting and Pool Recovery,”](#) in *Oracle Solaris Administration: ZFS File Systems*.

COMSTAR Replaces iSCSI Target Daemon

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

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

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

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

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

For information about COMSTAR, see [Chapter 14, “Configuring Storage Devices With COMSTAR,”](#) in *Oracle Solaris Administration: Devices and File Systems*.

Managing File Systems

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

This is a list of the information that is in this chapter:

- [“Oracle Solaris 11 File System Changes” on page 55](#)
- [“Managing ZFS File System Changes” on page 56](#)
- [“Considering ZFS Backup Features” on page 62](#)
- [“Migrating File System Data to ZFS File Systems” on page 62](#)

Oracle Solaris 11 File System Changes

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

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

The general file system differences are as follows:

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

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

Root File System Requirements and Changes

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

- After a system is installed, the root of the Solaris file system is mounted, which means files and directories are accessible.
- All subdirectories of the root file system that are part of the Oracle Solaris OS, with the exception of `/var`, must be contained in the same file system as the root file system.
- During an Oracle Solaris 11 installation, a separate `/var` file system is created automatically for a global zone and a non-global zone.
- In addition, all Oracle Solaris OS components must reside in the root pool, with the exception of the swap and dump devices.
- A default swap device and dump device are automatically created as ZFS volumes in the root pool when a system is installed. You cannot use the same volume for both swap and dump devices. In addition, you cannot use swap files in a ZFS root environment. For more information, see [“Swap and Dump Device Configuration Changes” on page 47](#).

Managing ZFS File System Changes

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

- ZFS file system encryption – You can encrypt a ZFS file system when it is created. For more information, see [Chapter 9, “Managing Security.”](#)
- ZFS file system deduplication – For important information about determining whether your system environment can support ZFS data deduplication, see [“ZFS Data Deduplication Requirements” on page 61](#).
- ZFS file system sharing – Includes both NFS and SMB file system sharing changes. For more information, see [“ZFS File System Sharing Changes” on page 59](#).
- ZFS man page change – The `zfs.1m` manual page has been revised so that core ZFS file system features remain in the `zfs.1m` page, but delegated administration, encryption, and share syntax and examples are covered in the following pages:
 - [zfs_allow\(1M\)](#)
 - [zfs_encrypt\(1M\)](#)

- `zfs_share(1M)`

Displaying ZFS File System Information

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

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

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

    NAME            STATE      READ WRITE CKSUM
    rpool           ONLINE    0     0     0
    c2t0d0s0       ONLINE    0     0     0

errors: No known data errors
```

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

```
# zfs list -r rpool
NAME                                USED  AVAIL  REFER  MOUNTPOINT
NAME                                USED  AVAIL  REFER  MOUNTPOINT
rpool                               5.39G 67.5G 74.5K  /rpool
rpool/ROOT                          3.35G 67.5G   31K  legacy
rpool/ROOT/solaris                  3.35G 67.5G  3.06G  /
rpool/ROOT/solaris/var              283M 67.5G  214M  /var
rpool/dump                          1.01G 67.5G 1000M  -
rpool/export                       97.5K 67.5G   32K  /rpool/export
rpool/export/home                   65.5K 67.5G   32K  /rpool/export/home
rpool/export/home/admin             33.5K 67.5G   33.5K /rpool/export/home/admin
rpool/swap                          1.03G 67.5G  1.00G  -
```

For a description of the root pool components, see [“Reviewing the Initial ZFS BE After an Installation” on page 73](#).

Resolving ZFS File System Space Reporting Issues

The `zpool list` and `zfs list` commands are better than the previous `df` and `du` commands for determining your available pool and file system space. With the legacy commands, you cannot easily discern between pool and file system space, nor do the legacy commands account for space that is consumed by descendent file systems or snapshots.

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

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

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

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

Resolving ZFS Storage Pool Space Reporting Issues

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

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

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

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

```
# zpool create tank mirror c0t6d0 c0t7d0
# zpool list tank
NAME  SIZE  ALLOC  FREE   CAP  DEDUP  HEALTH  ALTROOT
tank 136G 95.5K 136G   0%  1.00x  ONLINE  -
# zfs list tank
NAME  USED  AVAIL  REFER  MOUNTPOINT
tank  72K  134G   21K  /tank
```

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

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

Making ZFS File Systems Available

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

- A ZFS file system is mounted automatically when it is created and then remounted automatically when the system is booted.
- You do not have to modify the `/etc/vfstab` file to mount a ZFS file system, unless you create a legacy mount for ZFS file system. Mounting a ZFS file system automatically is recommended over using a legacy mount.
- You do not have to modify the `/etc/dfs/dfstab` file to share file systems. For more information about sharing ZFS file systems, see [“ZFS File System Sharing Changes” on page 59](#).
- Similar to a UFS root, the swap device must have an entry in the `/etc/vfstab` file.
- File systems can be shared between Oracle Solaris 10 and Oracle Solaris 11 systems by using NFS sharing.
- File systems can be shared between Oracle Solaris 11 systems by using NFS or SMB sharing.
- ZFS storage pools can be exported from an Oracle Solaris 10 system and then imported on an Oracle Solaris 11 system.

ZFS File System Sharing Changes

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

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

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

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

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

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

The primary new sharing differences are as follows:

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

Legacy ZFS Sharing Syntax

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

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

For example, to share a ZFS file system:

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

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

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

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

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

The above file system shares are published immediately.

ZFS Sharing Migration/Transition Issues

Review the share transition issues in this section.

- **Upgrading your system** – ZFS shares will be incorrect if you boot back to an older BE due to property changes in this release. Non-ZFS shares are unaffected. If you plan to boot back to an older BE, save a copy of the existing share configuration prior to the `pkg update` operation to be able to restore the share configuration on the ZFS datasets.

- In the older BE, use the `sharemgr show -vp` command to list all shares and their configuration.
- Use the `zfs get sharenfs filesystem` command and the `zfs sharesmb filesystem` commands to get the values of the sharing properties.
- If you boot back to an older BE, reset the `sharenfs` and `sharesmb` properties to their original values.
- **Legacy unsharing behavior** – Using the `unshare -a` command or `unshareall` command unpublishes a share, but does not update the SMF shares repository. If you try to re-share the existing share, the shares repository is checked for conflicts, and an error is displayed.

ZFS Data Deduplication Requirements

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

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

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

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

```
# zdb -S tank
```

```
Simulated DDT histogram:
```

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

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

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

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

2. Make sure your system has enough memory to support dedup.

- Each in-core dedup table entry is approximately 320 bytes.
- Multiply the number of allocated blocks times 320. For example:

$$\text{in-core DDT size} = 2.63\text{M} \times 320 = 841.60\text{M}$$

3. Dedup performance is best when the deduplication table fits into memory. If the dedup table has to be written to disk, then performance will decrease. If you enable deduplication on your file systems without sufficient memory resources, system performance might degrade during file system related operations. For example, removing a large dedup-enabled file system without sufficient memory resources might impact system performance. .

Considering ZFS Backup Features

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

Migrating File System Data to ZFS File Systems

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

Recommended Data Migration Practices

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

Migrating Data With ZFS Shadow Migration

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

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

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

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

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

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

- Data from the file system to be migrated is copied over to the shadow file system. For step-by-step instructions, see [“Migrating ZFS File Systems” in Oracle Solaris Administration: ZFS File Systems](#).

Review the following considerations when migrating file systems:

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

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

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

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

If the original UFS file system data includes POSIX-draft ACLs, they are translated into to NFSv4 ACLs. See the [Chapter 8, “Using ACLs and Attributes to Protect Oracle Solaris ZFS Files,”](#) in *Oracle Solaris Administration: ZFS File Systems*.

Managing Software

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

This is a list of the information that is in this chapter:

- [“Oracle Solaris 11 Package Changes” on page 65](#)
- [“Oracle Solaris 10 SVR4 and IPS Package Comparison” on page 66](#)
- [“Displaying Information About Software Packages” on page 68](#)
- [“Updating the Software on Your Oracle Solaris 11 System” on page 70](#)
- [“Managing Boot Environments” on page 72](#)

Oracle Solaris 11 Package Changes

IPS is a framework that provides the capability for software lifecycle management, which includes installation, upgrade, and the removal of packages. IPS utilizes 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 pkg(5) packages are stored in IPS package repositories that are populated by IPS publishers. See pkg(5).

The following IPS components are described in this chapter:

- **IPS command-line utilities** – IPS includes a suite of pkg commands that enable you to list, search, install, update, and remove software packages. See pkg(1). IPS commands also enable you to manage package publishers and copy or create package repositories. See [“Updating the Software on Your Oracle Solaris 11 System” on page 70](#).
- **IPS GUI tools (Package Manager and Update Manager)** – The Package Manager and Update Manager GUIs are used to update and manage software packages from the desktop.

- **IPS publishers and repositories** – A *publisher* is a forward domain name that identifies a person, group of persons, or an organization that publishes one or more packages. A *repository* is like a package archive, where clients can publish and retrieve packages. Several Oracle package repositories are available. See *Copying and Creating Oracle Solaris 11 Package Repositories*.

If you have a system with zones that does not have direct access to a public IPS repository, see “Using `https_proxy` and `http_proxy` on a System That Has Installed Zones” in *Oracle Solaris Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.

- **Boot environment management** – The `beadm` utility is used to create and manage boot environments, which are bootable instances of the Oracle Solaris operating system image, along with any other software packages that are installed in that image. See “Tools for Managing Boot Environments” on page 72.

For SVR4 package to IPS package conversion information, see the *IPS Developer's Guide*.

Oracle Solaris 10 SVR4 and IPS Package Comparison

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

- The `SUNW` prefix for package names is no longer used. With the introduction of IPS, all software packages are renamed. A set of mappings has been added to the former SVR4 package database for compatibility. The mappings ensure package dependencies are met for administrators who want to install a legacy SVR4 package.
- Certain SVR4 package commands, such as `pkgadd`, are retained, but the primary package installation and update interface is now the `pkg(5)` set of commands. If you previously used the `pkgadd` command to install a particular package, you can check whether that package is available as an IPS package. The IPS package name will most likely be different.

Locate an SVR4 package as follows:

```
$ pkg info -r svr4pkgname
```

Look for information in the `Renamed to` column. If you find the package in an IPS repository, change the `pkgadd svr4pkgname` command, as follows:

```
$ pkg install ipspkgname
```

See Chapter 12, “Managing Software Packages (Tasks),” in *Oracle Solaris Administration: Common Tasks*.

- 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.
- Package names are now FMRIs, similar to SMF service names. Package names are also hierarchical instead of abbreviated. For example, the core system library package in Oracle Solaris 10 is `SUNWcsL`. Now, the same package name is `pkg:/system/library`.

Note – Due to organizational restructuring of the files that are delivered with each package, there is no one-to-one mapping of Oracle Solaris 10 package names to Oracle Solaris 11 package names.

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

Remove the man pages as follows:

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

Remove the header files as follows:

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

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

- SVR4 packaging and patch tools are still supported in Oracle Solaris 10 Containers. These Oracle Solaris 10 branded, non-global zones run on Oracle Solaris 11, utilizing the zones and branded zones technology. See [“Oracle Solaris 11 Zone Features” on page 118](#).

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>
<code>pkginfo, pkgchk -l</code>	<code>pkg info, pkg list, pkg contents, pkg search</code>
<code>pkgchk</code>	<code>pkg verify, pkg fix, pkg revert</code>

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.

In comparison, IPS provides package installation groups that represent a larger server environment, desktop environments, an AI client environment, and so on.

Display package group information as follows:

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

Display the contents of these package groups:

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

Determine which package group is currently installed on your system:

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

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

If you want to install most packages, similar to installing the Solaris 10 SUNWCall package cluster, consider installing the `group/system/solaris-large-server` package group.

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.

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

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

List packages that are not installed, but are available to be installed.

```
$ pkg list -a amp
NAME (PUBLISHER)                VERSION                                IFO
amp                            0.5.11-0.133                          --r
group/feature/amp              0.5.11-0.175.0.0.0.0.2566             ---
web/amp                         0.5.11-0.174.0.0.0.0.0                --r
```

Display more information about a package. If you already know the package is not installed in the image, use the `-r` option to query the package repository, as follows:

```

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

Name: group/feature/amp
Summary: AMP (Apache, MySQL, PHP) Deployment Kit for Oracle Solaris
Description: Provides a set of components for deployment of an AMP (Apache,
MySQL, PHP) stack on Oracle Solaris
Category: Meta Packages/Group Packages ...
Web Services/Application and Web Servers (...)
State: Not installed
Publisher: solaris
Version: 0.5.11
Build Release: 5.11
Branch: 0.175.0.0.0.2566
Packaging Date: Tue Sep 27 13:06:15 2011
Size: 5.45 kB
FMRI: pkg://solaris/group/feature/amp@0.5.11,5.11-0.175.
0.0.0.2566:20110927T130615Z

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
Build Release: 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@0.5.11,5.11-0.174.
0.0.0.0:20110921T191502Z

```

List all of the package groups that are installed or that are available for installation:

```

% pkg list -a 'group/*'
NAME (PUBLISHER)                VERSION                IFO
group/feature/amp (solaris)     0.5.11-0.175.1.0.0.3.2590 ---
group/feature/developer-gnu (solaris) 0.5.11-0.175.1.0.0.3.2590 i--
group/feature/multi-user-desktop (solaris) 0.5.11-0.175.1.0.0.3.2590 ---
group/feature/storage-avs (solaris)    0.5.11-0.175.1.0.0.3.0    i--
group/feature/storage-nas (solaris)    0.5.11-0.175.1.0.0.3.0    i--
group/feature/storage-server (solaris) 0.5.11-0.175.1.0.0.3.0    i--
group/feature/trusted-desktop (solaris) 0.5.11-0.175.1.0.0.3.2590 i--
group/system/solaris-auto-install (solaris) 0.5.11-0.175.1.0.0.3.2590 ---
group/system/solaris-nas (solaris)     0.5.11-0.175.1.0.0.3.2590 ---
group/system/solaris-large-server (solaris) 0.5.11-0.175.1.0.0.3.2590 ---
group/system/solaris-small-server (solaris) 0.5.11-0.175.1.0.0.3.2590 i--

```

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

```
$ pkg search /usr/bin/emacs
INDEX      ACTION VALUE      PACKAGE
path      file   usr/bin/emacs  pkg:/editor/gnu-emacs@23.1-0.175.0.0.0.525
$ pkg search file::emacs
INDEX      ACTION VALUE      PACKAGE
basename  file   usr/bin/emacs  pkg:/editor/gnu-emacs@23.1-0.175.0.0.0.525
basename  file   usr/share/info/emacs  pkg:/editor/gnu-emacs@23.1-0.175.0.0.0.525
```

Updating the Software on Your Oracle Solaris 11 System

With IPS, you can update all of the packages on your system that have available updates, or you can update individual packages that are not constrained by the system. If a package is constrained, an appropriate message indicating why it has been constrained is provided. Package constraints generally represent a dependency or a versioning issue. For all package update operations, either a new BE is created or a backup BE is created before applying the software updates to the running BE.

The following options are available:

- **Adding software packages after an installation** – The Live Media contains a set of software that is appropriate for a desktop or laptop. The text installation media contains a smaller set of software that is more appropriate for a general-purpose server system. The text installer does not install the GNOME desktop. To add packages, including the Oracle Solaris Desktop (GNOME 2.30), after a text installation, see [“Adding Software After Text Installation”](#) in *Installing Oracle Solaris 11 Systems*.
- **Updating all of the packages on your installed system** – To update all of the packages on your system that have available updates, use the `pkg update` command, as follows:

```
# pkg update
```

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

See an example of using this command to update a BE in [“Managing Boot Environments”](#) on page 72.

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

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

You can also perform updates by using Package Manager and Update Manager. See [Chapter 2, “IPS Graphical User Interfaces,”](#) in *Adding and Updating Oracle Solaris 11 Software Packages*.

To upgrade a system that is running Oracle Solaris 11 Express or the Oracle Solaris Early Adopter release, see [Oracle Solaris 11 Release Notes](#).

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

Install an individual package as follows:

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

Update an individual package as follows:

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

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

Installing Maintenance Updates on an Oracle Solaris 11 System

Oracle customers with an active Oracle support plan have access to the support package repository so that you can routinely update your Oracle Solaris 11 systems. Updates to the support repository are called Support Repository Updates (SRUs) and occur on a regular basis. See [“How to Configure the Oracle Solaris support Repository”](#) on page 72.

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

The following summary provides information about selecting the update method that work best for your environment.

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

▼ How to Configure the Oracle Solaris support Repository

Use the following steps to configure the support repository, if you want to update your Oracle Solaris 11 FCS release. If you want to upgrade your Oracle Solaris 11 Express or Early Adopter system from the support repository, see the [Oracle Solaris 11 Release Notes](#).

1 Log into the following site.

<http://pkg-register.oracle.com/>

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

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

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

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

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

4 Set the publisher to the support repository.

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

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

```
# pkg update
```

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

Managing Boot Environments

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

Tools for Managing Boot Environments

In Oracle Solaris 11, the `beadm` utility replaces the `lu` set of commands for managing ZFS BEs. In addition, the `pkg update` command updates your existing BE and creates a clone BE, if necessary.

TABLE 6-2 Comparison of Boot Environment Command Syntax

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

See [Creating and Administering Oracle Solaris 11 Boot Environments](#) and [beadm\(1M\)](#).

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

Use the `beadm` command to create, rename, mount, unmount, activate, or destroy BEs. You can use the Package Manager to activate, rename, and delete BEs.

Reviewing the Initial ZFS BE After an Installation

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

```
# zfs list -r rpool
NAME                                USED  AVAIL  REFER  MOUNTPOINT
rpool                               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  -
```

- `rpool` – Is 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/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. If you update a ZFS BE by using `pkg update`, a new BE is created and automatically activated. If the updates to the existing BE are minimal, a backup BE is created before the updates are applied. The `pkg update` command displays whether a backup BE or a new BE is created.

1 Display your existing BE information.

```
# beadm list
BE      Active Mountpoint Space  Policy Created
-----
solaris NR    /                12.24G static 2011-10-04 09:42
```

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

2 Update your BE.

```
# pkg update
          Packages to remove: 117
          Packages to install: 186
          Packages to update: 315
          Create boot environment: Yes
DOWNLOAD
Completed          PKGS      FILES    XFER (MB)
                   618/618  29855/29855  600.7/600.7
.
.
.
```

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

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

```
# init 6
.
.
.
# beadm list
BE      Active Mountpoint Space  Policy Created
-----
solaris NR    /                12.24G static 2011-10-04 09:42
solaris-1 -    -                6.08G  static 2011-10-11 10:42
```

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

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

Managing Network Configuration

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

This is a list of the information that is in this chapter:

- [“How the Network Is Configured in Oracle Solaris 11” on page 75](#)
- [“Managing Network Configuration When in Manual Mode” on page 78](#)
- [“Managing Network Configuration When in Automatic Mode” on page 85](#)
- [“Commands That Are Used for Network Configuration \(Quick Reference\)” on page 88](#)

How the Network Is Configured in Oracle Solaris 11

Oracle Solaris 11 uses profile-based network configuration, which is comprised of two network configuration modes: manual (fixed) and automatic (reactive). Depending on which network configuration mode you choose during an installation, either the `DefaultFixed` network configuration profile (NCP) or the `Automatic` NCP is activated on the system. If the `DefaultFixed` NCP is active, the network is manually configured by using the `dladm` and `ipadm` commands. If the `Automatic` NCP or a user-defined NCP that you previously created is active, the `netcfg` and `netadm` commands (formerly `nwamcfg` and `nwamadm`) are used to create and manage network configuration.

In Oracle Solaris 11 Express, automatic network configuration was introduced in the Network Auto-Magic (NWAM) feature. For information about how this feature has changed, see [“Network Configuration Changes” on page 137](#).

Note the following additional information about automatic network configuration in Oracle Solaris 11:

- There are two main profile types: the network configuration profile (NCP) and the Location profile. An NCP specifies the configuration of network links and interfaces, for example IP addresses. A Location profile manages system-wide network configuration, for example naming services and IPfilter settings. At least one NCP and one Location profile must be active on the system at all times when you are using automatic network configuration.
- The `Automatic` NCP is a system-defined profile that cannot be modified or deleted. System-defined Locations include the `Automatic` and `NoNet` Locations. Unlike system-defined NCPS, system-defined Locations can be modified after the profile has been activated on a system for the first time.

For more information, see “[Overview of NWAM Configuration](#)” in *Oracle Solaris Administration: Network Interfaces and Network Virtualization* and “[Managing Network Configuration When in Automatic Mode](#)” on page 85.

How the Network Is Configured During an Installation

During a *fresh installation*, the network is configured as follows:

- For a GUI installation, the `Automatic` NCP is activated, and the network is automatically configured, based on current network conditions.
- For a text installation, you must choose `Automatic`, `Manual`, or `None`.
 - If you choose `Automatic`, the `Automatic` NCP is activated, and the network is automatically configured upon reboot.
 - If you choose `Manual`, the `DefaultFixed` NCP is activated, and you are presented with a series of installation screens that enable you to manually configure your network settings.
 - If you choose `None`, the `DefaultFixed` NCP is activated, but you do not provide network parameters during the installation. Thus, after a reboot, no network interface is plumbed or configured. Only the loopback IPv4 and IPv6 interfaces (`lo0`) are activated. You must manually configure your network by using `dladm` and `ipadm` in this case. See “[Managing Network Configuration When in Manual Mode](#)” on page 78.
- For an installation with AI, the network is configured according to the profile that you set up before the installation. By default, the interactive `sysconfig` tool runs during the installation, enabling you to set network parameters for the system. See “[Installing Oracle Solaris 11 by Using AI](#)” on page 33.

For information about how the network is configured after an upgrade from Oracle Solaris 11 Express, see “[Network Configuration Changes](#)” on page 137.

EXAMPLE 7-1 Verifying the Active NCP on a System

The following example shows the output of the `netadm list` command when the Automatic NCP is active on a system:

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

Note that in this example, a user-defined Location profile called `myloc` is also online. When using automatic network configuration, at least one NCP and one location must be active on the system at all times.

If the `DefaultFixed` NCP is active, and you run the `netadm list` command, the following output is displayed:

```
# netadm list
netadm: DefaultFixed NCP is enabled; automatic network management is not available.
'netadm list' is only supported when automatic network management is active.
```

When the `DefaultFixed` NCP is active, the network must be manually configured by using the `dladm` and `ipadm` commands.

EXAMPLE 7-2 Switching the Default NCP

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

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

Switch to the Automatic NCP as follows:

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

Note that the process of switching network configuration modes can sometimes take a few minutes. During this time, messages about various network services might be displayed on the screen. These messages can be ignored.

EXAMPLE 7-2 Switching the Default NCP (Continued)

To create user-defined NCPs when in automatic network configuration mode, see [“Managing Network Configuration When in Automatic Mode” on page 85](#).

Managing Network Configuration When in Manual Mode

If you are manually configuring your network, keep the following key points in mind:

- To manually configure the network by using the `dladm` and `ipadm` commands, the `DefaultFixed` NCP must be active (online) after the installation or upgrade. Use the `netadm` command to verify which NCP is currently active on your system. See [Example 7-1](#).

If the `DefaultFixed` NCP is *not* active, you will need to enable the profile before you can manually configure the network. See [Example 7-2](#).

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

Note – The `/etc/nsswitch.conf` file is still referenced in this release, but you do not directly edit the file to make configuration changes. See [“Configuring Naming Services When in Manual Mode” on page 81](#).

For more information about setting a system's `nodename` (also sometimes referred to as `hostname`) in this release, see [“System Configuration Changes and Migration of System Configuration to SMF” on page 93](#).

- During an installation, the system undergoes a one time upgrade to convert any existing `/etc` network configuration files to their respective `ipadm` and `dladm` configurations. The `dladm` command is used to configure datalinks. The `ipadm` command is used to configure link and IP interfaces. The `ipadm` command provides almost equivalent functionality to the `ifconfig` command. The `ipadm` command also replaces the `ndd` command. To compare `ifconfig` command options with `ipadm` subcommands, see [“Comparison Tables: ipadm Command and Other Networking Commands” in Oracle Solaris Administration: Network Interfaces and Network Virtualization](#).
- Network virtualization features are also configured and managed by using the `dladm` and `ipadm` commands. Objects that are in the link layer (Layer 2) of the network stack, for example, virtual local area networks (VLANs), tunnels, link aggregations, and the newer

virtual NICs (VNICs), are configured by using the `dladm` command. Interfaces that are on the IP layer (Layer 3) are configured by using the `ipadm` command. See [Chapter 19, “Configuring Virtual Networks \(Tasks\)”](#) in *Oracle Solaris Administration: Network Interfaces and Network Virtualization* and [Chapter 6, “Configuring IP Tunnels,”](#) in *Oracle Solaris Administration: IP Services*.

For additional information about setting network properties, see [Chapter 4, “Internet Protocol Suite Tunable Parameters,”](#) in *Oracle Solaris Tunable Parameters Reference Manual*.

Configuring Network Interfaces When in Manual Mode

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

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

The `-T` option can be used to specify three address types: `static`, `dhcp`, and `addrconf` (for auto-configured IPv6 addresses). In this example, the system is configured with a static IPv4 address. Use the same syntax to specify a static IPv6 address.

Configure an interface with DHCP as follows:

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

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/addr
```

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

```
# ipadm delete-ip net0
# ipadm create-ip net0
# ipadm create-addr -T addrconf net0/new-addr
```

For more information about the commands that are used to manually configure the network, see [“Commands That Are Used for Network Configuration \(Quick Reference\)”](#) on page 88.

See also [Chapter 9, “Configuring an IP Interface,”](#) in *Oracle Solaris Administration: Network Interfaces and Network Virtualization* and `ipadm(1M)`.

Displaying and Configuring Link Interfaces When in Manual Mode

When you perform a fresh installation, all datalinks are automatically assigned generic names by using a `net0`, `net1`, and `netN` naming convention, depending on the total number of network devices on a system. After installing, you can use different datalink names. See “[How to Rename a Datalink](#)” in *Oracle Solaris Administration: Network Interfaces and Network Virtualization*.

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

Display information about the datalinks on a system as follows:

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

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

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

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

Rename a datalink as follows:

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

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

Then, change the link's current name.

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


The *old-linkname* refers to the current name of the datalink. By default, the link name is hardware-based, such as `bge0`. The *new-linkname* refers to any name that you want to assign to the datalink. For more information, see [Chapter 8, “Datalink Configuration and Administration,”](#) in *Oracle Solaris Administration: Network Interfaces and Network Virtualization*. See also [Chapter 7, “Using Datalink and Interface Configuration Commands on Profiles,”](#) in *Oracle Solaris Administration: Network Interfaces and Network Virtualization*.

Configuring Naming Services When in Manual Mode

The SMF repository is the primary repository for all naming services configuration. The previous behavior of modifying a configuration file to configure naming services no longer works. These services, for example, `svc:/system/name-service/switch`, `svc:/network/dns/client`, and `svc:/system/name-service/cache` must be enabled or refreshed for any changes to take affect.

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

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

TABLE 7-1 SMF Service to Legacy File Mapping

SMF Service	Files	Description
<code>svc:/system/name-service/switch:default</code>	<code>/etc/nsswitch.conf</code>	Naming service switch configuration (Used by the <code>nscd</code> command)
<code>svc:/system/name-service/cache:default</code>	<code>/etc/nscd.conf</code>	Naming service cache (<code>nscd</code>)
<code>svc:/network/dns/client:default</code>	<code>/etc/resolv.conf</code>	DNS naming service
<code>svc:/network/nis/domain:default</code>	<code>/etc/defaultdomain</code> <code>/var/yp/binding/\$DOMAIN/*</code>	Shared NIS domain configuration (Used by all NIS services). Also historical shared use by LDAP naming services Note – Must be enabled when using <code>nis/client</code> or <code>ldap/client</code>
<code>svc:/network/nis/client:default</code>	N/A	NIS client naming service (<code>yppbind</code> and related files)

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

SMF Service	Files	Description
svc:/network/ldap/client:default	/var/ldap/*	LDAP client naming service (ldap_cachemgr and related files)
svc:/network/nis/server:default	N/A	NIS server naming service (ypserv)
svc:/network/nis/passwd:default	N/A	NIS server passwd service (rpc.yppasswd)
svc:/network/nis/xfr:default	N/A	NIS server xfr naming service (ypxfrd)
svc:/network/nis/update:default	N/A	NIS server update naming service (rpc.yppupdated)
svc:/system/name-service/upgrade:default	N/A	Naming legacy file to SMF upgrade service

EXAMPLE 7-3 Configuring Naming Services by Using SMF

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

```
# svccfg
svc:> select dns/client
svc:/network/dns/client> setprop config/search = astring: \
("us.company.com" "eu.company.com" "companya.com" "companyb.com" "company.com" )
svc:/network/dns/client> setprop config/nameserver = net_address: \
( 138.2.202.15 138.2.202.25 )
svc:/network/dns/client> select dns/client:default
svc:/network/dns/client:default> refresh
svc:/network/dns/client:default> validate
svc:/network/dns/client:default> select name-service/switch
svc:/system/name-service/switch> setprop config/host = astring: "files dns"
svc:/system/name-service/switch> select system/name-service/switch:default
svc:/system/name-service/switch:default> refresh
svc:/system/name-service/switch:default> validate
svc:/system/name-service/switch:default>
# svcadm enable dns/client
# svcadm refresh name-service/switch
# grep host /etc/nsswitch.conf
hosts: files dns
# cat /etc/resolv.conf
#
# opyright (c) 2011, Oracle and/or its affiliates. All rights reserved.
#
#
# _AUTOGENERATED_FROM_SMF_V1_
#
# WARNING: THIS FILE GENERATED FROM SMF DATA.
# DO NOT EDIT THIS FILE. EDITS WILL BE LOST.
```

EXAMPLE 7-3 Configuring Naming Services by Using SMF (Continued)

```
# See resolv.conf(4) for details.

search    us.company.com eu.company.com companya.com companyb.com company.com
nameserver 138.2.202.15
nameserver 138.2.202.25
.
.
.
```

EXAMPLE 7-4 Setting Multiple NIS Servers by Using SMF

The following example shows how to set multiple NIS servers.

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

Note that there is space between 1.2.3.4 and 5.6.7.8.

EXAMPLE 7-5 Setting Multiple DNS Options by Using SMF

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

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

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

resolv.conf Error-Checking Capabilities

Prior to the naming services to SMF migration, errors in the `resolv.conf` file configuration were processed silently and went undetected without producing any warnings. As a result, the `resolv.conf` file did not behave according to how it was configured. In Oracle Solaris 11, some basic error checking is performed through the use of SMF templates so that error conditions are properly reported. Note that other SMF services also all have some rudimentary error-checking capabilities. However, the `resolv.conf` error reporting is the most prominent due to the absence of error reporting in `libresolv2`. See [resolv.conf\(4\)](#).

Temporarily Resetting SMF Naming Services

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

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

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

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

Importing Network Configuration

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

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

In this example, the `nscfg` command reads information in the `/etc/resolv.conf` file, converts it, then stores the information in the `svc:/network/dns/client` SMF service.

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

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

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

For more information, see [nscfg\(1M\)](#).

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

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

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

```
# nscfg import -f svc:/system/name-service/switch:default
```
- 4 **Refresh the name service switch SMF service.**

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

Configuring LDAP When in Manual Mode

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

Managing Network Configuration When in Automatic Mode

Automatic network configuration is made up of a collection of properties that determine how the network is configured, depending on current network conditions. Configuration information is associated with several different types of profiles that are activated and deactivated by the system or by you. See “[Overview of NWAM Configuration](#)” in *Oracle Solaris Administration: Network Interfaces and Network Virtualization*.

An NCP configures network links and interfaces. A Location configures system-wide network settings, for example naming services and IPfilter settings. NCPs are made up of individual configuration objects that are called Network Configuration Units (NCUs). Each NCU represents a physical link or an interface that includes the properties that define the configuration for that particular link or interface, as shown in the output of the `netcfg list` command in the following example:

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

User-defined NCPs and Locations are created by using the `netcfg` command, either in command-line mode or interactively. For example, you would create a new NCP called `myncp` by using the `netcfg` command interactively as follows:

```
$ netcfg
netcfg> create ncp myncp
```

You can configure network settings for an NCP when you create it, or you can configure or reconfigure network settings for an existing NCP by using the `netcfg select` command either in command-line mode or interactively, as shown in the following example.

EXAMPLE 7-6 Configuring a Static IP Address for an existing NCP

In the following example, the `netcfg` command is used interactively to select the newly created NCP, `myncp`, then a static IP address is configured for this NCP.

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

EXAMPLE 7-7 Enabling an NCP

In the following example, a user-defined NCP named `myncp` is enabled by using the `netadm` command.

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

Configuring Naming Services When in Automatic Mode

System-wide network configuration, for example, naming services configuration, is managed in the Location profile. Properties are configured by using the `netcfg` command. The Automatic Location profile allows the system to automatically configure naming services. The Automatic Location profile is used to configure DNS through DHCP *only*. Note that the Automatic Location is different than the Automatic NCP, which configures link and IP interfaces. Before configuring naming service properties in a Location, you need to update the file that is to be referenced by the `nameservices-config-file` property of the specified Location. This file can be stored anywhere on the system. However, do not use the `/etc/nsswitch.conf` file name, as this file is overwritten.

See [Chapter 3, “NWAM Configuration and Administration \(Overview\),”](#) in *Oracle Solaris Administration: Network Interfaces and Network Virtualization*.

For example, you can create a new Location profile and then configure NIS as follows:

```

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

```

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

```

$ netcfg> select loc origloc
netcfg:loc:origloc> set nameservices=dns,nis
netcfg:loc:origloc> set nis-nameservice-configsrc=manual
netcfg:loc:origloc> set nis-nameservice-servers="1.2.3.38.1.3.3.36"
netcfg:loc:origloc> set default-domain="org.company.com"
netcfg:loc:origloc> set nameservices-config-file="/etc/nsswitch.dns"
netcfg:loc:origloc> end
Committed changes
netcfg> exit

```

Configuring LDAP When in Automatic Mode

Automatic network configuration mode provides limited support for LDAP. Only LDAP anonymous mode works when in automatic mode. If you want to use an LDAP proxy or LDAP self modes and some form of security credentials you must first enable the `DefaultFixed` profile and manually configure your network. For instructions, see [Chapter 12, “Setting Up LDAP Clients \(Tasks\)”](#), in *Oracle Solaris Administration: Naming and Directory Services*.

Managing Automatic Network Configuration From the Desktop

You can manage automatic network configuration from the desktop by using the NWAM GUI. The tool is similar to using the `netcfg` and `netadm` commands. You can connect to wired or wireless networks, configure a new wired or wireless connection, create Location profiles, and activate or deactivate profiles.

Note that IP-related information is configured in the Network Profile, under Connections. If the Network Profile information is not displayed in the upper right-hand corner of the desktop,

start the tool from the main menu by choosing System → Administration → Network Manager. See Chapter 6, “About the NWAM Graphical User Interface,” in *Oracle Solaris Administration: Network Interfaces and Network Virtualization* or the online help.

Commands That Are Used for Network Configuration (Quick Reference)

The following table is a quick reference for locating which commands to use for various network configuration tasks, based on network configuration mode that you are using.

TABLE 7-2 Commands That Are Used to Configure the Network

Configuration Task	Commands to Use When in Automatic Mode	Commands to Use When in Manual Mode
Switch network configuration modes (enable or disable an NCP or a Location profile).	<p>Enable automatic configuration: <code>netadm enable -p ncp Automatic</code></p> <p>Enable a user-defined NCP: <code>netadm enable -p ncp ncp-name</code></p> <p>Enable a Location profile (for setting system-wide network configuration): <code>netadm enable -p loc loc-name</code></p>	<p>Enable manual configuration: <code>netadm enable -p ncp DefaultFixed</code></p>
List the active network profile on the system.	<code>netadm list</code>	<p>Running <code>netadm list</code> results in an error message that says automatic network management is not available and that <code>netadm list</code> is only supported when automatic network management is active.</p> <p>Or, use the <code>svccprop -p netcfg/active_ncp physical:default</code> command to determine the current configuration mode.</p>
Configure link properties.	<code>netcfg "create ncp ncp-name; create ncu phys ncu-name; set property=value"</code>	<code>dladm set-linkprop -p property=value link</code>
Configure IP interfaces.	<code>netcfg "create ncp ncp-name; create ncu ip ncu-name; set property=value"</code>	<code>ipadm create-ip interface</code>

TABLE 7-2 Commands That Are Used to Configure the Network (Continued)

Configuration Task	Commands to Use When in Automatic Mode	Commands to Use When in Manual Mode
Configure IP addresses.	Static IP: netcfg "select ncp <i>ncp-name</i> ; select ncu ip <i>ncu-name</i> ; set ipv4-addrsrc=static; set ipv4-addr = 1.1.1.1/24" DHCP: netcfg "create ncp <i>ncp-name</i> ; create ncu ip <i>ncu-name</i> ; set ipv4-addrsrc=dhcp"	IPv4 or IPv6 static address: ipadm create-addr -T static -a <i>IP-address address-object</i> IPv4 DHCP address: ipadm create-addr -T dhcp <i>address-object</i> Auto-generated IPv6 address based on a system's MAC address: ipadm create-addr -T addrconf <i>address-object</i>
Modify existing network configuration.	Link properties: netcfg "select ncp <i>ncp-name</i> ; select ncu phys <i>ncu-name</i> ; set <i>property=value</i> " IP interface: netcfg "select ncp <i>ncp-name</i> ; select ncu ip <i>ncu-name</i> ; set <i>property=value</i> "	dladm set-linkprop -p <i>property datalink</i> ipadm set-prop [-t] -p prop= <i>value</i> [, ...] <i>protocol</i> ipadm set-addrprop [-t] -p prop= <i>value</i> [, ...] <i>addrobj</i> ipadm set-ifprop -p <i>property=value interface</i> ipadm set-prop -p <i>property=value protocol</i> ipadm set-addrprop -p <i>value addrobj</i>
Configure or modify naming services (NIS and DNS).	DNS from DHCP: netcfg "create loc <i>loc-name</i> ; set dns-nameservice-configsrc=dhcp" Manual DNS: netcfg "create loc <i>loc-name</i> ; set dns-nameservice-configsrc=static; set dns-nameservice-servers=1.1.1.1; set dns-nameservice-search=foo.com" For an existing Location, use netcfg "select..."	Set parameters for naming services: svccfg and svcadm
Configure LDAP.	Only LDAP anonymous mode works when in automatic mode. To use an LDAP proxy or LDAP self-modes, enable the DefaultFixed NCP.	ldapclient or SMF commands to select LDAP.

TABLE 7-2 Commands That Are Used to Configure the Network (Continued)

Configuration Task	Commands to Use When in Automatic Mode	Commands to Use When in Manual Mode
Configure default route.	netcfg "select ncp <i>ncp-name</i> ; select ncu ip ncu-name; set ipv4-default-route=1.1.1.1"	Set default route: route -p add default <i>routerIP-address</i> Set any static route: route -p add -net <i>nIP-address</i> -gateway <i>gIP-address</i>
Configure a system's host name (also sometimes referred to as a system's nodename).	svccfg -s sets the config/nodename property of the svc:system/identity:node SMF service to the desired name. Refresh and restart the service for the changes to take affect.	When the Automatic NCP is enabled on a system, the SMF service property is only set if the DHCP server does not provide a value for the nodename/hostname option (DHCP standard option code 12). See nodename(4)
Import naming service configuration.	Configured in the Locations profile.	/usr/sbin/nscfg import -f <i>FMRI</i> nscfg exports existing legacy files into the SMF repository.
Unconfigure and reconfigure a system (including all network configuration).	Unconfigure an Oracle Solaris instance: sysconfig unconfigure system Reconfigure an Oracle Solaris instance: sysconfig configure system	

Managing System Configuration

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

This is a list of the information that is in this chapter:

- “Comparing Oracle Solaris 10 and Oracle Solaris 11 System Configuration Tools” on page 91
- “System Configuration Changes and Migration of System Configuration to SMF” on page 93
- “System Console, Terminal Services, and Power Management Changes” on page 96
- “System Configuration Tools Changes” on page 97
- “System Registration Changes” on page 98
- “System Boot, Recovery, and Platform Changes” on page 98
- “Printer Configuration and Management Changes” on page 103
- “Internationalization and Localization Changes” on page 105

Comparing Oracle Solaris 10 and Oracle Solaris 11 System Configuration Tools

TABLE 8-1 Comparison of Oracle Solaris 10 Configuration Tools and Oracle Solaris 11 Configuration Tools

System Configuration Feature, Tool, or Function	Oracle Solaris 10	Oracle Solaris 11
System, network, and naming service configuration	Stored in various files within/etc	Managed through various SMF services

TABLE 8-1 Comparison of Oracle Solaris 10 Configuration Tools and Oracle Solaris 11 Configuration Tools
(Continued)

System Configuration Feature, Tool, or Function	Oracle Solaris 10	Oracle Solaris 11
System console service (serial port monitor) configuration	getty, pmadm, ttyadm, ttymon	svc:/system/console-login:terma SMF service svc:/system/console-login:termb SMF service
System configuration (nodename or hostname)	Edit /etc/nodename	svccfg -s sets the config/nodename property of the svc:system/identity:node SMF service to the desired name. You must refresh and restart the service for the changes to take affect. Note – If the system is configured to use DHCP, which is always the case when the Automatic NCP is enabled, the SMF service property is only set if the DHCP server does not provide a value for the nodename/hostname option (DHCP standard option code 12).
Power management	Edit /etc/power.conf or use pmconfig	poweradm
System unconfiguration and reconfiguration	sysidtool, sys-unconfig, sysidconfig, and sysidcfg	sysconfig or the SCI tool
System registration	Auto Registration tool	Oracle Configuration Manager
System recovery	Flash archive features	System recovery involves several steps. See “System Boot, Recovery, and Platform Changes” on page 98.
Printer configuration and administration	LP print commands, Solaris Print Manager	CUPS command-line tools, CUPS Print Manager, and CUPS web browser interface
Locale and time zone configuration	Edit /etc/default/init	Locale: svc:/system/environment:init Time zone: svc:/system/timezone:default

System Configuration Changes and Migration of System Configuration to SMF

In Oracle Solaris 11, certain aspects of system configuration have changed, including the migration of several system, network, and naming services configuration to SMF. For information about naming services that have migrated to SMF, see [Table 7-1](#).

The following key changes are introduced in this release:

- DNS server setup changes – The process for setting up a DNS server has changed in Oracle Solaris 11. For detailed instructions, see “[Administering DNS \(Tasks\)](#)” in *Oracle Solaris Administration: Naming and Directory Services*.
- `/etc/default/init` file is now read-only – Locale and time zone configuration has migrated to SMF. All changes to environment variables should be managed through the new `svc:/system/environment:init` SMF service.

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

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

For more information, see “[Internationalization and Localization Changes](#)” on page 105.

- `/etc/dfstab` file configuration – Publishing and unpublishing a file system share is now administered by using the `zfs` command. See [Chapter 5, “Managing File Systems.”](#)
- `/etc/hostname.<if>`, `/etc/dhcp.<if>`, and `/etc/hostname.ip*.tun*` configuration – Network configuration persistence through the editing of these files is no longer necessary. The `ipadm` and `dladm` commands are now used to manage this type of network configuration. See “[Managing Network Configuration When in Manual Mode](#)” on page 78.
- `/etc/nodename` configuration – A system's identify (nodename) is now configured through the `config/nodename` service property of the `svc:/system/identity:node` SMF service, as shown in the following example:

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

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

- Mapping of a system's nodename – In Oracle Solaris 10, during an installation, the `/etc/hosts` file is updated to map the system's nodename to one of the system's non-loopback IP addresses. In Oracle Solaris 11, this functionality no longer exists. Instead, the host name is mapped to a system's IPv4 and IPv6 addresses. For example:

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

Administrators who prefer to have system nodenames map to the IP address of a non-loopback interface, must manually modify `/etc/hosts` to introduce this mapping, as shown in the following example:

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

- 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” on page 96](#).
- 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 new `svc:/system/timezone:default` SMF service enables you set a system's time zone. See [“Locale and Time Zone Configuration Changes” on page 107](#).

SMF Administrative Changes

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

The different settings by administrator, profile, or manifest are captured in *layers*. Use the `svccfg listprop` command with the new `-l` option to explore the values in each of the layers. For example, `svccfg listprop -l all` prints all of the layers and the values in each layer. In addition, the `svccfg listcust` command can be used to list customizations *only*.

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

If the files cannot be removed, or the administrator does not want the service or instance to run on the system, and disabling the service or instance is not an option, the `svccfg delete`

command is an administrative customization that can be used. The `svccfg delete` command is considered a customization to the way the system is currently installed when the delivering files 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 customization, including customization such as `svccfg delete`, and return to the configuration that is provided by the service manifest, use the `delcust` subcommand of the `svccfg` command with care. For example, you would list and delete all of the customization on `sendmail-client:default` as follows:

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

For more information, see the [svccfg\(1M\)](#) man page..

System Process Changes

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

Process	Description
<code>fsflush</code>	System daemon that flushes pages to disk
<code>init</code>	Initial system process that starts and restarts other processes and SMF components
<code>intrd</code>	System process that monitors and balances system load due to interrupts
<code>kmem_task</code>	System process that monitors memory cache sizes
<code>pageout</code>	System process that controls memory paging to disk
<code>sched</code>	System process that is responsible for OS scheduling and process swapping
<code>vm_tasks</code>	System process with one thread per processor that balances and distributes virtual memory related workloads across CPUs for better performance.

Process	Description
<code>zpool -pool-name</code>	System process for each ZFS storage pool containing the I/O taskq threads for the associated pool

System Console, Terminal Services, and Power Management Changes

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

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

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

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

The `ttymon` program is used to offer login services for these terminals. Each terminal uses a separate instance of the `ttymon` program. Command-line arguments that are passed by the service to the `ttymon` program govern the terminal's behavior. For more information, see [Chapter 16, “Managing the System Console, Terminal Devices, and Power Services \(Tasks\),” in *Oracle Solaris Administration: Common Tasks*](#).

Power Management Configuration

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

Review the following potential power management transition issues:

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

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


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

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

- The GNOME power manager (GPM) feature, which runs when the GUI starts, changes the power management settings. This behavior is intentional to enable the integration of power management administration with GNOME Desktop behavior. See [“Managing System Power Services”](#) in *Oracle Solaris Administration: Common Tasks*.

System Configuration Tools Changes

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

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

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

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

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

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

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

If you do not specify an existing configuration profile, the SCI tool runs during the installation process. The tool enables you to provide specific configuration information for that Oracle Solaris instance and can be used to configure a freshly installed or an unconfigured system. 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. See [Chapter 6, “Unconfiguring or Reconfiguring an Oracle Solaris instance,” in *Installing Oracle Solaris 11 Systems*](#).

Launch the SCI tool from the command line as follows:

```
# sysconfig configure
```

See the [`sysconfig\(1M\)` man page](#) and “[Creating a Configuration Profile Using the SCI Tool](#)” in [Installing Oracle Solaris 11 Systems](#).

System Registration Changes

The Oracle Configuration Manager is used to collect and upload configuration information to the Oracle repository. This data is then used to provide better service to customers. In Oracle Solaris 10, the Auto Registration feature performs a similar function.

The Oracle Configuration Manager is run immediately after an installation, upon the first system reboot. This feature can be disabled prior to rebooting your system after an installation. See “[Using Oracle Configuration Manager](#)” in [Installing Oracle Solaris 11 Systems](#). If you are installing Oracle Solaris 11 by using AI, see “[Opting Out of Oracle Configuration Manager](#)” in [Installing Oracle Solaris 11 Systems](#) for information about disabling the feature prior to an installation with AI.

For more information about installing and administering Oracle Configuration Manager, see the [Oracle Configuration Manager Installation and Administration Guide](#) at http://download.oracle.com/docs/cd/E23562_01/index.htm.

System Boot, Recovery, and Platform Changes

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

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

- If you use a system's service processor (SP) or ILOM to recover from a system problem, accessing a system's SP or ILOM is identical to previous Solaris releases. The differences mostly pertain to how the system is booted after you get to a SPARC based system's ok PROM prompt or to an x86 based system's BIOS.
- In Oracle Solaris 10, you use the flash archive features to create a copy of a UFS or ZFS root environment and then restore the flash archive to recover the system environment, in the case of a system or device failure. In Oracle Solaris 11, the system recovery process includes the following steps:
 - Archiving the root pool snapshots on a remote system
 - Replacing any failed system component or device
 - Recreating the root pool and setting the `bootfs` property
 - Restoring the previously archive root pool snapshots
 - Manually installing the boot blocks

See [Chapter 12, “Archiving Snapshots and Root Pool Recovery,” in *Oracle Solaris Administration: ZFS File Systems*](#).

- Booting for system recovery – If the system cannot be booted, but the failure is not because the root pool is unavailable, you can use new boot options from the installation media or from an installation server to resolve the boot problem. See [“Booting for System Recovery Changes” on page 99](#).

Booting for System Recovery Changes

As in previous Oracle Solaris releases, you might need to boot the system for recovery purposes. The following error and recovery scenarios are similar to previous releases:

- Boot from the installation media or from an install server on the network to recover from a problem that is preventing the system from booting or to recover from a lost root password. On SPARC 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. If a problem with the `menu.lst` file exists, you do not have to mount the BE, just import the root pool, which automatically mounts the `root` file system that contains the boot-related components.

▼ How to Boot the 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 the boot recovery problem:

- 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` option to the `$kernel` line.

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

```
# init 0
ok boot -s
```

```
Boot device: /pci@780/pci@0/pci@9/scsi@0/disk@0,0:a File and args: -s
SunOS Release 5.11 Version 11.0 64-bit
Copyright (c) 1983, 2011, Oracle and/or its affiliates. All rights
reserved.
Booting to milestone "milestone/single-user:default".
Hostname: tardis.central
Requesting System Maintenance Mode
SINGLE USER MODE
```

```
Enter user name for system maintenance (control-d to bypass): root
Enter root password (control-d to bypass): xxxxxxxx
single-user privilege assigned to root on /dev/console.
```

Entering System Maintenance Mode

```
Jan 24 13:23:54 su: 'su root' succeeded for root on /dev/console
Oracle Corporation SunOS 5.11 11.0 November 2011
su: No shell /usr/bin/mybash. Trying fallback shell /sbin/sh.
root@tardis.central:~# TERM =vt100; export TERM
root@tardis.central:~# vi /etc/passwd
root@tardis.central:~# <Press control-d>
logout
svc.startd: Returning to milestone all.
```

- Resolve a problem with a menu.lst boot entry.

First, you must boot from media or the network by using one of the boot methods listed in Step 1. Then, import the root pool and fix the menu.lst entry.

```
x86# zpool import -f rpool
x86# cd /rpool/boot/grub
x86# vi menu.lst
x86# 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
```

Go to the next step to set the root password.

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 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 11.0 64-bit
Copyright (c) 1983, 2011, Oracle and/or its affiliates. All rights
reserved.
Booting to milestone "milestone/single-user:default".
Hostname: tardis.central
Requesting System Maintenance Mode
SINGLE USER MODE
```

```
Enter user name for system maintenance (control-d to bypass): root
Enter root password (control-d to bypass): <Press return>
single-user privilege assigned to root on /dev/console.
Entering System Maintenance Mode
```

```
Jan 24 13:23:54 su: 'su root' succeeded for root on /dev/console
Oracle Corporation SunOS 5.11 11.0 November 2011
root@tardis.central:~# passwd -r files root
New Password: xxxxxx
Re-enter new Password: xxxxxx
passwd: password successfully changed for root
root@tardis.central:~# <Press control-d>
logout
svc.startd: Returning to milestone all.
```

Boot, Platform, and Hardware Changes

Note the following boot, platform, and hardware feature changes in Oracle Solaris 11:

- **x86 platform support is 64-bit only** – Support for booting a 32-bit kernel on x86 platforms has been removed. Systems that have 32-bit hardware must either be upgraded to 64-bit hardware or continue to run Oracle Solaris 10. Note that 32-bit applications are not impacted by this change.
- **Bitmapped console support** – Oracle Solaris 11 includes support for high resolution and color depth consoles. By default, your machine will boot with a 1024x768x16-bit console, unless your video card does not support this setting. In which case, the setting will fall back to 800x600, then finally to 640x480. The console type (and also the older VGA TEXT 640x480 console) can be controlled through both kernel parameters and through options that you specify by editing the GRUB menu at boot time, as follows:
 - B console={text|graphics|force-text}
- **Fast Reboot support on SPARC and x86 platforms** – For SPARC based systems that support Fast Reboot, the boot process is accelerated by skipping certain POST tests. On x86 platforms, Fast Reboot implements an in-kernel boot loader that loads the kernel into memory and then switches to that kernel. 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. Fast Reboot is managed through SMF properties that can be enabled or disabled, as required. See [“Accelerating the Reboot Process”](#) in *Oracle Solaris Administration: Common Tasks*.

- **Removal of support for the SPARC sun4u architecture** – With the exception of the M-series (OPL) hardware, you cannot boot Oracle Solaris 11 on the sun4u architecture. If you attempt to boot Oracle Solaris 11 on one of these systems, the following error message is displayed:

```
Rebooting with command: boot
Error: 'cpu:SUNW,UltraSPARC-IV+' is not supported by this release of Solaris.
NOTICE: f_client_exit: Program terminated!
```

Printer Configuration and Management Changes

In Oracle Solaris 11, 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 in Oracle Solaris 11:

- **Solaris Print Manager removed** – Solaris Print Manager is no longer available. CUPS Print Manager replaces this tool.
- **LP print commands removed** – Several LP print commands, files, and services are no longer available. Some LP print commands, for example `lp`, `lpadmin`, `lpc`, `lpr` are still available. However, these commands are now managed by CUPS. For a complete list of the commands, services, and files that have been removed, refer to [“Removal of Legacy System Management Commands, Tools, Services, and Files”](#) on page 17.
- **Printer configuration is no longer stored in the NIS naming service** – 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 *Oracle Solaris Administration: Common Tasks*.
- **Removal of the `/etc/printers.conf` file** - In Oracle Solaris 10 and previous releases, the `/etc/printers.conf` file is where details about all of the printers that are set up by using the LP print service is stored. In Oracle Solaris 11, this file is no longer generated after a fresh installation. Any information about printers that were configured by using `lp print`

commands is removed. The resulting behavior is as though these printers were never configured on the system. Any previously configured printers must be reconfigured by using CUPS. Note that you do not need to delete existing printers prior to reconfiguring them. For information about setting up your printing environment to work with CUPS, see [“How to Set Up Your Printing Environment After Installing Oracle Solaris 11”](#) on page 104.

- Changes to behavior and use of the `~/ .printers` file - Printers that are configured on a per-user basis in the `~/ .printers` file no longer work. Printer configuration is now solely managed by using CUPS. The default printer can be set on a per-user basis, by setting either the `LPDEST` or `PRINTER` environment variables, or by using the new `lpoptions` command. The `lpoptions` command creates an `~/ .lpoptions` file that has the default printer entry listed within that file. By default, all print jobs are directed to this printer.

You can list specific options for a printer as follows:

```
# lpoptions -l printer-name
```

You can 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 *Oracle Solaris Administration: Common Tasks*.

- The `lp` entry in the `/etc/passwd` file is now as follows:

```
lp:x:71:8:Line Printer Admin:/:
```

The `lp` entry in the `/etc/group` file remains as it is in previous releases.

See [Chapter 15, “Setting Up and Administering Printers by Using CUPS \(Tasks\)”](#) in *Oracle Solaris Administration: Common Tasks*.

▼ How to Set Up Your Printing Environment After Installing Oracle Solaris 11

Use the following procedure to set up your printing environment to work with CUPS after a fresh installation. If you are upgrading from Oracle Solaris 11 Express, see [“How to Set Up Your Printing Environment After Upgrading From Oracle Solaris 11 Express”](#) on page 139.

- 1 **Verify that the `cups/scheduler` and the `cups/in-lpd` SMF services are enabled.**

```
# svcs -a | grep cups/scheduler
# svcs -a | grep cups/in-lpd
```

- 2 **If these services are not enabled, enable them.**

```
# svcadm enable cups/scheduler
# svcadm enable cups/in-lpd
```


3 Check that the `printer/cups/system-config-printer` package is installed.

```
# pkg info print/cups/system-config-printer
```

- If the package is already installed, you are ready to configure printers by using CUPS.
- If the package is not installed, install the package:


```
# pkg install print/cups/system-config-printer
```

Next Steps For instructions, see “Setting Up and Administering Printers by Using CUPS Command-Line Utilities” in *Oracle Solaris Administration: Common Tasks*.

Internationalization and Localization Changes

Oracle Solaris 11 introduces the following important internationalization and localization changes:

- **Language and locale support** – Oracle Solaris now supports over 200 locales. By default, only a core set of locales is installed on the system.

Core locales are a set of locales that 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 locale changes in this release include the addition of the Portuguese – Brazilian locale and the removal of the Swedish locale.

Core locales typically provide better support at the level of localized messages than the 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.

- **Language and locale packaging** – The locale facet mechanism replaces the `localeadm` command in Oracle Solaris 11. In Oracle Solaris 10, optional package components, such as documentation, localization, or debug files are split into separate packages. In Oracle Solaris 11, IPS enables you to store these various package components in the same package by using

special tags that are called *facets*. Facets simplify the packaging process, as well as minimize disk space usage. Locale facets are used to mark files or actions that are language or locale-specific.

Display the status of the facets on a system as follows:

```
$ pkg facet
```

The following example shows how to install the Danish locale and any available translations:

```
# pkg change-facet facet.locale.da=True
# pkg change-facet facet.locale.da_DK=True
```

Note – Non-UTF-8 locales, such as da_DK.ISO8859-1, are packaged separately. To enable these locales, install the system/locale/extra package.

See “Configuring Package Variants and Facets” in *Adding and Updating Oracle Solaris 11 Software Packages*.

- **Setting a system's default locale** – In Oracle Solaris 10, the default system locale is configured in `/etc/default/init`. In Oracle Solaris 11, this file is obsoleted, and the configuration has moved to the corresponding properties of the `svc:/system/environment:init` SMF service. See “Locale and Time Zone Configuration Changes” on page 107.
- **Short form locales** – Solaris 10 supports a number of short form locale names that do not follow the `language_country.encoding[@modifier]` format, for example, ja,de,de_AT, and so on. These locales are not supported in Oracle Solaris 11. For more information, 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 in Oracle Solaris 11. Locale name aliases are accepted and mapped to the corresponding canonical locale names. For example, the de locale is mapped to the canonical locale de_DE.ISO8859-1. For all of the locale name mappings, see `locale_alias(5)`.
- **Keyboard layout setting for the console** – In Oracle Solaris 11, the keyboard layout setting for the console has migrated to SMF. To change the keyboard layout in the console, modify the `keymap/layout` property of the `system/keymap:default` SMF service. The following example shows how to set the UK-English layout for the console.

```
# svccfg -s keymap:default setprop keymap/layout = UK-English
# svcadm refresh keymap
# svcadm restart keymap
```

Note – The keyboard layout in the graphical interface is set separately.

Locale and Time Zone Configuration Changes

Locale and time zone configuration is set in the `/etc/default/init` file in Oracle Solaris 10. In Oracle Solaris 11, this configuration is managed through SMF service properties.

For example, to change the default system locale to `fr_FR.UTF-8`, you would configure the SMF service property as follows:

```
# svccfg -s svc:/system/environment:init \  
setprop environment/LANG = astring: fr_FR.UTF-8  
# svcadm refresh svc:/system/environment
```

The service must be refreshed for changes to take affect.

1. For the time zone setting, make sure the TZ in the `/etc/default/init` file is set to `localtime`.

```
grep TZ /etc/default/init  
TZ=localtime
```

2. Then, set the time zone SMF property to the required time zone.

```
# svccfg -s timezone:default setprop timezone/localtime= astring: US/Mountain  
# svcadm refresh timezone:default
```

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

Managing Security

This chapter describes security feature changes in Oracle Solaris 11.

This is a list of the information that is in this chapter:

- “Security Feature Changes” on page 109
- “Roles, Rights, Privileges, and Authorizations” on page 111
- “File and File System Security Changes” on page 114

Security Feature Changes

Oracle Solaris 11 introduces the following key security changes:

- **Auditing** – Auditing is now a service and is enabled by default. No reboot is required when disabling or enabling this service. The `auditconfig` command is used to view information about audit policy and to change audit policy. The auditing of public objects generates less noise in the audit trail. In addition, auditing of non-kernel events has no performance impact.

For information about creating a ZFS file system for audit files, see “How to Create ZFS File Systems for Audit Files” in *Oracle Solaris Administration: Security Services*.

- **Basic Audit Reporting Tool (BART)** – The default hash that is used by BART is now SHA256, not MD5. In addition to SHA256 being the default, you can also select the hash algorithm. See Chapter 5, “Verifying File Integrity by Using BART,” in *Oracle Solaris Administration: Security Services*.
- **Cryptographic Framework** – This feature now includes more algorithms, mechanisms, plug-ins, and support for Intel and SPARC T4 hardware acceleration. Also, Oracle Solaris 11 provides better alignment with the NSA Suite B cryptography.
- **Kerberos DTrace providers** – A new DTrace USDT provider that provides probes for Kerberos messages (Protocol Data Unit) has been added. The probes are modeled after the Kerberos message types that are described in RFC4120.
- **Key Management enhancements:**

- PKCS#11 keystore support for RSA keys in the Trusted Platform Module
- PKCS#11 access to Oracle Key Manager for centralized enterprise key management
- **lofi command changes** – lofi now supports the encryption of block devices. See [lofi\(7D\)](#).
- **profiles command changes** – In Oracle Solaris 10, the command is only used to list profiles for a specific user or role, or a user's privileges for specific commands. In Oracle Solaris 11, you can also create and modify profiles in files and in LDAP by using the profiles command, See [profiles\(1\)](#).
- **sudo command** – The sudo command is new in Oracle Solaris 11. This command generates Oracle Solaris audit records when running commands. The command also drops the proc_exec basic privilege, if the sudoers command entry is tagged as NOEXEC.
- **ZFS file system encryption** – ZFS file system encryption is designed to keep your data secure. See “[Encrypting ZFS File Systems](#)” on page 115.
- **rstchown property** – The rstchown tunable parameter that is used in previous releases to restrict chown operations is now a ZFS file system property, rstchown, and is also a general file system mount option. See [Oracle Solaris Administration: ZFS File Systems and mount\(1M\)](#).

If you attempt to set this obsolete parameter in the /etc/system file, the following message is displayed:

```
sorry, variable 'rstchown' is not defined in the 'kernel'
```

Network Security Features

The following network security components are supported in this release:

- **Internet Key Exchange (IKE) and IPsec** – IKE now includes more Diffie-Hellman groups and can also use Elliptic Curve Cryptography (ECC) groups. IPsec includes AES-CCM and AES-GCM modes and is now capable of protecting network traffic for the Trusted Extensions feature of Oracle Solaris (Trusted Extensions).
- **IPfilter Firewall** – IPfilter Firewall, which is similar to the open source IPfilter feature, is compatible, manageable, and now highly integrated with SMF. This feature enables selective access to ports, based on IP address.
- **Kerberos** – Kerberos is now capable of mutual authentication of clients and servers. Also, support for initial authentication by using X.509 certificates with the PKINIT protocol has been introduced. See [Part V, “Kerberos Service,” in Oracle Solaris Administration: Security Services](#).
- **Secure by Default** – In Oracle Solaris 10, this feature was introduced, but was net services limited and was also turned off by default. In Oracle Solaris 11, this feature is enabled. The Secure by Default feature is used to disable and protect several network services from attack and provides minimization of network exposure. Note that only SSH is enabled.
- **SSH** – Support for host and user authentication by using X.509 certificates is now available.

Removed Security Features

The following security features are excluded from Oracle Solaris 11:

- Automated Security Enhancement Tool (ASET) – The ASET functionality is replaced by a combination of IPfilter, which includes `svc.ipfd`, BART, SMF, and other security features that are supported in Oracle Solaris 11.
- Smartcard – Smartcard support is no longer available.

Roles, Rights, Privileges, and Authorizations

The following information describes how roles, rights and privileges work in Oracle Solaris 11:

- **Assign versus delegate authorizations** – Oracle Solaris provides authorizations for delegating specific administrative rights to individual users and roles to implement separation of duty. In Oracle Solaris 10, authorizations ending in `.grant` are required to delegate an authorization to another user. In Oracle Solaris 11, two new suffixes, `.assign` and `.delegate`, for example, `solaris.profile.assign` and `solaris.profile.delegate`, are used. 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.
- **Media Restore rights profile** – This rights profile and set of authorizations can escalate the privileges of a non root account. The profile exists, but is not part of any other rights profile. Because the Media Restore rights profile provides access to the entire root file system, its use is a possible escalation of privilege. Deliberately altered files or substitute media could be restored. By default, the root role includes this rights profile.
- **Primary Administrator profile removed** – The initial user that is created at installation time is given the following roles and rights:
 - root role
 - System Administrator rights profile
 - Access to the sudo command for all commands that are run as root
- **Role authentication** – You can specify either user or role for the `roleauth` keyword. See [user_attr\(4\)](#).
- **root as a Role** – root is now a role by default, therefore, not *anonymous* and cannot remotely log in to a system. For information about changing the root role to a user, see “How to Change the root Role Into a User” in *Oracle Solaris Administration: Security Services*.
- **Oracle Solaris 11 basic privileges include the following:**
 - `file_read`

- `file_write`
- `net_access`
- **Profile shell versions of regular shells** – Every regular shell now has its own profile version. The following profile shells are available:
 - `pfbash`
 - `pfcs`
 - `pfksh`
 - `pfksh93`
 - `pfrksh93`
 - `pfsh`
 - `pftcsh`
 - `pfzsh`

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

- **Rights profiles** – The `user_attr`, `prof_attr`, and `exec_attr` databases are now read-only. These local files databases are assembled from fragments that are located in `/etc/user_attr.d`, `/etc/security/prof_attr.d`, and `/etc/security/exec_attr.d`. The fragment files are not merged into a single version of the file, but left as fragments. This change enables packages to deliver complete or partial RBAC profiles. Entries that are added to the local files repository with the `useradd` and `profiles` commands are added to the `local-entries` file in the fragment directory. To add or modify a profile, use the `profiles` command. See “[How to Create or Change a Rights Profile](#)” in *Oracle Solaris Administration: Security Services*.
- **Stop rights profile** – This profile enables administrators to create restricted accounts. See “[RBAC Rights Profiles](#)” in *Oracle Solaris Administration: Security Services*.
- **`pfsh script` command** – This command now runs the same as the `pfsh -c script` command. Previously, commands within a script would not be able to take advantage of RBAC, unless the script specified a profile shell as its first line. This rule required you to modify any scripts to use RBAC, which is now unnecessary because the caller of the script (or an ancestor within the session) can specify a profile shell.
- **`pfexec` command** – This command is now no longer `setuid root`. The new `PF_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 now searches for an entry in the Forced Privilege RBAC profile in `exec_attr(4)` to determine which privileges the program should run with. Instead of having the program start with `uid root` and all privileges, the program runs with the current `uid` and only the additional privileges that the Forced Privilege RBAC execution profile have assigned to that path name.

Viewing Your Privileges

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 9-1 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 -vl proc_clock_highres
      Allows a process to use high resolution timers.

```

File and File System Security Changes

The following sections describe changes to file and file system security.

aclmode Property Is Reintroduced

The `aclmode` property that determines how the ACL permissions on a file are modified during a `chmod` operation is reintroduced in Oracle Solaris 11. The `aclmode` values are `discard`, `mask`, and `passthrough`. The `discard` default value is the most restrictive, and the `passthrough` value is the least restrictive.

EXAMPLE 9-2 ACL Interaction With `chmod` Operations on ZFS Files

The following examples illustrate how specific `aclmode` and `aclinherit` property values influence the interaction of existing ACLs with a `chmod` operation that either reduces or expands any existing ACL permissions to be consistent with the ownership of a group.

In this example, the `aclmode` property is set to `mask` and the `aclinherit` property is set to `restricted`. The ACL permissions in this example are displayed in compact mode, which more easily illustrates changing permissions.

The original file and group ownership and ACL permissions are as follows:

```
# zfs set aclmode=mask pond/whoville
# zfs set aclinherit=restricted pond/whoville

# ls -lv file.1
-rwxrwx---+ 1 root    root      206695 Aug 30 16:03 file.1
  user:amy:r-----a-R-c---:-----:allow
  user:rory:r-----a-R-c---:-----:allow
  group:sysadmin:rw-p--aARWc---:-----:allow
  group:staff:rw-p--aARWc---:-----:allow
  owner@:rwxp--aARWcCos:-----:allow
  group@:rwxp--aARWc--s:-----:allow
  everyone@:-----a-R-c--s:-----:allow
```

A `chown` operation changes the file ownership on `file.1` and the output is now seen by the owning user, `amy`. For example:

```
# chown amy:staff file.1
# su - amy
$ ls -lv file.1
-rwxrwx---+ 1 amy      staff      206695 Aug 30 16:03 file.1
  user:amy:r-----a-R-c---:-----:allow
  user:rory:r-----a-R-c---:-----:allow
  group:sysadmin:rw-p--aARWc---:-----:allow
  group:staff:rw-p--aARWc---:-----:allow
  owner@:rwxp--aARWcCos:-----:allow
  group@:rwxp--aARWc--s:-----:allow
  everyone@:-----a-R-c--s:-----:allow
```

EXAMPLE 9-2 ACL Interaction With chmod Operations on ZFS Files (Continued)

The following chmod operation changes the permissions to a more restrictive mode. In this example, the modified sysadmin group's and staff group's ACL permissions do not exceed the owning group's permissions.

```
$ chmod 640 file.1
$ ls -lV file.1
-rw-r-----+ 1 amy      staff      206695 Aug 30 16:03 file.1
      user:amy:r-----a-R-c---:-----:allow
      user:rory:r-----a-R-c---:-----:allow
      group:sysadmin:r-----a-R-c---:-----:allow
      group:staff:r-----a-R-c---:-----:allow
      owner@:rw-p--aARWcCos:-----:allow
      group@:r-----a-R-c--s:-----:allow
      everyone@:-----a-R-c--s:-----:allow
```

The following chmod operation changes the permissions to a less restrictive mode. In this example, the modified sysadmin group's and staff group's ACL permissions are restored to allow the same permissions as the owning group.

```
$ chmod 770 file.1
$ ls -lV file.1
-rwxrwx---+ 1 amy      staff      206695 Aug 30 16:03 file.1
      user:amy:r-----a-R-c---:-----:allow
      user:rory:r-----a-R-c---:-----:allow
      group:sysadmin:rw-p--aARWc---:-----:allow
      group:staff:rw-p--aARWc---:-----:allow
      owner@:rwxp--aARWcCos:-----:allow
      group@:rwxp--aARWc--s:-----:allow
      everyone@:-----a-R-c--s:-----:allow
```

Encrypting ZFS File Systems

In previous Oracle Solaris releases and in this release, the Cryptographic Framework feature provides the encrypt, decrypt, and mac commands to encrypt files.

Oracle Solaris 10 does not support ZFS encryption, but Oracle Solaris 11 supports the following ZFS encryption features:

- ZFS encryption is integrated with the ZFS command set. Like other ZFS operations, key change and rekey operations are performed online.
- You can use your existing storage pools as long as they are upgraded. You have the flexibility of encrypting specific file systems.
- ZFS encryption is inheritable to descendent file systems. Key management can be delegated through ZFS delegated administration.
- Data is encrypted by using AES (Advanced Encryption Standard) with key lengths of 128,192, and 256 in the CCM and GCM operation modes.

- ZFS encryption uses the Cryptographic Framework feature, which gives it access to any available hardware acceleration or optimized software implementations of the encryption algorithms automatically.

EXAMPLE 9-3 Creating an Encrypted ZFS File System

The following example shows how to create an encrypted ZFS file system. The default encryption policy is to prompt for a passphrase, which must be a minimum of 8 characters in length.

```
# zfs create -o encryption=on tank/data
Enter passphrase for 'tank/data': xxxxxxxx
Enter again: xxxxxxxx
```

The default encryption algorithm is aes-128-ccm when a file system's encryption value is on.

After an encrypted file system is created, it cannot be unencrypted. For example:

```
# zfs set encryption=off tank/data
cannot set property for 'tank/data': 'encryption' is readonly
```

For more information, see “[Encrypting ZFS File Systems](#)” in *Oracle Solaris Administration: ZFS File Systems*.

Immutable Zones

New in Oracle Solaris 11, the `file-mac-profile` feature enables you to run zones with a read-only root file system. This feature enables you to choose between four predefined profiles that determine how much of a zone file system is read-only only, even for processes that have `allzone` privileges. See “[zonecfg file-mac-profile Property](#)” in *Oracle Solaris Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.

Managing Oracle Solaris Releases in a Virtual Environment

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

This is a list of the information that is in this chapter:

- “Installing and Managing Oracle Solaris 11 Virtualization Features” on page 117
- “Oracle Solaris 11 Zone Features” on page 118
- “Transitioning an Oracle Solaris 10 Instance to an Oracle Solaris 11 System” on page 120

Installing and Managing Oracle Solaris 11 Virtualization Features

The following table provides a brief description of virtualization features that are supported in Oracle Solaris 11.

TABLE 10-1 Oracle Solaris 11 Virtualization Features

Oracle Solaris 11 Feature	Description	Oracle Solaris 10 Support	For More Information
Oracle Solaris Resource Manager product components (also referred to as resource management)	Features that enable you to control how applications use available system resources	Yes	Part I, “Oracle Solaris Resource Management,” in <i>Oracle Solaris Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management</i>
Oracle VM Server for SPARC (Sun Logical Domains)	Hypervisor-based virtualization for SPARC servers	Yes	<i>Oracle VM Server for SPARC 2.1 Administration Guide</i>
Oracle VM Server 3.0 for x86 (Xen)	Hypervisor-based virtualization for x86 based servers	Yes	http://www.oracle.com/us/technologies/virtualization/oraclevm/index.html

TABLE 10-1 Oracle Solaris 11 Virtualization Features (Continued)

Oracle Solaris 11 Feature	Description	Oracle Solaris 10 Support	For More Information
Oracle VM VirtualBox	Hosted workstation and server virtualization for x86 based systems	Yes	http://www.oracle.com/technetwork/server-storage/virtualbox/downloads/index.html
Oracle Solaris Zones	A zone is a virtualized operating system environment created within a single instance of the Oracle Solaris operating system	Yes	Part II, “Oracle Solaris Zones,” in <i>Oracle Solaris Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management</i>

Oracle Solaris 11 Zone Features

- **Oracle Solaris 10 branded zones** – Oracle Solaris 10 Zones provide an Oracle Solaris 10 environment on Oracle Solaris 11. You can migrate an Oracle Solaris 10 system or zone to a `solaris10` zone on an Oracle Solaris 11 system in the following ways:
 - Create a zone archive and use the archive to create an `s10zone` on the Oracle Solaris 11 system. See “[Transitioning an Oracle Solaris 10 Instance to an Oracle Solaris 11 System](#)” on page 120.
 - 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 *Oracle Solaris Administration: Oracle Solaris Zones, Oracle Solaris 10 Zones, and Resource Management*.

The `zonep2vchk` tool identifies issues that could affect the migration, and creates zone configuration output for the target zone.

- **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 Systems*.
- **Whole root zones only** – Oracle Solaris Zones are whole-root type only, but you can configure your zones in a more flexible way, such as when disk space is limited or if you prefer a read-only zone root configuration. By default, zone boot environments are compressed.
- **Legacy branded zones** – The following legacy branded zone features are only supported in Oracle Solaris 10 releases:
 - Linux brand (`lx`)
 - Oracle Solaris 8 Containers (`solaris8`)
 - Oracle Solaris 9 Containers (`solaris9`)

- Exclusive IP zones by default** – Exclusive-IP zones enable you to assign a separate IP stack per zone. Each zone has the flexibility to configure IP within that stack completely separate to other zones. Administrators can easily observe network traffic, per zone, and apply individual network resources. However, in previous versions of Oracle Solaris this was dependent on the number of physical NICs an administrator had per system. The addition of network virtualization provides administrators enhanced flexibility with regard to managing zones, without the restrictions of physical network hardware. Newly created zones in Oracle Solaris 11 will be exclusive-IP zones with a VNIC, `net0`, whose underlying lower link is automatically selected at boot time. See [Chapter 10, “Managing Oracle Solaris Releases in a Virtual Environment.”](#)
- Network virtualization for zones** – Oracle Solaris 11 network virtualization features can be applied to a zone by creating a virtual NIC (VNIC) for the zone and applying bandwidth limits and traffic flows to the zone's assigned VNIC. The VNIC is created when the zone boots and deleted when the zone halts, and will be created within the non-global zone's datalink namespace. This feature allows you to provision a zone without having to learn the details of network configuration and topology. If you want to assign a preexisting datalink to an exclusive-IP zone, you can still do so during the zone configuration.

For example, create a virtual NIC, limit the SPEED of the VNIC, create an address for it, and then assign it to zone.

```
# dladm create-vnic -l net0 -p maxbw=600 vnic0
# ipadm create-addr -T static -a local=x.x.x.x/24 vnic0/v4static
.
.
zonecfg:s11zone> set ip-type=exclusive
zonecfg:s11zone> add net
zonecfg:s11zone:net> set physical=vnic0
zonecfg:s11zone:net> end
.
.
```

The `ip-type` value for the zone can either be shared or exclusive:

- The `ip-type=exclusive` value means that you are dedicating a datalink, which can be a virtual (VNIC) for exclusive use by the zone. This strategy brings some of the network stack management benefits to the zone. Historically, this hasn't been practical if a system has many zones, but only a maximum of 4 network interfaces.

Exclusive IP is now the recommended *ip-type* for zones. The `set physical` value identifies the system's network interface cards that is assigned to the zone. Using an `ip-type` of `exclusive` allows the zone to manage its IP stack directly.
- If `ip-type=shared` was identified in the above example, you would have to specify an IP address and other resources.
- NFS server support in non-global zones** – You can share file systems in a non-global zones by using the NFS protocol. The SMB (CIFS) sharing protocol is not currently available in a non-global zone.
- Zone monitoring** – System resources that are consumed by non-global zones can be monitored by using the `zonestat` command.

Oracle Solaris 10 Branded Zones Preparation

Prepare for migrating a Oracle Solaris 10 OS instance or zone to your Oracle Solaris 11 system.

- Confirm that your Oracle Solaris 10 instance or zone is running the Oracle Solaris 10 9/10 release, which is the minimum OS requirement.
- Copy the Oracle Solaris 11 `/usr/sbin/zonep2vchk` script to any location within the Oracle Solaris 10 system. Run this script to determine if any issues would prevent the Oracle Solaris 10 zone or instance from running successfully on a Oracle Solaris 11 system.

Keep in mind that this script is only for system migration purposes.

- Enable the Oracle Solaris 10 package and patch tools.

To use the Oracle Solaris 10 package and patch tools in your Oracle Solaris 10 zones, install the following patches on your source Oracle Solaris 10 system before the image is created.

- 119254-75, 119534-24, 140914-02 (SPARC platforms)
- 119255-75, 119535-24 and 140915-02 (x86 platforms)

The physical to virtual (P2V) process works without the patches, but the package and patch tools do not work properly within the Oracle Solaris 10 zones.

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

You can transition your Oracle Solaris 10 environment to a non-global zone on an Oracle Solaris 11 system by creating a zone archive and migrating the zone archive to an Oracle Solaris 11 system. The following steps describe this process.

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

```
s11sysB# pkg install system/zones/brand/brand-solaris10
```

2. Copy the `zonep2vchk` script from your Oracle Solaris 11 system to the Oracle Solaris 10 instance or system to identify any issues that might prevent the instance from running as a `solaris10` zone.

```
s10sys# ./zonep2vchk
--Executing Version: 1.0.5-11-15652

- Source System: tardis
  Solaris Version: Oracle Solaris 10 8/11 s10s_u10wos_17b SPARC
  Solaris Kernel: 5.10 Generic_147440-01
  Platform: sun4u SUNW,Sun-Fire-V440

- Target System:
  Solaris_Version: Solaris 10
  Zone Brand: native (default)
  IP type: shared
```



```
--Executing basic checks
```

```
.
```

3. Create a ZFS file system that will contain the flash archive of the Oracle Solaris 10 system instance, if necessary.

Then, create a NFS share of the ZFS file system on your Oracle Solaris 11 system. For example:

```
s11sysB# zfs create pond/s10archive
s11sysB# zfs set share=name=s10share,path=/pond/s10archive,prot=nfs,root=s10sysA
pond/s10archive
name=s10share,path=/pond/s10archive,prot=nfs,sec=sys,root=s10sysA
s11sysB# zfs set sharenfs=on pond/s10archive
```

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

```
s10sysA# hostid
8439b629
```

5. Create an archive of the Oracle Solaris 10 instance that you would like to migrate to a non-global zone on the Oracle Solaris 11 system.

```
s10sysA# flarcreate -S -n s10sysA -L cpio /net/s11sysB/pond/s10archive/s10.flar
```

6. Create a ZFS file system for the Oracle Solaris 10 zone.

```
s11sysB# zfs create -o mountpoint=/zones pond/zones
s11sysB# chmod 700 /zones
```

7. Create the non-global zone for the Oracle Solaris 10 instance.

```
s11sysB# zonecfg -z s10zone
s10zone: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:s10zone> create -t SYSsolaris10
zonecfg:s10zone> set zonepath=/zones/s10zone
zonecfg:s10zone> set ip-type=exclusive
zonecfg:s10zone> add anet
zonecfg:s10zone:net> set lower-link=auto
zonecfg:s10zone:net> end
zonecfg:s10zone> set hostid=8439b629
zonecfg:s10zone> verify
zonecfg:s10zone> commit
zonecfg:s10zone> exit
```

8. Install the Oracle Solaris 10 non-global zone.

```
s11sysB# zoneadm -z s10zone install -u -a /pond/s10archive/s10.flar
A ZFS file system has been created for this zone.
Progress being logged to /var/log/zones/zoneadm.201110921T135935Z.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
```

User Account Management and User Environment Changes

This chapter provides information about the management of user accounts, groups, roles, and a user's environment in Oracle Solaris 11.

This is a list of the information that is in this chapter:

- “Tools for Managing User Accounts, Groups, and Roles” on page 123
- “Creating and Managing User Accounts, Groups, and Roles” on page 124
- “User Environment Feature Changes” on page 127

Tools for Managing User Accounts, Groups, and Roles

Note – The Solaris Management Console graphical tool and all of its associated command-line tools have been removed. To create and manage users, groups, and roles, use the command-line tools that are described or referenced within this chapter.

TABLE 11-1 Tools for User Accounts, Groups, and Roles Management

Tool Name	Description	For More Information
useradd, groupadd, roleadd	Commands for adding users, groups, and roles.	Creating and Managing User Accounts, Groups, and Roles “How to Create a Role” in Oracle Solaris Administration: Security Services
usermod, groupmod, rolemod	Commands for modifying users, groups, and roles.	Oracle Solaris Administration: Security Services

TABLE 11-1 Tools for User Accounts, Groups, and Roles Management (Continued)

Tool Name	Description	For More Information
userdel, groupdel, roledel	Commands for deleting users, groups, and roles.	“How to Delete a User” in <i>Oracle Solaris Administration: Common Tasks</i> and <code>userdel(1M)</code> <code>groupdel(1M)</code> , <code>roledel(1M)</code>

Creating and Managing User Accounts, Groups, and Roles

In Oracle Solaris 11, you create and manage users, groups, and roles by using command-line tools only. There is currently no GUI tool for performing these tasks. In addition, the command-line tools that are used by the Solaris Management Console are no longer available. See “Removal of Legacy System Management Commands, Tools, Services, and Files” on page 17.

For information about managing using accounts and groups, see Chapter 2, “Managing User Accounts and Groups (Overview),” in *Oracle Solaris Administration: Common Tasks* and Chapter 3, “Managing User Accounts and Groups (Tasks),” in *Oracle Solaris Administration: Common Tasks*.

User Passwords

The default password hashing algorithm is now SHA256. This password hash is similar to the following:

```
$5$cgQk2iUy$AhHtVGx5Qd0.W3NCKj1kb8.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.

Passwords are encoded by using one of the other `crypt(3c)` algorithms, including the SHA256 algorithm, which is the default in the `policy.conf` file. Thus, passwords can now be much longer than eight characters. See `policy.conf(4)`.

Creating User Accounts

User account creation and management in Oracle Solaris 11 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.
- The `useradd` command relies on the automount service, `svc:/system/filesystem/autofs`, to mount home directories. This service should never be disabled. Each home directory entry for a user in the `passwd` database uses the format, `/home/username`, which is an `autofs` trigger that is resolved by the automounter through the `auto_home` map.
- The `useradd` command automatically creates entries in the `auto_home` map that correspond to the pathname that is specified by using the `-d` option of this command. If the pathname includes a remote host specification, for example, `foobar:/export/home/jdoe`, then the home directory for `jdoe` must be created on the system `foobar`. The default pathname is `localhost:/export/home/user`. Because `/export/home` is the mount point for a ZFS dataset, the user's home directory is created as a child ZFS dataset, with the ZFS permission to take snapshots delegated to the user. If a pathname that does not correspond to a ZFS dataset is specified, then a regular directory is created. If the `-S ldap` option is specified, then the `auto_home` map entry is updated on the LDAP server instead of the local `auto_home` map.

Modifying User Accounts

In Oracle Solaris 10, audit attributes cannot be assigned by using the `usermod` command. In Oracle Solaris 11, the `usermod` command works with LDAP and files. All security attributes can be assigned to a user by using this mechanism.

For example, an administrator can add a role to a user's account by using the `usermod` command.

```
# roleadd -K roleauth=user -P "Network Management" netmgt
# usermod -R +netmgt jdoe
```

See [usermod\(1M\)](#) for additional examples.

Creating Roles

Roles can be created locally and in an LDAP repository. To create a role and assign an initial password, you must be assigned the User Management rights profile. To assign security attributes to the role, you must be assigned the User Security rights profile. The important

difference for role authentication in Oracle Solaris 11 is the addition of the `roleauth=user` keyword and the `pam_tty_tickets` module for authentication caching. Note that the `pam_tty_tickets` module is not enabled by default. To add the module, edit the `/etc/pam.conf` file as follows:

```
# vi /etc/pam.conf
su  auth    required    pam_unix_cred.so.1
su  auth    sufficient  pam_tty_tickets.so.1
su  auth    requisite   pam_authtok_get.so.1
su  auth    required    pam_dhkeys.so.1
su  auth    required    pam_unix_auth.so.1
```

The entire `su` stack is required. The `pam_tty_tickets.so.1` module provides the cache. See [pam.conf\(4\)](#). For instructions on creating a role, see [“How to Create a Role”](#) in *Oracle Solaris Administration: Security Services*.

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 the share is published, which involves the following actions:

- The file system share is created by using the `zfs set share` command. At this time, specific share properties can be defined. If share properties are not defined, the default property values are used.
- The NFS or SMB share is published by setting the `sharenfs` or `sharesmb` property. The share is published permanently until the property is set to `off`.

See [“How to Share Home Directories That Are Created as ZFS File Systems”](#) in *Oracle Solaris Administration: Common Tasks*.

How Home Directories Are Mounted in Oracle Solaris 11

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 *Oracle Solaris Administration: Common Tasks*.

User Environment Feature Changes

Oracle Solaris 11 introduces user environment and command-line argument changes that include the following:

- Shell changes - The default shell, `/bin/sh`, is now linked to `ksh93`. The default user shell is the Bourne-again (`bash`) shell.
 - The legacy Bourne shell is available as `/usr/sunos/bin/sh`.
 - The legacy `ksh88` is available as `/usr/sunos/bin/ksh` from the `shell/ksh88` package.
 - Korn shell compatibility information is available in `/usr/share/doc/ksh/COMPATIBILITY`.
- The default user path is `/usr/bin`. The default path for the root role is `/usr/bin:/usr/sbin`.
- 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.
- The `MANPATH` environment variable is no longer required. The `man` command determines the appropriate `MANPATH` based on the `PATH` environment variable setting.
- Development 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.
- Files that were previously in the `/usr/sfw` directory are now in `/usr/bin`.

Default Login Shell and PATH Environment Variable

In Oracle Solaris 10, the default scripting shell (`/bin/sh`) is the Bourne shell. In Oracle Solaris 11, `/bin/sh` is the Korn shell (`ksh93`), and the default interactive shell is the Bourne-again (`bash`) shell. When used as a login shell, `bash` retrieves configuration information from the first instance of `.bash_profile`, `.bash_login`, or `.profile` file.

The default `PATH` environment variable for `bash` is `/usr/bin:/usr/sbin`, and `/usr/gnu/bin` now comes before `/usr/bin`.

Oracle Solaris 11 Man Page Changes

The following man page features are new or have changed:

- **Locating information** – 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 utilize index files for searching. A new SMF service, `svc:/application/man-index:default`, triggers the automatic regeneration of new index files whenever new man pages are added to the `/usr/share/man` and `/usr/gnu/share/man` directories, if these directories exist. This service is enabled by default.

- **Package Name Change** – The `SUNWman` package that contained the Oracle Solaris man pages in previous releases has changed to the smaller `system/manual` package. The bulk of the man pages are now packaged separately with their component technology packages. For example, `ls.1m` for the `/usr/bin/ls` command is part of the `system/core-os` package.
- **Man Page Display** – If the man pages are not displaying on your system, you can enable the man page display as follows:

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


Using Oracle Solaris Desktop Features

This chapter describes the desktop features that are supported in Oracle Solaris 11.

This is a list of the information that is in this chapter:

- “Oracle Solaris 11 Desktop Feature Summary” on page 129
- “Desktop Features That Have Been Removed” on page 132
- “Xorg Family of Servers” on page 133
- “Troubleshooting Desktop Transition Issues” on page 134

Oracle Solaris 11 Desktop Feature Summary

The default desktop environment in Oracle Solaris 11 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. For more information, see “[Installing the Oracle Solaris Desktop Software Package After an Installation](#)” on page 134.

Other new desktop features include the following:

- Accessibility feature enhancements
- Bluefish HTML editor
- Compiz OpenGL based window manager
- D-Bus IPC Framework
- Evince PDF viewer

- GIMP image editing program
- GNOME Python bindings
- Gobby text editing collaboration tool
- Multimedia support enhancements
- Planner and openproj project management tools
- Trusted Extensions integration
- xchat IRC client
- Xserver features that augment the desktop, such as virtual terminal (VT) switching

Key Desktop Features

The following key features are new or enhanced in Oracle Solaris 11:

- **Accessibility enhancements** – Users with disabilities can use a wide range of accessibility features, including Orca, espeak, and brltty. These features replace gnopegnicus and provide better text-to-speech support. The dasher on-screen keyboard also has been added in this release.

Note that the GNOME On-screen Keyboard (GOK) program that is used Oracle Solaris 10 is no longer available. The new dasher application can be used as a replacement for some users.

- **Command Assistant** – Locates command-line information in Oracle Solaris managed content, for example books and man pages. To add Command Assistant to the desktop panel, use the Add to Panel → Command Assistant dialog box.
- **Graphical login manager** – Oracle Solaris 10 uses the Common Desktop Environment (CDE) and dtlogin as the default login GUI. The GNOME graphical desktop manager (GDM) is also available in Oracle Solaris 10. In this release, GDM is the only graphical login option.

The GDM configuration process has also changed considerably in Oracle Solaris 11. To learn more, consult the `gdm` and `console-kit-daemon` man pages. ConsoleKit configuration features are now used to manage multi-seat environments. To troubleshoot transition issues, see “GNOME Desktop Manager Issues” on page 134.

- **Multimedia support:**

- **Audio devices** – The new 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.

- **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.
- **Additional media tools** – The Rhythmbox media player, Cheese photo/video tool, the Ekiga video conference tool, and the Brasero CD/DVD burner are included in this release.
- **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.
- **Network configuration management** – The NWAM feature is used to manage automatic network configuration from the desktop. This tool functions similarly to the `netcfg` and `netadm` command-line tools. See [Part I, “Network Auto-Magic,” in *Oracle Solaris Administration: Network Interfaces and Network Virtualization*](#).
- **Package Manager and Update Manager** – Are the graphical versions of the IPS command-line tools. Package Manager and Update Manager can be used to manage and update software packages from the desktop. See [Chapter 2, “IPS Graphical User Interfaces,” in *Adding and Updating Oracle Solaris 11 Software Packages*](#) for instructions on using these tools.
- **Print management** – CUPS is the default print service in Oracle Solaris 11, replacing the LP print service. Solaris Print Manager is no longer available. CUPS has a print manager that can be started from the desktop by choosing System → Administration → Print Manager.
- **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 now performed by the Hardware Abstraction Layer (HAL) through the `rmvolmgr` and `gvfs-hal-volume-monitor` commands, which are HAL-aware. See [`rmvolmgr\(1M\)`](#).

- **Seahorse** – GnuPG is now supported. The Seahorse application manages encryption keys and passwords in the `gnome-keyring`. Seahorse also replaces the `gnome-keyring-manager` for managing SSH and GnuPG keys.

- **Time Slider** – Manages ZFS snapshots. The tool can be used to regularly back up data by taking timed ZFS snapshots.
- **Solaris Trusted Extensions (GNOME) desktop** – This feature is now only supported in the Oracle Solaris Desktop (GNOME 2.30). In Oracle Solaris 10, Trusted Extensions is supported in both CDE and the GNOME Desktop. In Solaris 8 and earlier releases, Trusted Solaris support is limited to CDE.

This version of the Trusted Extensions desktop includes significant changes that improve usability, robustness, and functionality. In addition, zones and RBAC improvements have been made. For example, the `txzonemgr` GUI has significantly improved. This tool can now be used to manage most aspects of Trusted Extensions. If you are currently using Trusted CDE, you will need to migrate to the version of Trusted Extensions that is currently supported.

- **Virtual console terminals** – You can now switch between an X session and a virtual console terminal. This service is enabled by default. To switch between sessions, use the `Alt + Ctrl + F#` hotkey combination. For example, to switch to `vt2`, press `Alt + Ctrl + F2`. Also, you can create graphical VT sessions and then switch between those sessions by using the User Switcher panel applet. To add the applet to the desktop, right click the panel, then select the `Add to Panel...` option. To switch to a new or different graphical login session, click the applet, then select `Switch User`.
- **Web browser and Email** – Oracle Solaris 11 includes the new Firefox and Thunderbird applications.

Desktop Features That Have Been Removed

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

- **Common Desktop Environment (CDE)** – CDE is replaced by the Oracle Solaris Desktop (GNOME 2.30).
- **ESound** – Migrate to the GStreamer programs, such as `gst-launch`.
- **gnome-keyring-manager** – Seahorse replaces this feature.
- **GNOME On-screen Keyboard (GOK) program** – The `dasher` application can be used as a replacement in some instances.
- **GNOME System tools** (Introduced in an earlier Oracle Solaris 11 release):
 - `network-admin` – NWAM replaces this feature.
 - `services-admin` – Use the `/usr/bin/vp svcs` command.
 - `shares-admin` – Use the `/usr/bin/vp sharemgr` command.
 - `time-admin` – Use the `/usr/bin/vp time` command.
 - `users-admin` (GNOME Users and Groups tool) – No replacement is currently available. See [“Tools for Managing User Accounts, Groups, and Roles” on page 123](#).

The GNOME System tools are not available in Oracle Solaris 10.

- Solaris Management Console – This tool and its equivalent command-line tools are no longer available. See “[Tools for Managing User Accounts, Groups, and Roles](#)” on page 123.
- Solaris Print Manager – This tool is replaced by CUPS Print Manager in Oracle Solaris 11. See “[Printer Configuration and Management Changes](#)” on page 103.
- Xsun family of servers on SPARC – The Xorg family of servers is still supported. See “[Xorg Family of Servers](#)” on page 133.

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>
<code>/usr/openwin/bin/Xnest</code>	<code>/usr/bin/Xephyr</code>
<code>/usr/openwin/bin/Xvfb</code>	<code>/usr/bin/Xvfb</code>

X Server Key Maps

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

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

- 1 To update custom hot key configurations or enable the legacy mappings from the desktop, open the Keyboard panel from the System → Preferences menu.
- 2 Select the Layouts tab, then click the Options... button to open the Keyboard Layout Options dialog box.
- 3 Select the Maintain key compatibility with old Solaris keycodes option, then select the Sun Key Compatibility checkbox.

Troubleshooting Desktop Transition Issues

Refer to the following troubleshooting information when transitioning to the Oracle Solaris Desktop (GNOME 2.30).

Installing the Oracle Solaris Desktop Software Package After an Installation

The Oracle Solaris 11 text installer does not include the primary software package that includes the GNOME 2.30 desktop. If you use this installation method, you will need to install the `solaris-desktop` package afterwards. For information about using the `pkg install` command to add packages after a text installation, see [“Adding Software After Text Installation” in *Installing Oracle Solaris 11 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 there is not an exact one-to-one mapping between CDE and GDM login features. Some CDE login configuration choices are not available in the GDM login, and there are some GDM login configuration choices that are not available in the CDE login. For example, the GDM login screen does not offer a chooser screen by default.

Another example is the X Display Manager Control Protocol (XDMCP) feature, which is configured and enabled differently in Oracle Solaris 11 than Oracle Solaris 10. The new GDM provides the ability to run an XDMCP server, but this feature is disabled by default. You can enable the feature by modifying the GDM configuration file.

Another requirement of XDMCP is that X11 allow TCP/IP connections, which is also disabled by default. Refer to the `Xserver(1)` man page for instructions on how to enable this feature. See also the `gdm(1)` man page, the `yelp-tools` manual, and the online help.

- **Support for Oracle Solaris 10 GDM themes in Oracle Solaris 11** – In Oracle Solaris 10, GDM is shipped as a non-default login program, which includes a GUI configuration tool. In Oracle Solaris 11, GDM does *not* have this GUI configuration tool. Also, the GDM *themes* that work with GDM in Oracle Solaris 10 are not supported in this release. You can change the appearance of the new GDM login GUI by modifying the `/usr/share/gdm/gdm-greeter-login-window.ui` file, as desired.

Transitioning From Previous Oracle Solaris 11 Releases to Oracle Solaris 11

This appendix summarizes differences between the Oracle Solaris 11 Express and Oracle Solaris 11 releases.

For information about upgrading your Oracle Solaris 11 Express or Early Adopter system, see [Oracle Solaris 11 Release Notes](#).

Solaris 11 Express to Solaris 11 Transition Issues

Review the following sections to determine issues that might impact your migration to the Oracle Solaris 11 release.

Network Configuration Changes

In Oracle Solaris 11 Express, the `svc:/network/physical:default` SMF service was used for manual network configuration, and the `svc:/network/physical:nwam` SMF service was used for automatic network configuration. In this release, the `svc:/network/physical:nwam` SMF service is no longer used.

In Oracle Solaris 11 Express, the NWAM feature was introduced. The following aspects of this feature have changed:

- The switch between network configuration modes now happens at the profile level, not at the service instance level, as in previous releases.
- There is now only one network SMF service for both configuration modes, `svc:/network/physical:default`. The `svc:/network/physical:nwam` SMF service is deprecated. However, this service is still listed in the output of the `svcs -a` command on an Oracle Solaris 11 system:

```
# svcs -a | grep phys
disabled      Mar_11      svc:/network/physical:nwam
online        Mar_11      svc:/network/physical:default
```

- When using automatic network configuration, the `nwamd` daemon is enabled on the system.

```
# ps -ef | grep nwam
netadm      9438      1        0 05:23:20 ?                0:00 /lib/inet/nwamd
```

- The `nwamcfg` and `nwamadm` commands have been renamed `netcfg` and `netadm`, respectively. These commands can be run by root or by a user who has the appropriate privileges. See [netcfg\(1M\)](#) and [netadm\(1M\)](#).

During an *upgrade* from Oracle Solaris 11 Express, the network is configured as follows:

- If you were using automatic network configuration (NWAM) prior to the upgrade, after the upgrade, the `svc:network/physical:default` SMF service is now enabled, and the Automatic NCP is activated. If a user-defined NCP was active on the system prior to the upgrade, that NCP remains active. Note that the `svc:/network/physical:nwam` SMF service is deprecated in Oracle Solaris 11.
- If the network was configured by using traditional (manual) networking tools prior to the upgrade, after the upgrade, the `svc:network/physical:default` SMF service remains enabled, and the DefaultFixed NCP is activated.

Naming Service Configuration Changes

When upgrading from Oracle Solaris 11 Express, upon reboot, the system runs the new `svc:/system/name-service/upgrade` SMF service. This service is run by the `svc.startd` daemon after an early manifest import, but before most other services are started. The service searches for legacy naming service configuration files that have not been previously imported, then imports these services into the SMF repository. For the most part, the `nscfg` command detects configuration files that have been modified. The upgrade script detects configuration files that have not been imported and re-imports them, which resets the SMF configuration for the specified service. After the migration, all of the legacy configuration files, such as `resolv.conf`, `nsswitch.conf`, `/var/yp/*`, and `/var/ldap/`, are regenerated from the SMF data, when an appropriate service is started or refreshed. Note that legacy files are still used by these services. See [“Importing Network Configuration” on page 84](#) and [nscfg\(1M\)](#).

Printing Setup Changes

If you were running Oracle Solaris 11 Express prior to upgrading your system to Oracle Solaris 11, CUPS is already the default print service. There is no need to set up your printing environment to work with CUPS after the upgrade. However, if the LP print service was enabled, and you configured your printers by using `lp` print commands, those printers will need to be reconfigured by using CUPS.

▼ How to Set Up Your Printing Environment After Upgrading From Oracle Solaris 11 Express

- Prior to upgrading your system, determine which print service is enabled.

```
$ /usr/sbin/print-service -q
```

- If the command output displays "cups", then CUPS is already running on your system. There is no need to do anything further before upgrading your system.
- If the command output displays "lp", do the following:
 - a. Become the root role.
 - b. Back up the `/etc/printers.conf` file.
 - c. After the upgrade, reconfigure any existing printers by using one of the supported CUPS interfaces.

For instructions, see “[Setting Up and Administering Printers by Using CUPS Command-Line Utilities](#)” in *Oracle Solaris Administration: Common Tasks*.

Device Driver Customization Changes

If your device driver modifications are discarded from the `/kernel/drv` directory after upgrading to Oracle Solaris 11 from the Oracle Solaris 11 Express release, you need to move them to the `/etc/driver/drv` directory. For more information, see “[Device Identity and Configuration Changes](#)” on page 43.

Root File System Changes

During an Oracle Solaris 11 installation, a separate `/var` file system is created automatically for a global zone and a non-global zone. If a system is upgraded from Oracle Solaris 11 Express and `/var` was a directory before the upgrade, then `/var` remains a directory. For more information, see “[Oracle Solaris 11 File System Changes](#)” on page 55.

File System Sharing Changes

In the Oracle Solaris 11 Express release, you could use the `sharemgr` interface to create file system shares. This command is no longer available. For new file system sharing syntax, see “[ZFS File System Sharing Changes](#)” on page 59.

Default Path Changes

In early Oracle Solaris 11 releases, the default path included `/usr/gnu/bin` before `/usr/bin`. In Oracle Solaris, the default path is `/usr/bin`.