

D82701GC10

Edition 1.0

December 2013

D85048

**ORACLE®**

# **Oracle Solaris 11 System Administration for Experienced UNIX/Linux Administrators**

**Student Guide • Volume I**

**Authors**

Uma Sannasi  
J S Raghavendra

**Technical Contributors  
and Reviewers**

Geetha Nazare  
Shripad Patki  
John Hathaway  
Harry Burks  
Rosemary Martinak  
Todd Lowry  
Rajesh Rajasekharan  
Gary Riseborough  
David S Maxwell  
Glynn Foster  
Alex Barclay  
Deepak Dhanukodi  
Yousuf Mohammed  
Niveditha Sananth  
Sharvani Nagamalli  
Pranamy Jain  
Sreejith Mohan  
Manish Pawar

**Editors**

Daniel Milne  
Richard Wallis  
Smita Kommini

**Graphic Designers**

Prakash Damodaran  
Seema Bopaiah

**Publishers**

Syed Ali  
Srividya Rameshkumar

**Copyright © 2013, Oracle and/or its affiliates. All rights reserved.**

**Disclaimer**

This document contains proprietary information and is protected by copyright and other intellectual property laws. You may copy and print this document solely for your own use in an Oracle training course. The document may not be modified or altered in any way. Except where your use constitutes "fair use" under copyright law, you may not use, share, download, upload, copy, print, display, perform, reproduce, publish, license, post, transmit, or distribute this document in whole or in part without the express authorization of Oracle.

The information contained in this document is subject to change without notice. If you find any problems in the document, please report them in writing to: Oracle University, 500 Oracle Parkway, Redwood Shores, California 94065 USA. This document is not warranted to be error-free.

**Restricted Rights Notice**

If this documentation is delivered to the United States Government or anyone using the documentation on behalf of the United States Government, the following notice is applicable:

**U.S. GOVERNMENT RIGHTS**

The U.S. Government's rights to use, modify, reproduce, release, perform, display, or disclose these training materials are restricted by the terms of the applicable Oracle license agreement and/or the applicable U.S. Government contract.

**Trademark Notice**

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

# Contents

## Preface

### 1 Course Introduction

- Overview 1-2
- Course Goals 1-3
- Skills Gained 1-4
- Course Agenda: Day 1 1-5
- Course Agenda: Day 2 1-6
- Course Agenda: Day 3 1-7
- Course Agenda: Day 4 1-8
- Course Agenda: Day 5 1-9
- How Prepared Are You? 1-10
- Introductions 1-11
- Your Learning Center 1-12
- Your Lab Environment 1-13

### 2 Administering System Software by Using IPS

- Objectives 2-3
- Lesson Agenda 2-4
- Oracle Solaris: The Operating System 2-5
- Supported Platforms 2-6
- Software Management Prior to Oracle Solaris 11 2-7
- Software Management in Oracle Solaris 11 2-8
- Overview of IPS 2-9
- Publishers, Packages, and Repositories 2-10
- FMRI 2-11
- Repository Origins and Mirrors 2-12
- Images and BEs 2-13
- IPS Commands 2-14
- Quiz 2-15
- Lesson Agenda 2-16
- Why Do You Need a Local Repository? 2-17
- What Is a Local Repository? 2-18
- Local Repository Configuration Options 2-19
- Configuring a Local IPS Repository by Using SMF 2-20

Quiz	2-23
Lesson Agenda	2-24
Configuring Client Access to the Local IPS Server	2-25
Verifying Prerequisites Setup	2-26
Setting the Local IPS Publisher	2-27
Testing Client Access to the Local IPS Server	2-28
Lesson Agenda	2-29
IPS Interfaces	2-30
Package Management: CLI	2-31
Package Management: Package Manager	2-32
Lesson Agenda	2-33
Software Updates	2-34
Software Update Process	2-35
Update Interfaces	2-36
Updating Your Local Repository	2-37
Updating the OS by Using the CLI	2-38
Updating the OS by Using the GUI	2-40
Quiz	2-41
Lesson Agenda	2-42
Upgrading Oracle Solaris 11 to Oracle Solaris 11.1 OS	2-43
Upgrading the OS by Using the Oracle Solaris Support Repository	2-44
Upgrading the OS by Using the Oracle Solaris Release Repository	2-49
Quiz	2-51
Lesson Agenda	2-52
Overview of Boot Environments	2-53
Boot Environment Creation	2-54
BE Management Utilities	2-55
Administering Boot Environments	2-56
Listing the BEs on the System	2-57
Creating a New BE	2-58
Renaming an Existing Inactive BE	2-59
Destroying an Existing Inactive BE	2-60
Activating an Existing Inactive BE	2-61
Verifying the New BE	2-62
Mounting an Inactive BE	2-63
Unmounting an Inactive BE	2-64
Installing a Package on an Inactive Mounted BE	2-65
Uninstalling a Package on an Inactive Mounted BE	2-66
Creating a Backup of a BE	2-67
Creating a BE From an Existing Backup	2-68
Managing BEs with Package Manager	2-69

Quiz 2-70  
Summary 2-71

### **3 Administering Services by Using SMF**

Objectives 3-3  
Lesson Agenda 3-4  
Managing Services in Older UNIX OSs 3-5  
Managing Services Since Oracle Solaris 10 3-6  
Overview of SMF 3-7  
SMF Concepts 3-8  
SMF Service 3-9  
Service Identifier 3-10  
Service States 3-11  
SMF Components 3-12  
SMF Profile 3-13  
When Are SMF Profiles Applied? 3-14  
SMF Profile: Example 3-15  
Service Configuration Repository 3-16  
SMF Master Restarter Daemon (svc.startd) 3-17  
Milestone 3-18  
SMF Manifests 3-19  
SMF Manifest: Example 3-20  
SMF Repository Backups 3-21  
SMF Repository Snapshots 3-22  
Quiz 3-23  
Lesson Agenda 3-24  
Administering SMF Services 3-25  
Listing Services Information 3-26  
Displaying the Status of a Service Instance 3-27  
Displaying the Service Dependents and Dependencies 3-28  
Disabling a Service 3-29  
Enabling a Service 3-30  
Refreshing and Restarting a Service 3-31  
Managing SMF Services Properties 3-32  
Modifying inetd Service Properties 3-33  
Managing SMF Services by Using the GUI 3-34  
Quiz 3-35  
Lesson Agenda 3-36  
Configuring SMF Services 3-37  
Creating a Service 3-38  
Creating a Service: Example 3-39

Creating a Service by Using svcbundle	3-43
Modifying a Service's Manifest	3-44
Changing an Environment Variable of a Service	3-45
Changing a Property for an inetd-Controlled Service	3-46
Creating and Applying an SMF Profile	3-47
Changing Services and Their Configurations by Using the net services Command	3-48
Setting Up Service State Transition Notifications	3-49
Managing Notifications	3-51
Lesson Agenda	3-52
Least Privilege and SMF	3-53
Service Privileges	3-54
SMF Rights Profile	3-55
Authorizations and Rights	3-56
Service-Specific Property Groups	3-57
Quiz	3-58
Lesson Agenda	3-59
Troubleshooting SMF Services	3-60
Debugging a Service That is Not Starting	3-61
Restoring a Service in the Maintenance State	3-63
Restoring a Service in the Maintenance State: Example	3-64
Reverting to an SMF Snapshot	3-65
Reverting to an SMF Snapshot: Example	3-66
Repairing a Corrupt Repository	3-67
Repairing a Corrupt Repository: Example	3-71
Debugging Services During a System Boot	3-73
Addressing system/filesystem/local:default Service Failures During Boot	3-74
Summary	3-76

#### **4 Administering ZFS**

Objectives	4-3
Lesson Agenda	4-4
Overview of ZFS	4-5
Transactional File System	4-6
Scalability	4-7
Pooled Storage	4-8
Dynamic Striping in a Storage Pool	4-10
Data Integrity	4-11
Mirrored Storage Pool Configuration (RAID-1)	4-12
Parity Storage Pool Configuration (RAID-Z)	4-13
ZFS File System	4-14

Snapshots	4-15
ZFS Clones	4-16
Quiz	4-17
Lesson Agenda	4-18
Administering ZFS Storage Pools	4-19
Creating ZFS Storage Pools	4-20
Determining Local Storage Disk Availability	4-21
Default Mount Point for Storage Pools	4-22
Creating a Basic ZFS Storage Pool	4-23
Creating a Mirrored Storage Pool	4-24
Creating a RAID-Z Storage Pool	4-25
Creating a ZFS Storage Pool with Log Devices	4-26
Creating a ZFS Storage Pool with Cache Devices	4-27
Displaying ZFS Storage Pool Information	4-28
Destroying ZFS Storage Pools	4-29
Managing ZFS Storage Pool Properties	4-30
Quiz	4-31
Lesson Agenda	4-32
Managing Devices in ZFS Storage Pools	4-33
Adding Devices to a Storage Pool	4-34
Attaching Devices to a Storage Pool	4-35
Detaching Devices from a Storage Pool	4-36
Taking Devices Offline in a Storage Pool	4-37
Bringing Devices Online in a Storage Pool	4-38
Replacing Devices in a Storage Pool	4-39
Designating Hot Spares in a Storage Pool	4-40
Creating Hot Spares in a Storage Pool	4-41
Adding Hot Spares to a Storage Pool	4-42
Replacing a Faulted Device With a Hot Spare	4-43
Removing Hot Spares in a Storage Pool	4-44
Quiz	4-45
Lesson Agenda	4-46
Administering ZFS File Systems	4-47
Creating a ZFS File System	4-48
Renaming a ZFS File System	4-49
Destroying a ZFS File System	4-50
Mounting ZFS File Systems	4-51
Unmounting a ZFS File System	4-53
ZFS File System Properties	4-54
ZFS File System Native Properties	4-55
Setting Quotas for ZFS File Systems	4-57

- Setting Quotas for Users 4-58
- Displaying User Space Usage 4-59
- Removing User Quotas 4-60
- Quiz 4-61
- Lesson Agenda 4-62
- Administering ZFS Snapshots and Clones 4-63
- Creating a ZFS Snapshot 4-64
- Displaying a ZFS Snapshot 4-65
- Viewing Snapshot Space Accounting 4-66
- Destroying a ZFS Snapshot 4-67
- Renaming a ZFS Snapshot 4-68
- Rolling Back a ZFS Snapshot 4-69
- Identifying ZFS Snapshot Differences 4-70
- Sending ZFS Snapshot Data 4-71
- Receiving ZFS Snapshot Data 4-72
- Replicating ZFS Snapshot Data Remotely 4-73
- Creating a ZFS Clone 4-74
- Destroying a ZFS Clone 4-75
- Replacing a ZFS File System with a ZFS Clone 4-76
- Quiz 4-78
- Lesson Agenda 4-79
- Securing ZFS File Systems 4-80
- Delegated Administration 4-81
- Delegating ZFS Permissions 4-82
- Disabling ZFS Delegated Permissions 4-84
- Removing ZFS Delegated Permissions 4-85
- Data Encryption 4-86
- Encrypting a ZFS Storage Pool and ZFS File System 4-87
- Summary 4-88

## **5 Configuring the Network**

- Objectives 5-3
- Agenda 5-4
- Networking in Oracle Solaris 11 5-5
- Network Stack in Oracle Solaris 11 5-6
- Agenda 5-7
- Prerequisites for Configuring a Network 5-8
- Profile-Based Network Configuration 5-9
- NCPs 5-10
- Fixed NCP 5-11
- Reactive NCP 5-12



Comparison Between Fixed and Reactive NCPs	5-13
Network Configuration and Administration Commands	5-14
netcfg Command	5-15
netadm Command	5-16
Commands to Configure Profiles	5-17
Configuring and Administering Datalink and Network Interfaces	5-18
dladm Command	5-19
dladm Types/Classes	5-20
Administering Datalinks with dladm Commands	5-21
ipadm Command	5-22
Administering Network Interfaces with the ipadm Command	5-23
Quiz	5-25
Agenda	5-26
Network Virtualization	5-27
Components of a Virtual Network	5-28
Network Virtualization in Zones	5-29
Network Virtualization in LDOMs	5-30
Configuring and Administering Virtual Networks	5-31
Creating a Virtual Network	5-32
Administering Virtual Networks	5-33
Migrating a VNIC	5-34
Quiz	5-35
Private Virtual Network	5-36
Features of a Private Virtual Network	5-37
Creating a Private Virtual Network	5-38
Establishing Communication Between Networks	5-39
Quiz	5-40
Agenda	5-41
High Availability	5-42
Overview of IPMP	5-43
IPMP Components	5-45
Types of IPMP Configurations	5-46
Failure and Repair Detection in IPMP	5-47
Configuring and Administering an IPMP Group	5-48
Creating an IPMP Group	5-49
Commands to Administer an IPMP Group	5-50
Quiz	5-51
Overview of Link Aggregation	5-52
Link Aggregation Types	5-53
Trunk Aggregation	5-54
LACP for Trunk Aggregation	5-55

- Policies in Trunk Aggregation 5-56
- Back-to-Back Configuration in Trunk Aggregation 5-57
- Quiz 5-58
- DLMP Aggregation 5-59
- DLMP at Work 5-60
- Comparison Between Trunk Aggregation and DLMP Aggregation 5-61
- Preconfiguration Requirements for Link Aggregation 5-62
- Creating a Link Aggregation 5-64
- Commands to Administer Link Aggregations 5-65
- Quiz 5-66
- Agenda 5-67
- Overview of Network Resource Management 5-68
- Datalink Properties 5-69
- Flows 5-70
- Commands for Network Resource Management 5-71
- dladm for Allocating Datalink Properties 5-72
- flowadm for Managing Flows 5-73
- Quiz 5-74
- Managing Network Resources 5-75
- Configuring Virtual Speed 5-76
- Configuring CPU Pools for Datalinks 5-77
- Allocating CPUs to Datalinks 5-78
- Agenda 5-80
- Need for Network Security 5-81
- Overview of Link Protection 5-82
- Link Protection Types 5-83
- Configuring and Administering Link Protection 5-84
- Summary 5-86

## **6 Administering Oracle Solaris Zones**

- Objectives 6-3
- Lesson Agenda 6-4
- Oracle Solaris Zones: Overview 6-5
- Types of Zones 6-6
- Zone States 6-7
- Zone Commands 6-9
- Quiz 6-10
- Lesson Agenda 6-11
- Zone Configuration Process 6-12
- Creating a ZFS File System for Zones 6-14
- Configuring a Zone 6-15

Displaying a Zone Configuration	6-17
Verifying That a Zone Is in the configured State	6-19
Gathering Information for the System Configuration Profile	6-20
Creating the SC Profile	6-21
Installing the Zone	6-22
Booting the Zone	6-23
Quiz	6-24
Lesson Agenda	6-25
Network Connectivity in Zones	6-26
Virtual Network Configuration	6-27
Checking the Virtual Network Configuration in a Zone	6-28
Verifying That a Zone's Virtual Network Interface Connection Is Operational	6-29
Quiz	6-30
Lesson Agenda	6-31
Administering an Oracle Solaris Zone	6-32
Displaying Zone Configuration Information	6-33
Logging In and Logging Out of a Zone	6-35
Halting, Shutting, and Starting a Zone	6-36
Quiz	6-37
Lesson Agenda	6-38
Zone Resource Management	6-39
Resource Pools	6-40
How Resource Pools Work	6-41
Allocating a Resource Pool to a Zone	6-42
Enabling Services for Resource Pools	6-43
Configuring a Persistent Resource Pool	6-44
Displaying the Resource Pool Configuration File	6-45
Modifying the Resource Pool Configuration File	6-47
Displaying and Committing the Modified Resource Pool Configuration File	6-49
Displaying the Resource Pool Configuration That Is Currently in Use	6-52
Displaying All Active Resource Pools	6-53
Binding the Zone to a Persistent Resource Pool	6-55
Allocating the Pool to the Zone	6-56
Rebooting the Zone	6-57
Confirming the Availability of the Resource Pool	6-58
Removing the Resource Pool Configuration	6-60
Removing the Resource Pool Configuration from the Zone	6-61
Rebooting the Zone	6-62
Checking the Resource Pool Configuration for the Zone	6-63
Deleting the Resource Pool	6-65
Resource Capping	6-66

Allocating Physical Memory Resources with Resource Capping	6-67
Quiz	6-68
Lesson Agenda	6-69
Securing Oracle Solaris Zones	6-70
Delegated Administration	6-71
Zone Link Protection	6-72
Exclusive IP	6-73
Immutable Zones	6-74
Cryptographic Services in Zones	6-75
Privileges	6-76
Users and Rights Profiles in Oracle Solaris Zones	6-78
Summary	6-79

## **7 Administering Privileges and RBAC**

Objectives	7-3
Lesson Agenda	7-4
Assignment of User Privileges and Roles	7-5
Process Rights Management	7-6
Areas of Privilege	7-7
Sets of Privileges	7-8
Administering Privileges	7-10
Determining the Privileges Available to the Shell	7-11
Determining the Privileges on a Process	7-13
Displaying the Description of a Privilege	7-14
Determining the Privileges That Are Directly Assigned to You	7-15
Determining the Privileged Commands That a User Can Use	7-16
Assigning Privileges to a User or Role	7-17
Limiting the Privileges of a User or Role	7-18
Debugging Privilege Failure	7-19
Debugging Privilege Use in a Profile Shell	7-20
Debugging Privilege Use in a Regular Shell	7-21
Quiz	7-22
Lesson Agenda	7-23
Role-Based Access Control (RBAC)	7-24
Roles	7-25
Rights Profile	7-26
Authorizations	7-27
Privileges	7-28
Security Attributes	7-29
Key RBAC Files	7-30
Key RBAC Files: user_attr	7-31

- Key RBAC Files: auth\_attr 7-32
- Key RBAC Files: exec\_attr 7-34
- Key RBAC Files: prof\_attr 7-36
- Relationships Among the RBAC Files 7-38
- Profile Shells 7-40
- Quiz 7-41
- Configuring RBAC 7-42
- Creating a Role 7-43
- Creating a Rights Profile 7-45
- Cloning a Rights Profile 7-46
- Modifying a Rights Profile 7-47
- Assigning a Rights Profile to a Role 7-49
- Assigning a Role to a User 7-50
- Assuming a Role 7-52
- Restricting an Administrator to Explicitly Assigned Rights 7-53
- Assigning the Rights Profile to a User 7-54
- Delegating an Authorization to a User 7-55
- Assigning Authorization to a Role 7-57
- Rights Profiles 7-58
- Modifying a System-Wide RBAC Policy 7-59
- Quiz 7-61
- Summary 7-62

## **8 Installing the Oracle Solaris 11 Operating System**

- Objectives 8-3
- Lesson Agenda 8-4
- Oracle Solaris Installation 8-5
- Preparing for the Installation 8-6
- Reviewing the Release Notes 8-7
- Selecting the Installation Option 8-8
- Identifying System Requirements 8-9
- Downloading Images 8-10
- Agenda 8-11
- Installing Oracle Solaris 11 by Using the Live Media Installer 8-12
- Introducing the Live Media Desktop 8-14
- Initiating the Installation with Live Media 8-15
- Welcome Screen 8-16
- Oracle Solaris 11 Live Media: Disk Discovery 8-17
- Selecting a Disk 8-18
- Setting the Time Zone, Date, and Time 8-19
- Providing User Information 8-20

Support Registration	8-21
Reviewing Installation Specifications	8-22
Monitoring the Installation	8-23
Verifying the Installation	8-24
Reviewing the Installation Log	8-25
Rebooting the System	8-28
Login Screen	8-29
Checking the Login Username	8-30
Checking the Login Password	8-31
Lesson Agenda	8-32
Installing Oracle Solaris 11 by Using the Text Installer	8-33
Verifying the Installation	8-34
Verifying Login Information	8-35
Verifying the System's Host Name	8-36
Displaying Basic System Information	8-37
Displaying the System's Release Information	8-38
Displaying Disk Configuration Information	8-39
Displaying Disk Configuration Information: Format Menu	8-40
Displaying Disk Configuration Information: Partition Table	8-41
Displaying the Installed Memory Size	8-42
Displaying Information About Network Services	8-43
Displaying Network Interface Information	8-44
Baseline System Information Commands: Summary	8-45
Quiz	8-46
Lesson Agenda	8-47
Automated Installer (AI): Overview	8-48
Automated Installer: Components	8-49
Automated Installer: Process	8-50
Automated Installer: Flowchart	8-51
Performing an AI Installation	8-52
Reviewing AI Installation Server Requirements	8-53
Verifying the Server Software Requirements	8-54
Verifying the Static IP Address	8-55
Verifying That DNS Is Operational	8-56
Enabling the DNS Multicast Service	8-57
Verifying That IPS Is Available Locally	8-58
Verifying That the DHCP Server Is Enabled	8-59
Configuring the AI Server	8-60
Installing the OS on the Client System	8-61
Identifying Client System Requirements	8-62
Identifying the Installation Files	8-63

Performing the Installation 8-64  
 Reviewing Client Installation Messages 8-65  
 Quiz 8-67  
 Summary 8-68

## **9 Monitoring System Resources**

Objectives 9-3  
 Agenda 9-4  
 Monitoring and Observability Tools 9-5  
 iostat Utility 9-6  
 kstat Utility 9-7  
 mpstat Utility 9-8  
 pgstat Utility 9-9  
 fsstat Utility 9-10  
 poolstat Utility 9-11  
 svcs Utility 9-12  
 netstat Utility 9-13  
 dlstat Utility 9-14  
 flowstat Utility 9-15  
 ipmpstat Utility 9-16  
 acctadm Utility 9-17  
 zonestat Utility 9-18  
 vmstat Utility 9-19  
 prstat Utility 9-20  
 truss Utility 9-21  
 ptree Utility 9-22  
 Quiz 9-23  
 Agenda 9-24  
 DTrace: Overview 9-25  
 DTrace: Capabilities 9-26  
 DTrace: Components 9-27  
 Probes 9-28  
 Providers 9-29  
 Consumers 9-30  
 D Language 9-31  
 DTrace Toolkit 9-32  
 DTrace Toolkit: Important Scripts 9-33  
 Before Using DTrace 9-34  
 Launching DTrace 9-35  
 DTrace: Example 9-36  
 Summary 9-37





---

# Preface

---



## **Profile**

### **Before You Begin This Course**

- ◆ You should have good knowledge of any of the following UNIX Operating Systems: AIX, Linux, HP-UX.

### **Prerequisites**

- ◆ Skilled AIX/Linux/HP-UX Administrators or Users

### **How This Course Is Organized**

#### **Oracle Solaris 11 System Administration for Experienced UNIX/Linux**

**Administrators** is an instructor-led course featuring lectures and hands-on exercises. Online demonstrations and written practice sessions reinforce the concepts and skills introduced.

## Related Courses/Publications

### Oracle Courses

Title	Part Number
Oracle Solaris 11 Zones Administration	D75929GC10
Oracle Solaris 11 Network Administration	D78415GC10
Oracle Solaris 11 Security Administration	D77033GC10
Oracle Solaris 11 Performance Management	D78767GC10

### Additional Publications

- System release bulletins
- Installation and user's guides
- *read.me* files
- International Oracle User's Group (IOUG) articles
- *Oracle Magazine*

## Typographic Conventions

The following table lists the typographical conventions that are used in text and code.

### Typographic Conventions in Text

Convention	Object or Term	Example
Uppercase	Commands, functions, column names, table names, PL/SQL objects, schemas	Use the <code>SELECT</code> command to view information stored in the <code>LAST_NAME</code> column of the <code>EMPLOYEES</code> table.
Lowercase, italic	Filenames, syntax variables, usernames, passwords	<b>where:</b> <i>role</i> is the name of the role to be created.
Initial cap	Trigger and button names	Assign a When-Validate-Item trigger to the ORD block. Select Cancel.
Italic	Books, names of courses and manuals, and emphasized words or phrases	For more information on the subject see <i>Oracle SQL Reference Manual</i>
Quotation marks	Lesson module titles referenced within a course	Do <i>not</i> save changes to the database. This subject is covered in Lesson 3, “Working with Objects.”

## Typographic Conventions (continued)

### Typographic Conventions in Code

Convention	Object or Term	Example
Uppercase	Commands, functions	<code>SELECT employee_id FROM employees;</code>
Lowercase, italic	Syntax variables	<code>CREATE ROLE <i>role</i>;</code>
Initial cap	Forms triggers	<code>Form module: ORD Trigger level: S_ITEM.QUANTITY item Trigger name: When-Validate-Item . . .</code>
Lowercase	Column names, table names, filenames, PL/SQL objects	<code>. . . OG_ACTIVATE_LAYER (OG_GET_LAYER ('prod_pie_layer')) . . .  SELECT last_name FROM employees;</code>
Bold	Text that must be entered by a user	<code>CREATE USER scott IDENTIFIED BY tiger;</code>

# 1

## Course Introduction

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Overview

- Course goals
- Course agenda
- Introductions
- Your learning center
- Your lab environment

## Note

- The class is from 9:00 AM to 5:00 PM each day.
- There will be several short breaks throughout the day with an hour's break for lunch.

The Oracle logo, consisting of the word "ORACLE" in white, uppercase, sans-serif font, centered on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Welcome to the Oracle Solaris 11 System Administration for Experienced UNIX/Linux Administrators course. This course is aimed at skilled AIX/Linux/HP-UX administrators who need to make informed decisions as they adopt, deploy, and administer Oracle Solaris 11 in their data centers.

This is a system administration course that uses extensive hands-on exercises to highlight some of the key features and technologies of Oracle Solaris 11, such as IPS, SMF, ZFS, Oracle Solaris Zones, and network virtualization.

To begin, we would like to take about an hour to give you an overview of the course, starting with the course goals, followed by the agenda, and introductions. We conclude with a few details about the classroom setting. You will then receive an orientation of the lab environment.



## Course Goals

The purpose of this course is to:

- Impart application-level skills and knowledge to successfully administer Oracle Solaris 11
- Equip you with the knowledge and skills required to configure and administer some of the key features and technologies of Oracle Solaris 11, IPS, SMF, ZFS, network virtualization, and Oracle Solaris Zones
- Introduce you to the various monitoring and observability tools for monitoring system resources and diagnosing issues

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Skills Gained

After completing this course, you will be able to:

- Administer system software by using IPS
- Administer services by using SMF
- Administer ZFS
- Configure the network
- Configure Oracle Solaris Zones
- Administer privileges and RBAC
- Install the Oracle Solaris 11 operating system
- Monitor system resources

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Course Agenda: Day 1

- Lesson 1: Course Introduction
- Lesson 2: Administering System Software by Using IPS
  - Describe the features and capabilities of Image Packaging System (IPS)
  - Configure a local IPS repository
  - Configure client access to the local IPS server
  - Administer software packages
  - Update the Oracle Solaris 11 operating system
  - Upgrade Oracle Solaris 11 to Oracle Solaris 11.1 operating system
  - Administer boot environments

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Course Agenda: Day 2

- Lesson 3: Administering Services by Using SMF
  - Describe the features and capabilities of SMF
  - Administer SMF services
  - Configure SMF services
  - Secure SMF services
  - Troubleshoot SMF services
- Lesson 4: Administering ZFS
  - Describe the merits of using ZFS in data management
  - Administer ZFS storage pools
  - Manage devices in ZFS storage pools

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Course Agenda: Day 3

- Lesson 4: Administering ZFS
  - Administer ZFS file systems
  - Administer ZFS snapshots and clones
  - Secure ZFS file systems
- Lesson 5: Configuring the Network
  - Describe the network stack in Oracle Solaris 11
  - Configure a network interface
  - Implement network virtualization
  - Configure network high availability
  - Implement resource management
  - Secure the network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Course Agenda: Day 4

- Lesson 6: Administering Oracle Solaris Zones
  - Explain the fundamentals of Oracle Solaris Zones
  - Configure Oracle Solaris Zones
  - Configure network connectivity in Oracle Solaris Zones
  - Administer Oracle Solaris Zones
  - Manage system resources in Oracle Solaris Zones
  - Secure Oracle Solaris Zones
- Lesson 7: Administering Privileges and RBAC
  - Administer process rights management
  - Configure Role-Based Access Control (RBAC)

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Course Agenda: Day 5

- Lesson 8: Installing the Oracle Solaris 11 Operating System
  - Perform pre-installation tasks
  - Install Oracle Solaris 11 on a single host by using the Live Media Installer
  - Install Oracle Solaris 11 on a single host by using the Text Installer
  - Install Oracle Solaris 11 on multiple hosts by using the Automated Installer
- Lesson 9: Monitoring System Resources
  - Identify the tools for monitoring system resources
  - Explain the role of DTrace in diagnosing system issues

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## How Prepared Are You?

A **Yes** for an answer to the following questions indicates you are prepared to take this course:

- Do you have system administration experience in any flavor of UNIX operating system (such as AIX, Linux, HP-UX)?
- Are you a Linux user?

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



# Introductions

Now that you have been introduced to the course format, introduce yourself to the other students and the instructor, addressing the following items:

- Name
- Company affiliation
- Title, function, and job responsibility
- Experience related to topics presented in this course
- Reasons for enrolling in this course
- Expectations from this course

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Your Learning Center

The instructor will acquaint you with the following details:

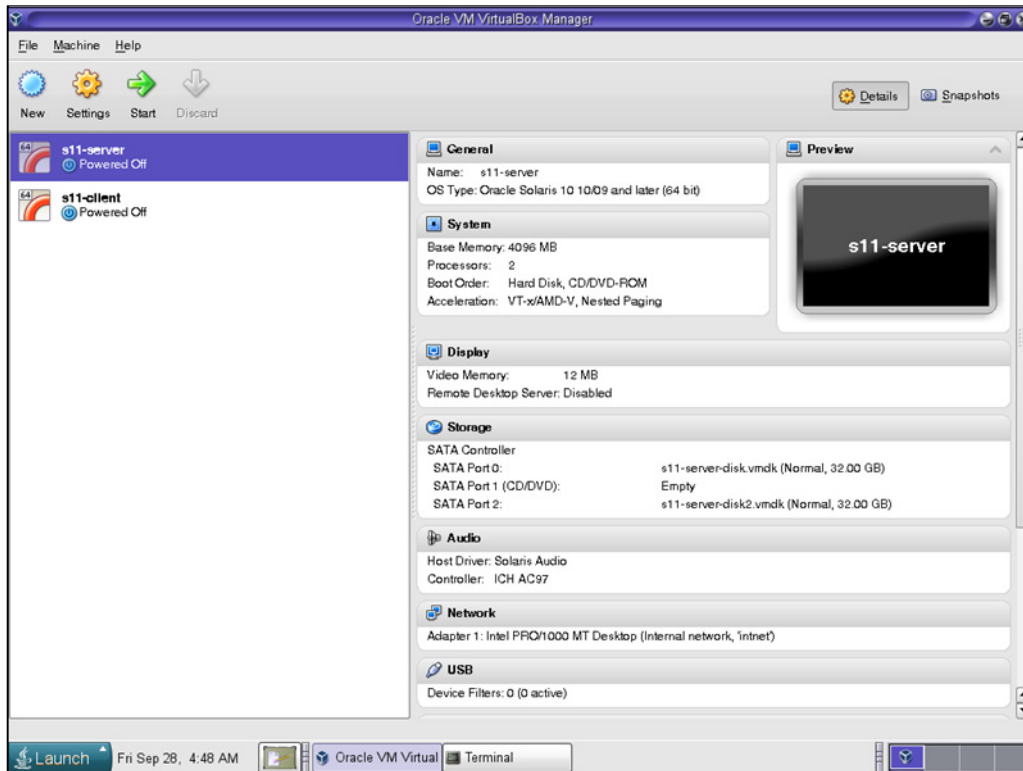
- Layout of the training facility
  - Restrooms
  - Break rooms and designated smoking areas
  - Cafeterias and restaurants in the area
- Emergency evacuation procedures
- Instructor contact information
- Cell phone usage
- Online course attendance confirmation form

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The instructor will acquaint you with the layout of the training facility, review the emergency evacuation procedures, provide you with contact information, review the use of cell phones in the classroom, and finally walk you through the Oracle University online course attendance confirmation form. Now that you have an idea of what you are going to be doing over the next three days, you can get started with an introduction to the lab environment.

# Your Lab Environment

**ORACLE**

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

As part of each lesson, you will be given the opportunity to practice in a lab environment. The lab environment we use in this course is based on the Oracle VM VirtualBox virtualization software, an example of which is shown on the slide. VirtualBox is a cross-platform virtualization application. It extends the capabilities of your existing computer so that it can run multiple operating systems inside multiple virtual machines at the same time.

Open your activity guide to “Practices for Lesson 1: Course Introduction.” Your instructor will walk you through the material, and you will have a chance to familiarize yourself with the lab environment, configuration, and setup.

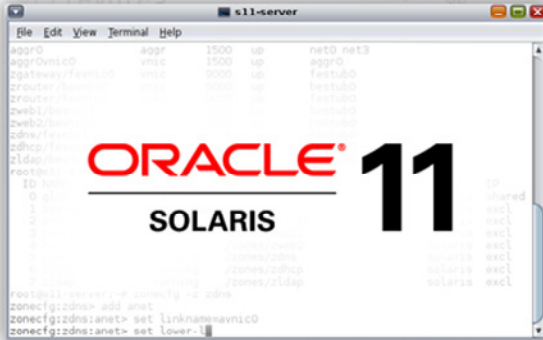


# 2

## Administering System Software by Using IPS

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## System Administration for Experienced UNIX/Linux Administrators



**Administering System  
Software by Using IPS**



**Administering Services  
by Using SMF**



**Administering ZFS**



**Configuring the Network**



**Administering Oracle Solaris  
Zones**



**Administering Privileges  
and RBAC**



**Installing the Oracle Solaris 11  
Operating System**



**Monitoring System Resources**

**ORACLE**

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Objectives

After completing this lesson, you should be able to:

- Describe the features and capabilities of Image Packaging System (IPS)
- Configure a local IPS repository
- Configure client access to the local IPS server
- Administer software packages
- Update the Oracle Solaris 11 operating system
- Upgrade Oracle Solaris 11 to Oracle Solaris 11.1 operating system
- Administer boot environments

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Lesson Agenda

- Describing the features and capabilities of IPS
- Configuring a local IPS repository
- Configuring client access to the local IPS server
- Administering software packages
- Updating the Oracle Solaris 11 operating system
- Upgrading Oracle Solaris 11 to Oracle Solaris 11.1 operating system
- Administering boot environments

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Oracle Solaris: The Operating System

- Oracle Solaris is an enterprise-grade UNIX-based operating system (OS) that delivers mission-critical cloud infrastructure with built-in virtualization, simplified software lifecycle management, data management, and advanced data protection.
- Oracle Solaris is co-engineered with Oracle's software and hardware to run Oracle's enterprise applications, scalable systems, high-performance interconnects, and optimized data center storage—all aimed at simplifying deployment and maintenance activities, and reducing ongoing operational costs.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Supported Platforms

Architecture	Systems	Virtualization	OS Virtualization
SPARC	M-Series	Dynamic Domains + Logical Domains	Oracle Solaris Zones
	T-Series	Oracle VM for SPARC formerly known as LDoms	
x86	X86 (64 bit processor)	Oracle VM for x86	
Plus other third-party virtualization offerings from vendors including VMware, Windows, and Red Hat			



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Solaris 11 is supported across a range of architectures and virtualization layers. The table in the slide summarizes the various platforms Oracle Solaris 11 can be installed on. Review the Oracle Solaris 11.1 [Release Notes](#) as you plan to install the OS on your system. The Release Notes describes the important installation, update, and runtime issues that you might need to consider before installing the OS. You will learn more about the specifics of installing the OS in “Lesson 8: Installing Oracle Solaris 11 Operating System.” This lesson discusses one of the key features of Oracle Solaris 11, Image Packaging System (IPS), which provides simplified software lifecycle management.

## Software Management Prior to Oracle Solaris 11

- In previous releases of the Oracle Solaris platform, administrators used:
  - System V Release 4 (SVR4) packaging to install software onto a system
  - Then a set of commands to install patches to update the system
- Software patching and upgrading involved risk and uncertainty because of associated activities, such as having to:
  - Reconcile the package dependencies manually
  - Identify software that is not currently installed but is needed
  - Download patches manually

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Software Management in Oracle Solaris 11

- Previously used processes for managing system updates and upgrades became more complex as Oracle Solaris evolved to include new technologies, such as:
  - Oracle Solaris Zones
  - ZFS
  - Service Management Facility (SMF)
- With IPS, introduced in Oracle Solaris 11, there is:
  - No more patching
  - Minimal system down time because of:
    - Integrated boot environment (BE) management
    - Automatic package dependency management

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Given the thousands of OS instances installed in some of today's large virtualized data centers, manual methods of tracking and installing patches can result in errors that could negatively impact application availability and security.

## Overview of IPS

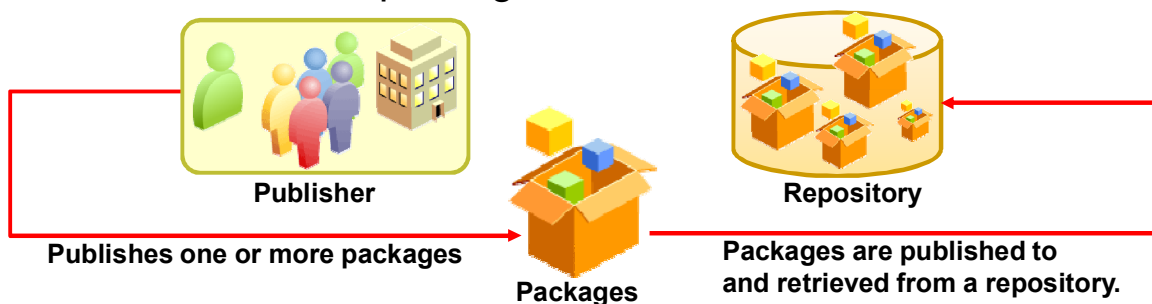
- IPS is a network-based system-wide software management solution.
- IPS provides a framework for complete software lifecycle management such as installation, upgrade, and removal of software packages.
- The following concepts are useful in understanding the IPS framework and its role in managing system software:
  - Publishers, packages, and repositories
  - Fault Management Resource Identifiers (FMRI)s
  - Repository origins and mirrors
  - Images and boot environments

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Publishers, Packages, and Repositories

- A publisher identifies a person or organization that publishes one or more packages.
- A package is a collection of directories, files, links, drivers, dependencies, groups, users, and license information in a defined format.
- A repository is a location specified by a Universal Resource Identifier (URI) where packages are published and from where packages are retrieved.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A **package archive** is a file that contains publisher information and one or more packages provided by that publisher.

# FMRI

- Each IPS package is uniquely described by an FMRI.
- For example,  
`pkg://solaris/diagnostic/wireshark@1.4.2,5.11-0.174:20110128T0635Z`

FMRI Segment	Description
pkg	Scheme
solaris	Publisher
diagnostic	Category
wireshark	Package name
1.4.2	Component version
5.11	Build version
0.174	Branch version
20110128T0635Z	Package time stamp



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The FMRI includes descriptive information about the package, such as the package name, version information, and date. The table describes the various segments involved in a sample FMRI.

**Note:** There can be an arbitrary number of “categories” in an FMRI. For example, `pkg://solaris/entire` does not have a category at all, while `pkg://solaris/x11/server/xorg/driver/xorg-video-nv` has four categories.

## Repository Origins and Mirrors

- An origin is a package repository that contains:
  - Package metadata (catalogs, manifests, and search indexes)
  - Package content (files)
- A mirror is a package repository that contains only package content.

The Oracle logo, consisting of the word "ORACLE" in white, uppercase, sans-serif font, centered on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

- An **origin** is a package repository that contains both package **metadata** (such as catalogs, manifests, and search indexes) and package **content** (files). If multiple origins are configured for a given publisher in an image, the IPS client attempts to choose the best origin from which to retrieve package data.
- A **mirror** is a package repository that contains only package content. IPS clients access the origin to obtain a publisher's catalog, even when the clients download package content from a mirror. If a mirror is configured for a publisher, the IPS client prefers the mirror for package content retrieval. If multiple mirrors are configured for a given publisher in an image, the IPS client attempts to choose the best mirror from which to retrieve package content. If all mirrors are unreachable, do not have the required content, or are slower, the IPS client retrieves the content from an origin.



## Images and BEs

- An image is a location where IPS packages can be installed.
- An image can be one of three types:
  - Full image
  - Zone image
  - User image
- A BE is a bootable instance of an image.
  - You can maintain multiple BEs on your system, and each BE can have different software versions installed.
  - When you boot your system, you have the option to boot into any of the BEs on the system.
  - A new BE is created automatically as a result of package operations or you can also explicitly create a new BE.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

An image can be one of three types:

- Full image: Is capable of providing a complete system
- Zones image: Is linked to a full image (the parent image) to ensure that the software is consistent with what is installed in the global zone
- User image: Contains only relocatable packages

## IPS Commands

Command	Description
pkg	Creates an image, installs packages to an image, and manages packages on an image
pkgdepend	Generates and resolves dependencies for packages
pkgmogrify	Transforms the raw manifest file generated with pkgdepend
pkgsend	Publishes packages from an image to an existing repository
pkgrecv	Downloads the contents of a package from a server
pkgrepo	Creates and manages a local repository's storage and properties
pkgdepotd	Runs the network repository or sets up mirror repositories



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table summarizes the IPS commands that can be used to manage IPS.

**Note:** pkgdepend, pkgmogrify, and pkgsend are used to create IPS packages, and not to manage IPS packages or repositories. In that sense, they are developer commands, and not essentially administration commands.

## Quiz

You require some additional Oracle Solaris 11.1 packages to be installed on your system. Where can you possibly get the packages from?

- a. SVR4 packaging
- b. IPS repositories
- c. BEs

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

## Lesson Agenda

- Describing the features and capabilities of IPS
- **Configuring a local IPS repository**
- Configuring client access to the local IPS server
- Administering software packages
- Updating the Oracle Solaris 11 operating system
- Upgrading Oracle Solaris 11 to Oracle Solaris 11.1 operating system
- Administering boot environments

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Why Do You Need a Local Repository?

- A local package repository is necessary when your network clients do not have access to Oracle's support repository.
- A few other reasons you might want to have a local copy of the IPS repository include:
  - Performance
  - Security
  - Replication

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A few other reasons you might want to have a local copy of the IPS repository include:

- **Performance:** Having a local package repository provides clients with access to packages at local network speeds.
- **Security:** You might not want your client systems to have access to the Internet.
- **Replication:** You want to ensure that an installation that you perform next year is exactly the same as the installation you perform today.

## What Is a Local Repository?

- IPS supports the ability to create local IPS repositories, such as local to your machine, your network, or your geographic area.
- Oracle's `support` repository provides a complete archive of software packages to allow you to set up a local IPS repository to which client systems can connect.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Local Repository Configuration Options

- When setting up a local IPS repository, there are two options for serving packages from the local repository to the clients:
  - From a local web server (using SMF)
  - From a local directory (file system–based)
- In either case, you need to configure each client to use the local repository that you create.

The Oracle logo, consisting of the word "ORACLE" in white, uppercase letters on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

This lesson describes creating the local IPS repository by using SMF. You will learn more about SMF in “Lesson 3: Administering Services by Using SMF.” In the context of setting up a local IPS repository in this lesson, you will be introduced to some of the SMF commands that are required for the repository configuration.

Configuring a file system–based repository is an alternative method of retrieving the repository from its default location on the Internet and serving the packages from a directory on your local file system.

# Configuring a Local IPS Repository by Using SMF

Perform the following tasks to configure a local IPS repository by using SMF:

1. Creating a ZFS dataset for the local IPS repository
2. Obtaining software packages from the Oracle Solaris download site
3. Making the contents of the repository available
4. Configuring the repository server service
5. Setting the local IPS publisher
6. Testing IPS on the local server

**Demo:** Click [here](#) to view the procedure.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Do the following to create a local IPS repository by using SMF:

**1. Creating a ZFS dataset for your local IPS repository**

```
# zfs create rpool/export/IPS
```

**Note:** You will learn more about ZFS in “Lesson 4: Administering ZFS.”

**2. Obtaining software packages from the Oracle Solaris download site**

Download the Oracle Solaris 11 repository image from the Oracle Solaris download site:  
<http://www.oracle.com/technetwork/server-storage/solaris11/downloads/index.html>.

The repository image provides you with a complete archive of software packages to allow you to set up a local network IPS repository that client systems can connect to.

The repository image is divided into two files:

- Download Part A SPARC, x86 (3.3 GB)
- Download Part B SPARC, x86 (3.1 GB)



Copy both files to the ZFS dataset that you created earlier, uncompress them, and concatenate them. You can then run the `ls` command on the ZFS dataset to see the concatenated ISO file.

```
# unzip sol-11_1-repo-full.iso-a.zip
# unzip sol-11_1-repo-full.iso-b.zip
# cat sol-11_1-repo-full.iso-a sol-11_1-repo-full.iso-b > sol-11_1-repo-full.iso
```

### 3. Making the repository content available

After the concatenated repository ISO image is in the ZFS dataset, you need to make the content of the dataset available to the depot server (`pkg.depotd`). The `lofiadm -a` command allows you to associate a file with a block device.

```
# lofiadm -a sol-11_1-repo-full.iso /dev/lofi/1
```

Next, mount the device by using the `mount -F hsfs` command. The `-F` option specifies the file system type on which to operate.

```
# mount -F hsfs /dev/lofi/1 /mnt
```

**Note:** Oracle Solaris 11 onwards, the `lofiadm` command is not required. The above set of commands can be simplified to:

```
# mount -F hsfs -o ro sol-11_1-repo-full.iso /mnt
```

Now, run the `rsync` program to copy the repository files to the ZFS dataset.

```
# rsync -aP /mnt/repo /export/IPS
```

Next, refresh the repository catalog by using the `pkgrepo refresh` command so that the package search operations work correctly.

```
# pkgrepo refresh -s /export/IPS/repo
```

### 4. Configuring the repository server service

Now that your local copy is ready, use the `svccfg` command to configure the repository server service, specifying the location of your local repository and setting the `readonly` property to `true`:

```
# svccfg -s application/pkg/server setprop \
    pkg/inst_root=/export/IPS/repo
# svccfg -s application/pkg/server setprop pkg/readonly=true
# svcprop -p pkg/inst_root application/pkg/server
/export/IPS/repo
```

Next, refresh the service and then enable it.

```
# svcadm refresh application/pkg/server
# svcadm enable application/pkg/server
```

Verify that the service is enabled.

```
# svcs application/pkg/server
STATE      STIME      FMRI
online     17:00:56   svc:/application/pkg/server:default
```

**Note:** As an alternative, you can also use an NFS share for your IPS repository.

```
# zfs set share.nfs=on /export/IPS \
name=repoSolaris11,path/export/IPS,prot=nfs
# dfshares s11-ss
RESOURCE      SERVER  ACCESS  TRANSPORT
solaris:/export/IPS solaris -      -
```

## 5. Setting the local IPS publisher

The default publisher for Oracle Solaris 11 systems is `solaris`, and the default origin for that publisher is `http://pkg.oracle.com/solaris/release`. The publisher's origin is identified by its URI. This is the location of a machine or resource on the Internet.

To enable your clients to retrieve packages from your local repository, you need to reset the origin for the `solaris` publisher for each client by using the `pkg set-publisher` command.

```
# pkg publisher
PUBLISHER  TYPE      STATUS  P LOCATION
solaris    origin    online  F http://pkg.oracle.com/solaris/release
```

```
# pkg set-publisher -G '*' -g http://server.mydomain.com/
solaris
```

```
# pkg publisher
PUBLISHER  TYPE      STATUS  P LOCATION
solaris    origin    online  F http://server.mydomain.com/
```

## 6. Testing IPS on the local server

Test that the repository server is set up correctly by searching for the `entire` package.

```
# pkg search entire
INDEX      ACTION    VALUE                PACKAGE
pkg.fmri   set       solaris/entire       pkg:/entire@0.5.11-
0.175.1.0.0.24.2
```

The search returns the package information. This indicates that your local IPS repository server has been configured correctly.

## Quiz

You have downloaded and concatenated the two parts of the IPS repository files. Which of the following commands will you use, to access the resulting ISO image under the `/mnt` directory?

- a. `mount -F ufs -o ro sol-11_1-repo-full.iso /mnt`
- b. `mount -F hsfs -o ro sol-11_1-repo-full.iso /mnt`
- c. `mount -F nfs -o ro sol-11_1-repo-full.iso /mnt`
- d. `mount -F zfs -o ro sol-11_1-repo-full.iso /mnt`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

## Lesson Agenda

- Describing the features and capabilities of IPS
- Configuring a local IPS repository
- **Configuring client access to the local IPS server**
- Administering software packages
- Updating the Oracle Solaris 11 operating system
- Upgrading Oracle Solaris 11 to Oracle Solaris 11.1 operating system
- Administering boot environments

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Configuring Client Access to the Local IPS Server

After the local IPS repository is configured, you need to perform the following steps to configure each client to use the local repository server.

1. Verifying the prerequisite setup of the client
2. Setting the local IPS publisher
3. Testing client access to the local IPS server

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Verifying Prerequisites Setup

- Determine the client host and domain names.

```
# hostname  
client1  
# domainname  
mydomain.com
```

- Verify that the client system can access DNS services and can connect with the local IPS server.

```
# nslookup server  
Server:          192.168.0.100  
Address:         192.168.0.100#53  
Name:            server.mydomain.com  
Address:         192.168.0.100  
  
# ping server  
server is alive
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Setting the Local IPS Publisher

Set the publisher to the local IPS repository by using the `pkg set-publisher` command.

```
# pkg publisher
PUBLISHER      TYPE      STATUS P  LOCATION
solaris        origin   online F  http://pkg.oracle.com/solaris/release

# pkg set-publisher -G '*' -g http://server.mydomain.com/ solaris

# pkg publisher
PUBLISHER      TYPE      STATUS P  LOCATION
solaris        origin   online F  http://server.mydomain.com/
```

**To Do:** Make a note of the local publisher's URI; you will need it to complete the next task.



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

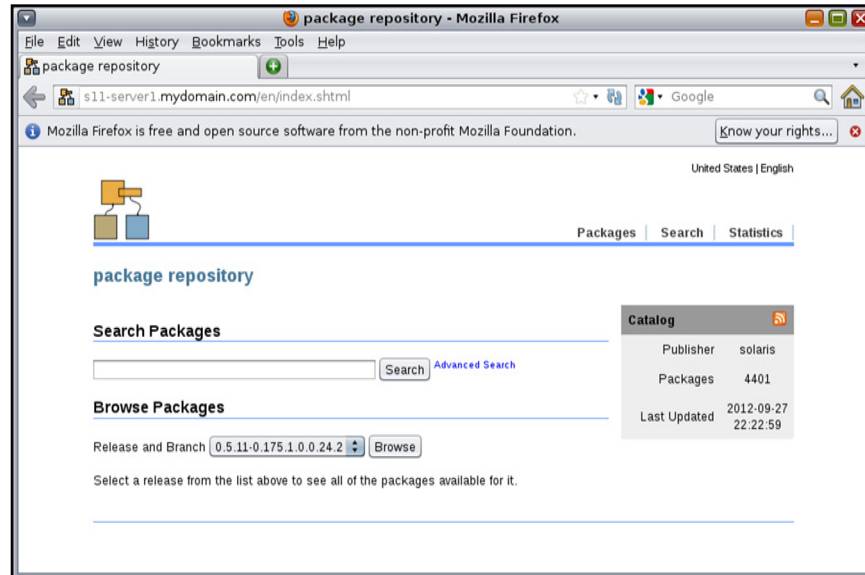
After you have verified network connectivity between the client and the local IPS repository server, set the publisher for each client to the local IPS publisher, just as you did when you configured the local IPS server.

For the server, the publisher is set to the ports on which the local repositories are configured. On the client, the publisher must be set to the server on which the local IPS repositories are configured.

First, check the current publisher. Next, set the publisher to the local IPS repository server, and then verify that the publisher is now the local IPS publisher.

## Testing Client Access to the Local IPS Server

To test client access to the IPS repository server, open the local publisher URI in a browser.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The final task is to test the client machine's access to the local IPS repository server. To do this, open the local publisher's URI in a browser. If a "package repository" page is returned, you have successfully configured the client's access to the local IPS repository server. You can now use your local IPS repository to manage your software package needs.

**Note:** In this example, the URI would be `http://s11-server1.mydomain.com`.



## Lesson Agenda

- Describing the features and capabilities of IPS
- Configuring a local IPS repository
- Configuring client access to the local IPS server
- **Administering software packages**
- Updating the Oracle Solaris 11 operating system
- Upgrading Oracle Solaris 11 to Oracle Solaris 11.1 operating system
- Administering boot environments

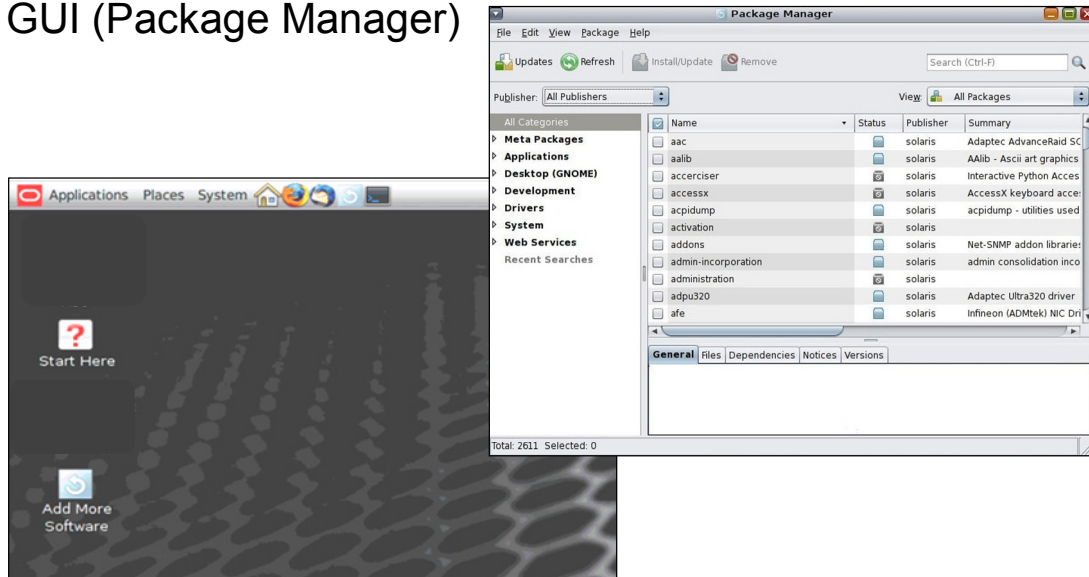
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# IPS Interfaces

IPS supports the following interfaces to administer packages:

- Command-line
- GUI (Package Manager)



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

With IPS, you have the option of using a command-line interface (CLI), graphical user interface (GUI), or web browser to perform package-management tasks.

# Package Management: CLI

Package-Management Task	IPS Command
Display package state and version information	<code>pkg list</code>
Display package information	<code>pkg info</code>
Display the contents of a package	<code>pkg contents</code>
Install package updates	<code>pkg update</code>
Install a package	<code>pkg install</code>
Verify package installation	<code>pkg verify</code>
Search for a package	<code>pkg search</code>
Uninstall a package	<code>pkg uninstall</code>

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

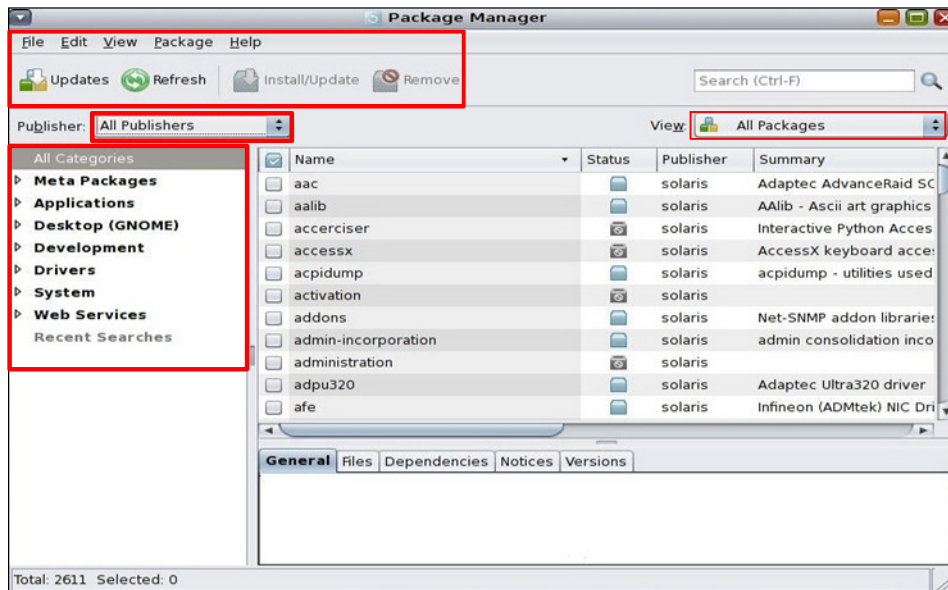
You can administer software packages by using the CLI. Some of the administration tasks include:

- Listing packages
- Displaying package information
- Installing and updating packages
- Viewing a package installation action without installing
- Verifying a package installation
- Searching for a package
- Uninstalling a package

The table in the slide summarizes, by task, the IPS package management and administration commands.

# Package Management: Package Manager

In addition to CLI, you can use Package Manager to manage software packages.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** Although Package Manager supports many of the same package-management tasks as those supported by CLI, the GUI organizes and displays the package information in slightly different ways, as can be seen in the screenshot in the slide.

## Lesson Agenda

- Describing the features and capabilities of IPS
- Configuring a local IPS repository
- Configuring client access to the local IPS server
- Administering software packages
- **Updating the Oracle Solaris 11 operating system**
- Upgrading Oracle Solaris 11 to Oracle Solaris 11.1 operating system
- Administering boot environments

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Software Updates

- The Oracle Solaris product engineering group releases software updates for the OS on a regular basis.
- The updates are published as Support Repository Updates (SRUs) to a web-based Oracle repository for distribution, <http://pkg.oracle.com/solaris/support/>.
- Each SRU is a single unit of change that you can use to update your system.
- SRUs contain a number of bug fixes and critical security fixes that, when applied to an existing Oracle Solaris 11 system, ensure that the system runs without any issues.

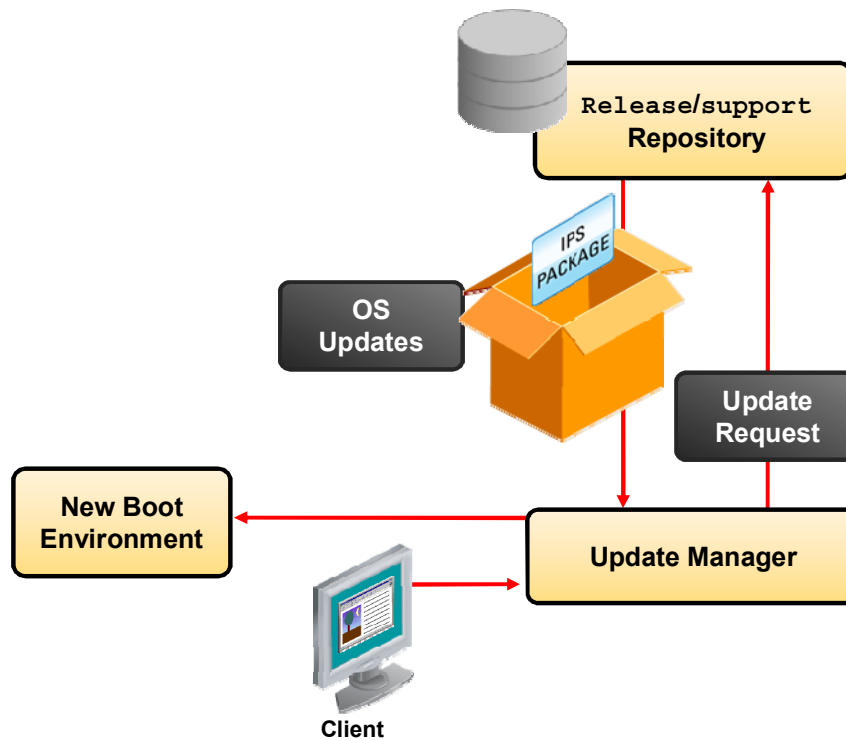
The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered within a solid red rectangular bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** Oracle customers with an active Oracle support plan have access to the SRUs.

**Best Practice:** Update the newly installed OS with the latest software release before you perform any more configuration activities, and then update the system on a regular basis thereafter.

# Software Update Process



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

During the update process, also called *updating an image*, each package is updated by default from the publisher that provided the current installed version. If particular packages are updated that affect the operating system's core programs, a new boot environment is created. You will learn more about boot environments later in the lesson.

## Update Interfaces

- Update Manager is used to update all installed packages to the latest SRUs.
- Update Manager is an IPS feature that automatically and continuously monitors for updates.
- Update Manager can be invoked by using the following options:
  - CLI (`pkg update` command)
  - GUI (Package Manager)

**Best Practice:** Before invoking Update Manager, ensure that all the publishers that provide you with SRUs are available from your system, either in a local or a web-based IPS repository.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Updating Your Local Repository

Before you update your local repository by using the following commands, ensure that your repository server is running the same or a newer version of the Oracle Solaris 11 OS as the version of the packages you are updating to:

1. Use the `pkgrecv` command to update the repository.

```
# pkgrecv -s http://pkg.oracle.com/solaris/release/ -d \  
/export/repoSolaris11 '*'
```

2. Run the `pkgrepo refresh` command to catalog any new packages found in the repository and update all search indexes.

```
# pkgrepo -s /export/repoSolaris11 refresh  
Initiating repository refresh.
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Before you update your local repository, ensure that your repository server is running the same or a newer version of the Oracle Solaris 11 OS as the version of the packages you are updating to. For example, if the server is running Oracle Solaris 11 and you want to update your repository to the Oracle Solaris 11 Update 1 repository, update the server to Oracle Solaris 11 Update 1 before you update your repository.

## Updating the OS by Using the CLI

Having updated the repository, you can now update the OS:

1. Check whether there are any updates available by using the `pkg list -u` command.

```
# pkg list -u
NAME (PUBLISHER)          VERSION          IFO
consolidation/SunVTS/SunVTS-incorporation  0.5.11-0.172.0.0.0.0.0  i--
consolidation/ips/ips-incorporation         0.5.11-0.175.0.0.0.2.2576 i--
consolidation/osnet/osnet-incorporation     0.5.11-0.175.0.0.0.2.1   i-
entire                                       0.5.11-0.175.0.0.0.2.0   i--
...
system/zones                             0.5.11-0.175.0.0.0.2.1   i--
```

2. Check for the latest package version, for example:

```
# pkg info -r system/zones
Name: system/zones
Summary: Solaris Zones
...
FMRI: pkg://solaris/system/zones@0.5.11,5.11-
0.175.0.1.0.4.1:20111110T192300Z
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Updating the OS by Using the CLI

### 3. Do a dry run of an update by using `pkg update -nv`.

```
# pkg update -nv
Packages to update: 13
Estimated space available: 11.91 GB
Estimated space to be consumed: 136.51 MB
Create boot environment: Yes
Activate boot environment: Yes
Create backup boot environment: No
... <Output truncated>
```

### 4. Perform the update by using `pkg update`.

```
# pkg update
Packages to update: 13
Create boot environment: Yes
Create backup boot environment: No
Download: consolidation/SunVTS/SunVTS-incorporation ... Done
...<Output Truncated>...
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note for Step 3:** It is a good practice to perform a dry run of a system update to see what changes will be made to the system.

Observe from the output that the system automatically creates a BE and there are 13 packages to update that will take up an estimated 136 MB. If for any reason something unexpected happens, you can rollback and reboot with the older BE. This makes the update procedure risk-free.

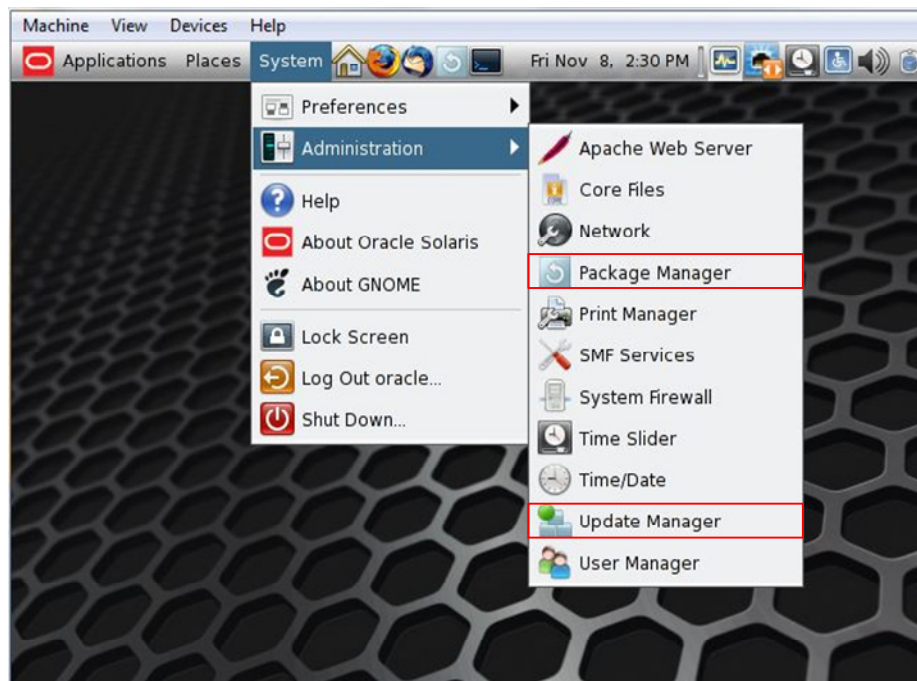
**Note for Step 4:** The output indicates that 13 packages have been downloaded and installed, and a new BE has been created. To list the BEs created on the system, use the `beadm list` command.

```
# beadm list
```

BE	Active	Mountpoint	Space	Policy	Created
-----	-----	-----	-----	-----	-----
solaris	N	/	404.0K	static	2011-11-23 00:10
solaris-1	R	-	2.39G	static	2011-11-25 00:15

In this case, the current BE is called `solaris`, but the new BE `solaris-1` will be active on reboot.

## Updating the OS by Using the GUI



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can invoke either Package Manager or Update Manager from the **System > Administration** menu on the desktop. With either Package Manager or Update Manager, the update procedure is rather intuitive.

**Note:** You can also access Package Manager from the command line: `/usr/lib/pm-launch packagemanager`. For more information about the Package Manager command-line options, see the Oracle Solaris 11 IPS documentation.

## Quiz

A new update is available for the `communication/im/pidgin` package. Which of the following commands would you use to make a note of the system changes that could occur following the update operation?

- a. `# pkg info -r communication/im/pidgin`
- b. `# pkg update -nv communication/im/pidgin`
- c. `# pkg update communication/im/pidgin`
- d. `# pkg list -u communication/im/pidgin`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

## Lesson Agenda

- Describing the features and capabilities of IPS
- Configuring a local IPS repository
- Configuring client access to the local IPS server
- Administering software packages
- Updating the Oracle Solaris 11 operating system
- **Upgrading Oracle Solaris 11 to Oracle Solaris 11.1 operating system**
- Administering boot environments

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Upgrading Oracle Solaris 11 to Oracle Solaris 11.1 OS

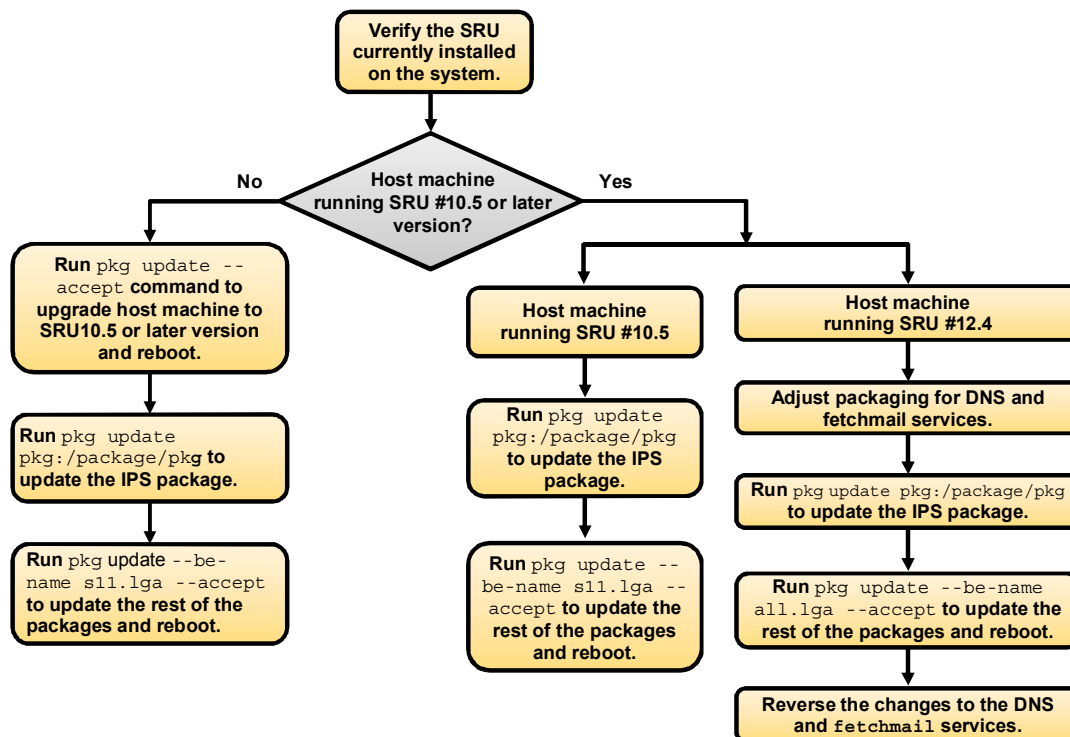
There are two approaches for upgrading the OS.

- By using the Oracle Solaris **support** repository, for those who have an Oracle support agreement  
<https://pkg.oracle.com/solaris/support>
- By using the Oracle Solaris **release** repository, for those without an Oracle support agreement  
<http://pkg.oracle.com/solaris/release>

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Upgrading the OS by Using the Oracle Solaris Support Repository



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you have an Oracle support agreement, you can use the Oracle Solaris support repository, in which case the package publisher definition should look like the following:

```
% pkg publisher
```

PUBLISHER	TYPE	STATUS	P	LOCATION
Solaris	origin	online	F	<a href="https://pkg.oracle.com/solaris/support">https://pkg.oracle.com/solaris/support</a>

The release process for Oracle Solaris 11 was changed to accelerate the release of critical bug fixes. A consequence of this process is that the update might not contain some bug fixes released in an SRU for the preceding release.

For example, the most recent SRU for Oracle Solaris 11 11/11 is SRU#12.4. Some packages in SRU#12.4 have a version number that is higher than the version number in the Oracle Solaris 11.1 release. To preserve these bug fixes, the version difference will prevent the upgrade to the Oracle Solaris 11.1 release. If the upgrade process is not blocked because of the version numbers, some fixes that are part of the SRU will be removed if you upgrade to the Oracle Solaris 11.1 release. For convenience, these bug numbers are listed in <http://www.oracle.com/technetwork/articles/servers-storage-admin/howto-update-11dot1-ips-1866781.html#bugs>.



The steps that are required might be different due to some systems running different SRU versions. Before you start upgrading, first verify which SRU your system is running by displaying the information about `pkg:/entire`. In the following example, SRU#10.5 is installed.

```
# pkg info entire
```

```
    Name: entire
    Summary: entire incorporation including Support Repository Update
              (Oracle Solaris 11 11/11 SRU 10.5).
    Description: This package constrains system package versions to the same
                  build.  WARNING: Proper system update and correct package
                  selection depend on the presence of this incorporation.
                  Removing this package will result in an unsupported system.
                  For more information see
                  https://support.oracle.com/CSP/main/article
                  ?cmd=show&type=NOT&doctype=REFERENCE&id=1372094.1.
    Category: Meta Packages/Incorporations
    State: Installed
    Publisher: solaris
    Version: 0.5.11 (Oracle Solaris 11 SRU 10.5)
    Build Release: 5.11
    Branch: 0.175.0.10.0.5.0
    Packaging Date: Fri Aug 03 18:26:27 2012
    Size: 5.45 kB
    FMRI: pkg://solaris/entire@0.5.11,5.11-
    0.175.0.10.0.5.0:20120803T182627Z
```

If your system has Oracle Solaris 11 11/11 installed, without any SRUs, the package information looks like:

```
# pkg info entire
```

```
    Name: entire
    Summary: Incorporation to lock all system packages to the same build
    Description: This package constrains system package versions to the same
                  build.  WARNING: Proper system update and correct package
                  selection depend on the presence of this incorporation.
                  Removing this package will result in an unsupported system.
    Category: Meta Packages/Incorporations
    State: Installed
    Publisher: solaris
    Version: 0.5.11
    Build Release: 5.11
    Branch: 0.175.0.0.0.2.0
```

Packaging Date: October 20, 2011 02:38:22 PM

Size: 5.45 kB

FMRI: pkg://solaris/entire@0.5.11,5.11-0.175.0.0.0.2.0:20111020T143822Z

Depending on the SRU that your system is running, perform one of the following procedures:

- Upgrading a System Running an SRU Before SRU#10.5
- Upgrading a System Running SRU#10.5 or SRU#11.4
- Upgrading a System Running at Least SRU#12.4

**Note:** The amount of time each procedure takes is dependent on many factors, including system and network resources.

### Upgrading a System Running an SRU earlier to SRU#10.5

You can choose to update to SRU#10.5 or to a later SRU release. Note that SRUs starting with SRU#12.4 have certain fixes that will not be present in Oracle Solaris 11.1 until the first Oracle Solaris 11.1 SRU is available.

1. Verify which Oracle Solaris 11 SRU you have installed.
2. Run the `pkg update` command to update the system. It creates a new boot environment.
  - To update to SRU#10.5, use the following command:  
`# pkg update --accept entire@0.5.11,5.11-0.175.0.10`
  - To update to the latest SRU, use the following command:  
`# pkg update --accept`
3. Reboot using the updated boot environment.  
`# reboot`
4. Follow the instructions for upgrading the SRU (that you installed in step 2) by referring to one of the following sections.
  - Upgrading a system running SRU#10.5 or SRU#11.4
  - Upgrading a system running at least SRU#12.4

### Upgrading a System Running SRU#10.5 or SRU#11.4

1. Verify which Oracle Solaris 11 SRU you have installed.
2. (SPARC only) For any SPARC-based system that has one or more zones installed, for each zone installed on the system, remove the `pkg:/system/ldoms/ldomsmanager` package as follows:  
`# for z in `zoneadm list`; do zlogin $z pkg uninstall ldomsmanager; done`
3. Update the IPS package.  
`# pkg update pkg:/package/pkg`

**Note:** Due to earlier bugs in some packages, it was possible to incorrectly install those packages on a system. This command removes the following bad packages, if they are installed:

- x86: `pkg://solaris ldoms/ldoms-incorporation`
- SPARC:
  - `pkg://solaris/consolidation/nvidia/nvidia-incorporation`
  - `pkg://solaris/driver/network/ethernet/elxl`
  - `pkg://solaris/driver/network/ethernet/pcn`
  - `pkg://solaris/driver/network/ethernet/dnet`
  - `pkg://solaris/driver/network/ethernet/iprb`

4. Update the remaining system packages.

```
# pkg update --be-name s11.1ga --accept
```

5. Reboot using the updated boot environment.

```
# reboot
```

### Upgrading a System Running at Least SRU#12.4

If any of the following packages are installed, you must either remove them and add them back after the update to Oracle Solaris 11.1, or disassociate them from the constraints on the system (called *unlocking*), as described in this procedure:

- `pkg://solaris/network/dns/bind`
- `pkg://solaris/service/network/dns/bind`
- `pkg://solaris/mail/fetchmail`

1. Adjust packaging for the DNS and fetchmail services. If you are not using these packages, you can remove them. If you have dependencies on these packages, you must unlock them. Only follow this step if one or more of the three packages listed is installed on your server.

- To remove the packages, run this command:

```
# pkg uninstall pkg://solaris/network/dns/bind
pkg://solaris/service/network/dns/bind
pkg://solaris/mail/fetchmail
```

If you have any zones installed on your system, you must also uninstall these packages in each zone. For example:

```
# for z in `zoneadm list`; do zlogin $z pkg uninstall
pkg://solaris/network/dns/bind; done
```

- To unlock the packages, run this command:

```
# pkg change-facet facet.version-lock.mail/fetchmail=false
# pkg change-facet facet.version-lock.service.network/dns/bind=false
# pkg change-facet facet.version-lock.network/dns/bind=false
```

The previous commands must be performed in each configured zone with the packages installed.

2. Update the IPS package.  
# `pkg update pkg:/package/pkg`
3. Update the rest of the packages.  
# `pkg update --be-name s11.1ga --accept`
4. Reboot using the updated boot environment.  
# `reboot`
5. Reverse the changes to the DNS and fetchmail services.
  - If you removed the packages in Step1, reinstall them.  
# `pkg install pkg://solaris/service/network/dns/bind`  
# `pkg install pkg://solaris/mail/fetchmail`  
**Note:** The first command installs `pkg://solaris/network/dns/bind`, so it is not necessary to install that package.
  - If you unlocked the packages in step 1, when the first SRU is released for Oracle Solaris 11.1, lock the packages.  
# `pkg change-facet facet.version-lock.mail/fetchmail=true`  
# `pkg change-facet facet.version-lock.service/network/dns/bind=true`  
# `pkg change-facet facet.version-lock.network/dns/bind=true`

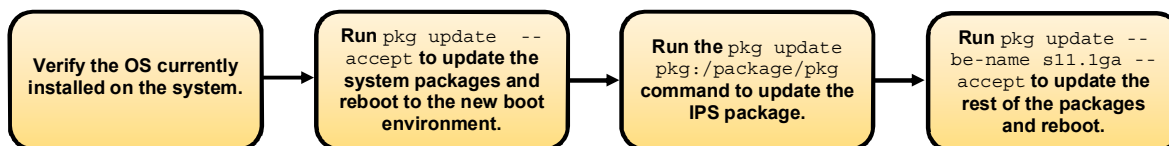
Relocking the packages updates them as required.

### Upgrading a System to Oracle Solaris 11.1 with the Latest SRU

Once the first SRU for Solaris 11.1 is released, use the following procedure to upgrade to Oracle Solaris 11.1 with the new SRU.

1. Verify which Oracle Solaris 11 SRU you have installed.
2. (For early SRUs only) On systems running an SRU earlier than 10.5, upgrade to the latest Oracle Solaris 11 SRU.
  - a. Update the SRU using the `pkg update` command. This command updates the OS to Oracle Solaris 11 with the latest SRU.  
# `pkg update --accept`
  - b. Reboot using the updated boot environment.  
# `reboot`
3. (For all SRUs) Update to Oracle Solaris 11.1 with the latest SRU.  
# `pkg update --accept --be-name s11.1sru`
4. Reboot using the updated boot environment.  
# `reboot`

# Upgrading the OS by Using the Oracle Solaris Release Repository

**ORACLE**

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use the following procedure to upgrade an Oracle Solaris 11 11/11 system by using Oracle's web-based release repository, if you do not have a support agreement, or have no SRUs installed.

1. Update the system packages by using the `pkg update` command. This command creates a new boot environment.  
`# pkg update --accept`

**Note:** If your system cannot directly connect to Oracle's web-based release repositories, download the Oracle Solaris 11.1 Pre-Upgrade Repository Image from <http://www.oracle.com/technetwork/server-storage/solaris11/downloads/index.htm>. This image contains packages that you must update prior to updating to Oracle Solaris 11.1. Refer to the README file of the ISO for the necessary steps.

2. Reboot by using the updated boot environment.  
`# reboot`

3. (SPARC only) In any SPARC-based system that has one or more zones installed, for each zone installed on the system, remove the `pkg:/system/ldoms/ldomsmanager` package.  

```
# for z in `zoneadm list`;
do zlogin $z pkg uninstall ldomsmanager;
done
```
4. Update the IPS package.  

```
# pkg update pkg:/package/pkg
```
5. Update the rest of the packages.  

```
# pkg update --be-name s11.1ga -accept
```
6. Reboot using the updated boot environment.  

```
# reboot
```

For more information about the upgrade process, refer to

<http://www.oracle.com/technetwork/articles/servers-storage-admin/howto-update-11dot1-ips-1866781.html>.

## Quiz

You are about to perform a system upgrade from Oracle Solaris 11 to Oracle Solaris 11.1. Which of the following commands would you use to identify the current SRU of your system based on which you need to choose the appropriate upgrade path?

- a. `pkg publisher`
- b. `uname -a`
- c. `pkg info entire`
- d. `cat /etc/release`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: c**

## Lesson Agenda

- Describing the features and capabilities of IPS
- Configuring a local IPS repository
- Configuring client access to the local IPS server
- Administering software packages
- Updating the Oracle Solaris 11 operating system
- Upgrading Oracle Solaris 11 to Oracle Solaris 11.1 operating system
- **Administering boot environments**

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



# Overview of Boot Environments

- A boot environment (BE) is a bootable instance of an Oracle Solaris 11 OS image.
- Multiple BEs can be maintained on a system.
- BEs can have different software versions installed.
- BEs make updating software a low-risk operation.

The Oracle logo, consisting of the word "ORACLE" in white, uppercase letters on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A boot environment (BE) is a bootable instance of an OS image. That is, a BE contains a version of the OS that can be started and is functional.

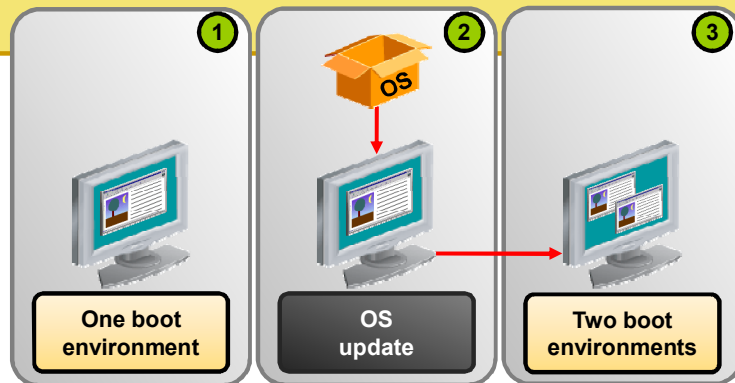
You can maintain multiple BEs on your system, and each BE can have different software versions installed. When you boot your system, you have the option to boot in to any of the BEs on the system.

With multiple boot environments, the process of updating software becomes a low-risk operation. System administrators can create backup boot environments before making software updates to their system. Administrators have the option of booting a backup boot environment, if necessary.

## Boot Environment Creation

When you first install the Oracle Solaris operating system, a new BE is automatically created.

If you then run the update function on this image to update all the installed packages to the latest version, a new BE is again created. The system sets this new BE as the default boot choice the next time the system is booted. The original or current BE remains as an alternative boot choice.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The result is that you now have two BEs: the new BE contains the most current version of the operating system and the alternative BE contains an older version of the operating system. Having the alternative boot environment enables you to return to that version of the operating system if you encounter issues with the new version.

## BE Management Utilities

IPS provides two BE management utilities that you can use for performing administration tasks:

- `beadm` command: Provides a full range of options for managing boot environments. Requires root privileges.
- Package Manager: Provides a subset of boot environment management options.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Administering Boot Environments

The following are some of the administration tasks you can perform with BEs:

- Listing existing BEs
- Creating a new BE
- Renaming an existing inactive BE
- Destroying an existing inactive BE
- Activating an existing inactive BE
- Mounting and unmounting an inactive BE
- Installing and uninstalling a package on an inactive, mounted BE
- Creating a snapshot of a BE
- Creating a BE from an existing snapshot

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Administering boot environments is part of a system administrator's responsibility. Recall that a new BE is automatically created after an upgrade procedure. The new BE provides you with a new current BE and an alternative BE based on the older version of the OS.

## Listing the BEs on the System

To list the BEs on a system, run `beadm list`.

# <code>beadm list</code>						
BE	Active	Mountpoint	Space	Policy	Created	
--	-----	-----	-----	-----	-----	
solaris	NR	/	4.53G	static	2013-11-08	05:47
solaris-1	-	-	67.0K	static	2013-11-08	01:01
solaris-2	-	-	67.0K	static	2013-11-08	01:01

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Before you start managing the boot environments on a system, you should determine which boot environments exist on the system. To determine this, use the `beadm list` command, as shown in the example.

The `beadm list` command displays the following information:

- **BE:** Name of the boot environment.
- **Active:** Boot status of the boot environment. In the example for the `solaris` boot environment, `N` means that the boot environment is currently active and `R` means that it will be the boot environment that will be active on reboot as well. A dash (-) indicates that the BE is currently inactive.
- **Mountpoint:** Where the boot environment is mounted. In the example, the `solaris` boot environment is mounted in the root (/) directory.
- **Space:** Size of the boot environment.
- **Policy:** Either static or volatile.
- **Created:** Date the boot environment was created.

**Note:** A number of options can be used with the `beadm list` command but that are outside the scope of this course. For more information about these options, see the Oracle Solaris 11 documentation for managing boot environments.

## Creating a New BE

To create a new BE, run `beadm create beName`.

```
# beadm create test1
# beadm list
BE          Active  Mountpoint Space  Policy  Created
--          -
solaris     NR      /          4.53G  static  2013-11-08 05:47
solaris-1   -       -          67.0K  static  2013-11-11 01:01
solaris-2   -       -          67.0K  static  2013-11-11 01:01
test1       -       -          67.0K  static  2013-11-11 01:34
#
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The system automatically creates a new boot environment as part of an update operation if the software packages that are being installed affect the core operating system files.

However, there may be times when you need to manually create a new boot environment. For example, you might want to create a backup of an existing boot environment prior to modifying the original boot environment for the purpose of testing a new application.

To create a new boot environment from the active boot environment, use the `beadm create` command with the name of the new boot environment, as shown in the example. The `beadm` command creates a new boot environment that is a clone of your active boot environment. This clone is inactive. Running `beadm list` again lists the new boot environment.

**Note:** You can also create a boot environment from an inactive boot environment. For more information about how to perform this task, see the Oracle Solaris 11 documentation for managing boot environments.

## Renaming an Existing Inactive BE

To rename a BE, run `beadm rename beName newBeName`.

```
# beadm rename test1 apptest1
# beadm list
BE          Active  Mountpoint Space  Policy  Created
--          -
apptest1    -        -        67.0K  static  2013-11-11 01:03
solaris     NR        /        4.53G  static  2013-11-08 05:47
solaris-1   -        -        67.0K  static  2013-11-11 01:01
solaris-2   -        -        67.0K  static  2013-11-11 01:01
#
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can rename an existing inactive boot environment by using the `beadm rename` command with the current boot environment name, followed by the new boot environment name. In this example, the boot environment name `test1` is being changed to `apptest1`.

You can run `beadm list` to verify that the name change has been made. In the example, you can see that the name has changed.

**Note:** You cannot rename an active boot environment.

## Destroying an Existing Inactive BE

To destroy a BE, run `beadm destroy beName`.

```
# beadm destroy solaris-2
Are you sure you want to destroy solaris-2? This action cannot
be undone (y/[n]): y
# beadm list
BE          Active  Mountpoint Space  Policy  Created
--          -
apptest1    -        -        67.0K  static  2013-11-11 01:03
solaris     NR        /        4.53G  static  2013-11-08 05:47
solaris-1   -        -        67.0K  static  2013-11-11 01:01
#
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you no longer need a particular boot environment, or if you want to make more room on your system, you can delete it by using the `beadm destroy` command with the boot environment name. In this example, the inactive boot environment `solaris-2` is being destroyed. The system asks you for confirmation before destroying the boot environment.

**Note:** You can use the `-F` option to force the boot environment to be destroyed.

To verify that the boot environment has been removed, you can run `beadm list` again. The boot environment should not be listed.

Consider the following when destroying a boot environment:

- You cannot destroy the boot environment that is currently booted.
- The `beadm destroy` command automatically removes the destroyed boot environment's entry from the x86 GRUB menu or the SPARC boot menu.



## Activating an Existing Inactive BE

To activate a BE, run `beadm activate beName`.

```
# beadm activate apptest1
# beadm list
BE          Active  Mountpoint Space  Policy  Created
--          -
apptest1    R        -        4.53K  static  2013-11-11 01:03
solaris     N        /        901.0K static  2013-11-08 05:47
solaris-1   -        -        138.0K static  2013-11-11 01:01
# init 6
```

ORACLE

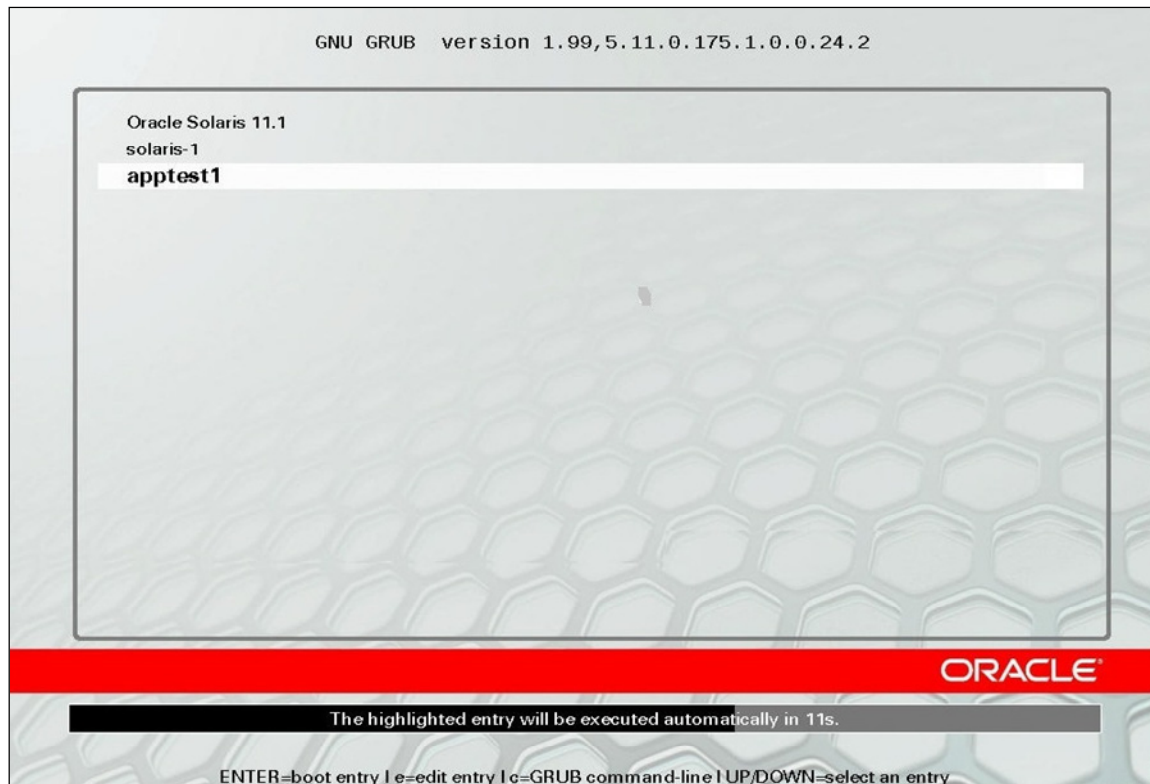
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To activate an existing inactive boot environment, use the `beadm activate` command followed by the boot environment name. In this example, the `apptest1` environment is being activated.

To verify that the boot environment has been activated, you can run `beadm list` again. Notice that the `Active` status of the current boot environment `solaris` has changed from `NR` to `N`, and the newly activated boot environment `apptest1` now has an `R` in the `Active` column. Recall that `R` means that this boot environment will become the active boot environment on reboot.

To activate the new boot environment, you must reboot the system by using the `init 6` command.

# Verifying the New BE



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

On an x86 machine, after the system has rebooted and the GNU GRUB menu appears, as shown in the example, you can verify that the new boot environment is now the default active boot environment. The default active boot environment is the highlighted entry.

On a SPARC machine, when you have multiple BEs, you issue `init 0` to go to OBP (ok prompt) and then you use the following command sequence:

```
ok boot -L
Rebooting with command: boot -L
Boot device: /pci@7c0/pci@0/pci@1/pci@0,2/LSILogic,sas@2/disk@0
File and args:      -L

1 Oracle Solaris 11 11.1
2 solaris-1
3 apptest1
Select environment to boot: [ 1 - 3 ]: 3
To boot the selected entry, invoke:
boot [<root-device>] -Z rpool/ROOT/apptest1

Program terminated
ok boot -Z rpool/ROOT/apptest1
```

## Mounting an Inactive BE

To mount an inactive boot environment, run `beadm mount beName mountpoint`.

```
# beadm mount solaris-1 /solaris-1
# beadm list
```

BE	Active	Mountpoint	Space	Policy	Created
--	-----	-----	-----	-----	-----
solaris	NR	/	2.38G	static	2013-11-08 03:50
solaris-1	-	/solaris-1	169.01M	static	2013-12-10 22:14

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Suppose you need to install and test several new packages on the operating system. However, you do not want to impact the production environment. In that case, you can mount an existing inactive BE that is a copy of the existing active BE and use that for testing.

To mount an existing inactive BE, you must run the `beadm mount` command followed by the name of the inactive BE and the location of the mount point. The example in the slide mounts the `solaris-1` inactive BE to `/solaris-1`.

**Note:** If the directory for the mount point does not exist, the `beadm` utility creates the directory and then mounts the boot environment on that directory. If the boot environment is already mounted, the `beadm mount` command fails and does not remount the boot environment at the newly specified location.

To verify that the inactive BE is mounted, run the `beadm list` command again, as shown in the example. Observe that the boot environment is mounted but remains inactive.

## Unmounting an Inactive BE

To unmount an inactive BE, run `beadm unmount beName`.

```
# beadm unmount solaris-1
# beadm list
BE          Active  Mountpoint Space  Policy  Created
--          -
solaris     NR      /          2.38G   static  2013-11-08 03:50
solaris-1   -        -          170.01M static  2013-12-10 22:14
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Before you reboot the BE, you must unmount it. To unmount an inactive BE, run the `beadm unmount` command, followed by the name of the inactive BE, as shown in the example in the slide.

**Note:** You cannot unmount the BE that is currently booted.

To verify that the inactive BE is unmounted, run the `beadm list` command again, as shown in the example. Observe that the boot environment is now unmounted.

From this point, you can activate the inactive boot environment by using the `beadm activate` command, and then reboot the system.

## Installing a Package on an Inactive Mounted BE

To install a package on an inactive boot environment,  
run `pkg -R mountpoint install packagename`.

```
# pkg -R /solaris-1 install newpkg
Creating plan...
      Packages to install:      1
      Create boot environment:  No
      Create backup boot environment:  No

DOWNLOAD                                PKGS      FILES      XFER (MB)   SPEED
Completed                              1/1        3/3        0.1/0.1     43.8k/s

PHASE                                ITEMS
Installing new actions                19/19
Updating package state database        Done
Updating image state                   Done
Creating fast lookup database           Done
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

After verifying that the inactive BE is mounted, you can install one or more packages by using the `pkg install` command with the uppercase `-R` option, the mount point location, and the package name, as shown in the example in the slide.

**Note:** The uppercase `-R` option specifies an alternate root to be used to identify the specific packages to be installed or updated.

## Uninstalling a Package on an Inactive Mounted BE

To uninstall a package on an inactive BE, run `pkg -R mountpoint uninstall packagename`.

```
# pkg -R /solaris-1 uninstall newpkg
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To uninstall a package, use `pkg uninstall` with the uppercase `-R` option, the mount point location, and the package name, as shown in the example in the slide.

## Creating a Backup of a BE

To create a backup of the BE, run `beadm create BEname@snapshotdescription`.

```
# beadm create solaris@backup
# beadm list -a solaris
```

BE/Dataset/Snapshot	Active	Mountpoint	Space	Policy	Created
-----	-----	-----	-----	-----	-----
solaris					
rpool/ROOT/solaris	NR	/	2.27G	static	2013-10-29 11:32
rpool/ROOT/solaris/var	-	/var	112.37M	static	2013-10-29 11:32
rpool/ROOT/solaris/var@2012-10-29-12:17:23	-	-	760.5K	static	2013-10-29 17:47
rpool/ROOT/solaris/var@backup	-	-	24.0K	static	2013-10-29 18:06
rpool/ROOT/solaris/var@install	-	-	18.86M	static	2013-10-29 11:37
rpool/ROOT/solaris@2012-10-29-12:17:23	-	-	50.27M	static	2013-10-29 17:47
rpool/ROOT/solaris@backup	-	-	0	static	2013-10-29 18:06
rpool/ROOT/solaris@install	-	-	53.28M	static	2013-10-29 11:37

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To create a backup of the boot environment, use the `beadm create` command followed by the BE name and a backup description. In the example in the slide, a backup of the `solaris` BE is being created.

To verify that the backup was created, you can run `beadm list -a` followed by the BE name, as shown in the example.

# Creating a BE From an Existing Backup

To create a BE from an existing backup, run `beadm create -e BEname@snapshotdescription beName`.

```
# beadm create -e solaris@backup solaris-2
# beadm list
BE          Active Mountpoint Space   Policy Created
--          -
solaris     NR      /           2.38G  static 2013-11-08 03:50
solaris-1   -       -           170.01M static 2013-12-10 22:14
solaris-2   -       -           28.0K  static 2013-12-10 22:59
# beadm activate solaris-2
# beadm list
BE          Active Mountpoint Space   Policy Created
--          -
solaris     N       /           2.38G  static 2013-11-08 03:50
solaris-1   -       -           170.01M static 2013-12-10 22:14
solaris-2   R       -           28.0K  static 2013-12-10 22:59
# init 6
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A backup of a BE is not bootable. However, you can create a new BE from an existing backup, and then activate and boot the new BE.

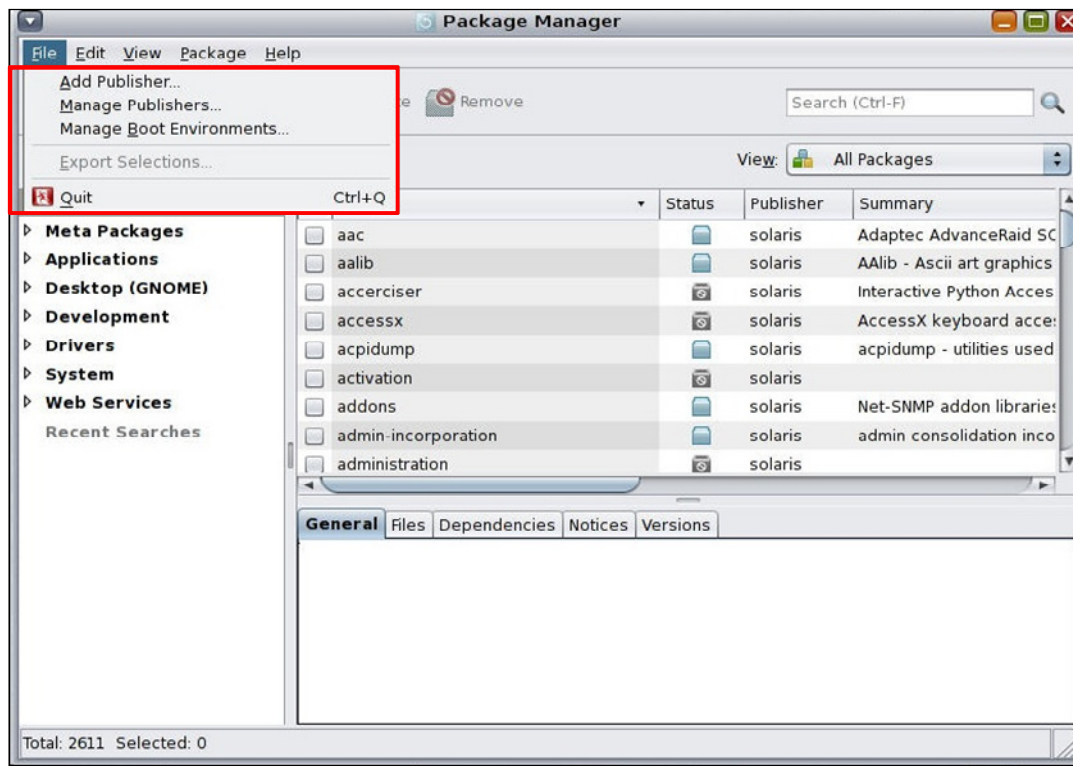
In the example in the slide, a new BE called `solaris-2` is created from the backup `solaris@backup`.

The `beadm list` command shows that `solaris-2` is created.

To make the new BE the current boot environment, you must activate it and then reboot the system, as shown in the example. Note that if you run the `beadm list` command again, you can see that `solaris-2` is now the BE that is active on reboot as designated by the `R` in the `Active` column.



# Managing BEs with Package Manager



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Although you cannot create new BEs by using Package Manager, you can use this utility to rename, destroy, and activate existing, inactive boot environments that are on the system.

To manage boot environments with Package Manager, open **Package Manager** and then select **Manage Boot Environments** from the **File** menu. This launches the **Manage Boot Environments** window. From there on, the GUI provides an intuitive interface to manage and administer BE-related activities.

## Quiz

Having booted into the `solaris-1` BE, you wish to install the `thunderbird` package in the inactive BE, `solaris`. Which of the following commands would you use to make `solaris` accessible for installing the package?

- a. `beadm activate solaris`
- b. `beadm mount solaris /mnt`
- c. `beadm unmount solaris-1`
- d. `pkg -R solaris install thunderbird`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

## Summary

In this lesson, you should have learned how to:

- Describe the features and capabilities of IPS
- Configure a local IPS repository
- Configure client access to the local IPS server
- Administer software packages
- Update the Oracle Solaris 11 operating system
- Upgrade Oracle Solaris 11 to Oracle Solaris 11.1 operating system
- Administer boot environments

ORACLE

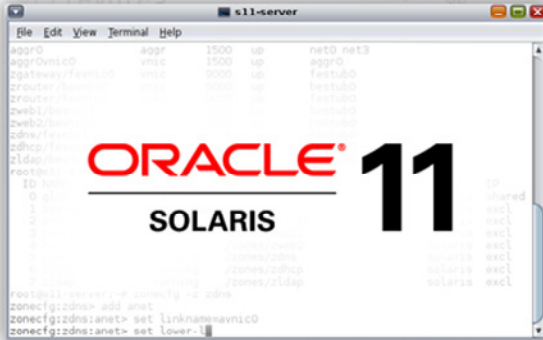
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Administering Services by Using SMF

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## System Administration for Experienced UNIX/Linux Administrators



Administering System  
Software by Using IPS



Administering Services  
by Using SMF



Administering ZFS



Configuring the Network



Administering Oracle Solaris  
Zones



Administering Privileges  
and RBAC



Installing the Oracle Solaris 11  
Operating System



Monitoring System Resources

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Objectives

After completing this lesson, you should be able to:

- Describe the features and capabilities of SMF
- Administer SMF services
- Configure SMF services
- Secure SMF services
- Troubleshoot SMF services

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Lesson Agenda

- Describing the features and capabilities of SMF
- Administering SMF services
- Configuring SMF services
- Securing SMF services
- Troubleshooting SMF services

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Managing Services in Older UNIX OSs

- The Solaris Operating Systems (OSs) prior to Oracle Solaris 10 use a System V type `init` process to manage system services.
- When a system boots, the kernel starts the `init` daemon.
- The `init` daemon reads the `/etc/inittab` configuration file.
- The configuration file initiates the `run` command (`rc`) scripts that take the system to the default run level 3, which is multi-user with server functionality enabled.
- However, this approach has limitations on service failure detection, dependency management, and service restart, as well as performance limitations due to the sequential startup of services.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Solaris 1.x and Solaris 2.x (including Oracle Solaris 8, 9, 10, and 11) initiate system services by running the `init` daemon. It is the first process that is started by the kernel and has PID 1. What is different in Oracle Solaris 8 and Oracle Solaris 9 as compared to Oracle Solaris 10 and Oracle Solaris 11 is what the `init` process does after it is started. In Oracle Solaris 8 and Oracle Solaris 9, the `/sbin/init` process reads `/etc/inittab`, which starts a System V type service initialization. However, in Oracle Solaris 10 and Oracle Solaris 11, the `/sbin/init` process reads `/etc/inittab`, which starts the SMF service initialization.

## Managing Services Since Oracle Solaris 10

- Service Management Facility (SMF), first introduced in Oracle Solaris 10, is a feature of the OS for managing system and application services.
- SMF replaces the System V `rc` scripting start-up mechanism.
- SMF improves the availability of a system by ensuring that service status is monitored, failed services are restarted together with their dependencies, and service logs are consistently maintained.
- The dependency relationship in SMF provides a key benefit in that services can be started in parallel during system start up, allowing faster boot than the legacy `init` framework, which starts processes in sequence.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Overview of SMF

- SMF provides the framework for managing:
  - System services
  - Interaction of services with other services
- SMF contains information about:
  - Procedures to start, stop, and restart services
  - Service startup behavior and status
  - Misconfigured services (such as an explanation of why a service is not running)

**Note:** SMF is also one of the components of the wider Oracle Solaris Predictive Self-Healing capability.
- SMF provides individual log files for each service

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered within a solid red rectangular bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Self-Healing Capability:** An SMF service, once started, can start several different processes that are tied together as part of a service contract. This means that an administrator needs to manage only the higher-level service, rather than worrying about a series of individual processes and what start order might be required by those processes. If a service fails for any reason, whether during a hardware or software fault, SMF automatically detects the failure and restarts the service and any dependent services.

## SMF Concepts

The following concepts are useful in understanding the SMF framework and its role in administering services:

- SMF service
- Service identifiers
- Service states
- SMF utilities
- SMF components
  - SMF profiles
  - Service configuration repository
  - Restarters
  - SMF manifests
- SMF repository backups
- SMF snapshots

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# SMF Service

An SMF service:

- Is an entity that provides a resource or list of capabilities to applications and other resources
- Is the software state of a device (for example, a configured network device or mounted file system)
- Is structured within SMF by:
  - Category (examples: `application`, `network`, `system`)
  - Service name (examples: `login`, `SSH server`, `hostid`)
  - Instance name: Specific configuration of a service
    - A single-instance service would usually use the instance name `default`. Example: `svc:/network/nfs/rquota:default`
    - Multiple instances of a service. Examples:  
`svc:/network/dhcp/server:ipv4`  
`svc:/network/dhcp/server:ipv6`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Services that are managed by SMF are structured by category, service name, and instance name.

- Examples of categories: `application`, `device`, `legacy`, `milestone`, `network`, `platform`, `site`, `system`
- Examples of service names: `login`, `SSH server`, and `hostid`
- Example of an instance: A web server is a service. A specific web server daemon that is configured to listen on port 80 is an instance. An instance name would be `default`, if it is a single instance. There can be multiple instances of a service. For example, a system can have more than one configured network interface or more than one mounted file system.

## Service Identifier

- SMF identifies each service instance by a service identifier.
- This service identifier is in the form of a Fault Management Resource Identifier (FMRI).
- For example: `svc:/network/telnet:default`

FMRI Segment	Description
<code>svc</code>	Indicates that this service is managed by SMF
<code>network</code>	Is the category to which the telnet service belongs to
<code>telnet</code>	Is the actual name of the service
<code>default</code>	Is the instance of the telnet service



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Recall that the Oracle Solaris 11 packages also use an FMRI. The FMRI in SMF indicates the service type or category, service name, and instance name. An example of a service FMRI is shown in the slide. Note that you do not always have to use the full FMRI when executing a command for a service. You can use a shorter form. For example, with `svc:/system/filesystem/root:default`, you can just use `filesystem/root`.

**Note:** All SMF-related FMRI's are prefixed with the `svc:/` scheme, except for legacy services, which are prefixed with the `lrc:/` scheme.

# Service States

Each service is in one of the following states at a given time:

- degraded
- disabled
- legacy rc\*.d script-initiated instance
- maintenance
- offline
- online
- uninitialized

ORACLE

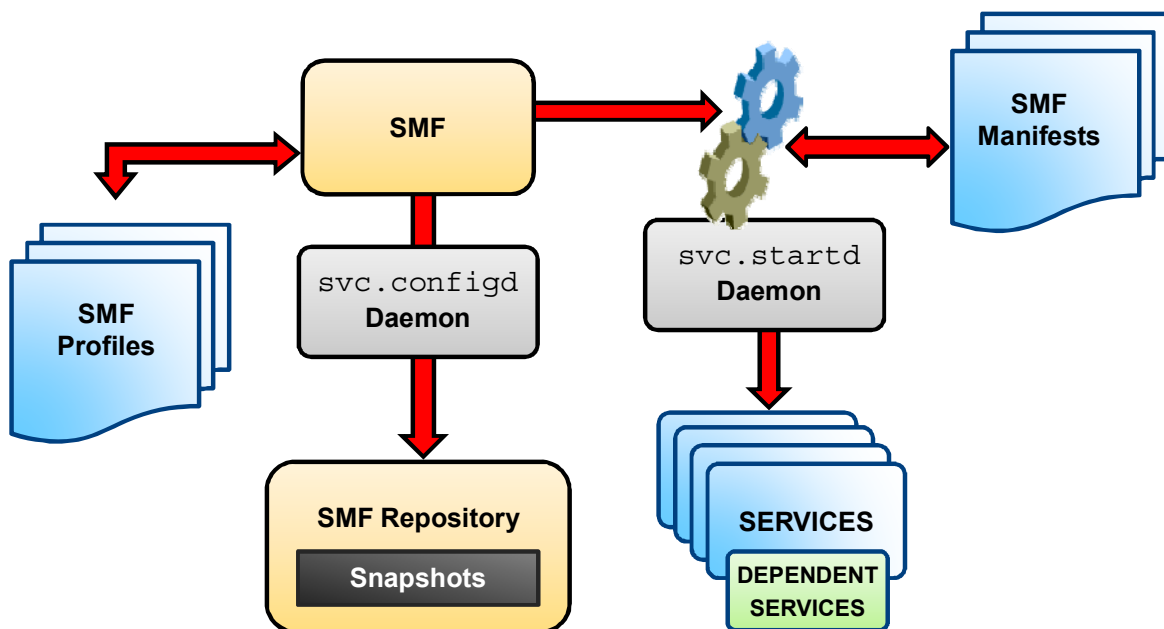
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `svcs` command displays the state, start time, and FMRI of service instances. The state of each service is one of the following:

- **degraded:** The service instance is enabled, but is running at a limited capacity.
- **disabled:** The service instance is not enabled and is not running.
- **legacy\_run:** The legacy service is not managed by SMF, but the service can be observed. This state is only used by legacy services.
- **maintenance:** The service instance has encountered an error that must be resolved by the administrator.
- **offline:** The service instance is enabled, but the service is not yet running or available to run.
- **online:** The service instance is enabled and has successfully started.
- **uninitialized:** This state is the initial state for all services before their configuration has been read.

An asterisk “\*” is appended to the state for instances in transition. A question mark “?” is displayed if the state is absent or unrecognized.

# SMF Components



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

When a system is booted, the `init` program reads the `/etc/inittab` file and starts the `svc.startd` daemon, which in turn consults the SMF manifests to gather property and instance information about each service before starting each service and its associated dependents. The SMF uses the Service Configuration Repository (also known as the SMF Repository) to store state and configuration information about each service instance in addition to per-service snapshots that are taken at the time each service is successfully started and used as backups. The SMF repository is managed by the `svc.configd` daemon.



## SMF Profile

- An SMF profile is an XML file that allows customization of services and instances delivered by the system.
- Profiles delivered with the OS include:
  - `/etc/svc/profile/generic_open.xml`
  - `/etc/svc/profile/generic_limited_net.xml`
  - `/etc/svc/profile/ns_*.xml`
  - `/etc/svc/profile/platform_*.xml`

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

An SMF profile is an XML file that allows customization of services and instances delivered by the system. Profiles are available for configuration customization using a file rather than a set of scripts, or to customize configuration at deployment or installation time. All configurations may be customized by using a profile, including adding instances for system-supplied services.

Some profiles that are delivered with the operating system release include:

- `/etc/svc/profile/generic_open.xml`: This profile enables the standard services that have been started by default in earlier releases.
- `/etc/svc/profile/generic_limited_net.xml`: This profile disables many of the Internet services that have been started by default in earlier releases. The `network/ssh` service is enabled to provide network connectivity.
- `/etc/svc/profile/ns_*.xml`: This profile enables services associated with the name service that is configured to run on the system.
- `/etc/svc/profile/platform_*.xml`: This profile enables services associated with particular hardware platforms.

## When Are SMF Profiles Applied?

- The `/etc/svc/profile/generic.xml` profile:
  - Is applied during the first boot after a new installation or an upgrade
  - Is symbolically linked to `generic_open.xml` or `generic_limited_net.xml`
- The content of `site.xml` in `/etc/svc/profile`:
  - Is applied during first boot
  - Is applied in subsequent reboots, if the file is modified

**Note:** Profiles in `/etc/svc/profile` are applied during early manifest import.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

During the first boot after a new installation or an upgrade, the `/etc/svc/profile/generic.xml` profile is applied. This file is usually symbolically linked to `generic_open.xml` or `generic_limited_net.xml`. In addition, if a profile called `site.xml` is in `/etc/svc/profile` during the first boot or is added between boots, the contents of this profile are applied.

By using the `site.xml` profile, the initial set of enabled services may be customized by the administrator.

**Note:** Similar to manifests, profiles in `/etc/svc/profile` are applied during the early manifest import. Profiles in `/var/svc/profile` are applied during the later manifest import.

**Caution:** The `generic_xxx` profiles are mutually exclusive. Any conflicting definitions between files in `/etc/svc/profile/site` are treated as conflicts, and the affected service instances are put into the maintenance state.

## SMF Profile: Example

```
<?xml version='1.0'?>
<!DOCTYPE service_bundle SYSTEM
    '/usr/share/lib/xml/dtd/service_bundle.dtd.1'>
<!--
<header content omitted>
<service_bundle type='profile' name='generic_open'
  xmlns:xi='http://www.w3.org/2003/XInclude' >
  <!--
    Include name service profile, as set by system id tools.
  -->
  <xi:include href='file:/etc/svc/profile/name_service.xml' />

  <!--
    svc.startd(1M) services
  -->
  <service name='system/coreadm' version='1' type='service'>
    <instance name='default' enabled='true' />
  </service>
  <service name='system/cron' version='1' type='service'>
    <instance name='default' enabled='true' />
  </service>
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

This example presents an excerpt from the `/etc/svc/profile/generic_open.xml` file. As discussed, this profile enables the standard services that have been started by default in earlier releases. Each service is listed in the same basic format:

```
<service name='system/coreadm' version='1' type='service'>
  <instance name='default' enabled='true' />
</service>
```

## Service Configuration Repository

- In SMF, state and configuration information about each service instance is stored in the service configuration repository.
- The repository is an SQLite database and is stored in the `/etc/svc/repository.db` file.
- The repository is managed by the `svc.configd` daemon.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered within a solid red rectangular bar.

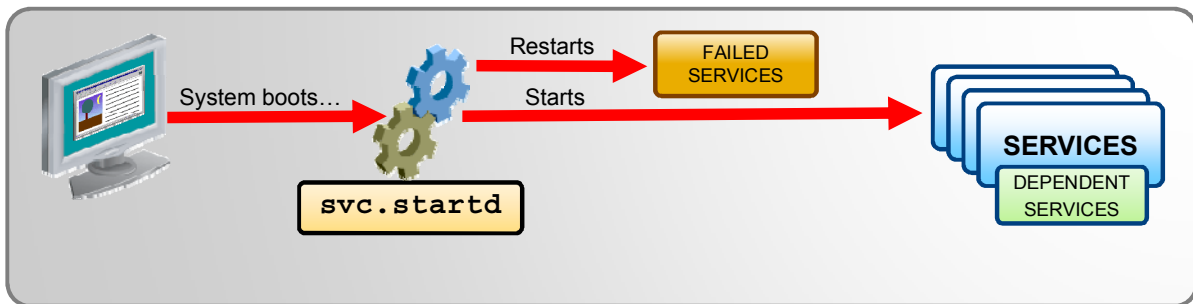
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `svc.configd` daemon is the interface between the repository and the user, and ensures that a consistent picture of the repository is presented to the user.

A service known as the `manifest-import` service takes a backup of the repository during reboot. This backup of the repository ensures that failback is possible.

## SMF Master Restarter Daemon (`svc.startd`)

- The master restarter daemon (`svc.startd`) is the master process starter and restarter and is responsible for:
  - Managing service dependencies for the entire system
  - Ensuring that the system boots properly
  - Starting, restarting, and shutting down services
- The `svc.startd` daemon is also responsible for ensuring that the system boots to the appropriate milestone.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

SMF controls the starting and restarting of services through the master restarter daemon (`svc.startd`). When services are first starting up, the `svc.startd` daemon retrieves service information from the configuration repository and then starts the services when their dependencies have been met. The master restarter daemon is also responsible for restarting failed services and for shutting down services whose dependencies are no longer satisfied.

## Milestone

- A milestone can be regarded as a system state to reach.
- The milestones that can be used at boot time are:
  - none
  - single-user
  - multi-user
  - multi-user-server
  - all
- In addition, the `milestone` subcommand with the `svcadm` command can be used to change the run level of a system by selecting a milestone at which to run.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A milestone is a special service that depends on a set of services to be started before its start is satisfied. Services that depend on the milestone service will have their dependency satisfied and can start after other dependencies are satisfied.

**Note:** If no milestone is specified at boot up, `svc.startd` boots to the built-in milestone `all`, which includes all the system-enabled services.

## SMF Manifests

- An SMF manifest is an XML file that describes information about:
  - The service
  - The service dependencies
  - Any required service configuration
  - How SMF should start and stop the service
- Manifests are imported to load the properties of that service and its instances into the repository.
- The standard location for manifests is `/lib/svc/manifest`.
- Manifests are imported and upgraded during the boot process before any services start.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Caution:** Do not make changes to manifests delivered by Oracle or third-party software vendors. Do not directly edit those manifests in `/lib/svc/manifest` and `/var/svc/manifest`, because any customizations will be lost upon upgrade. Instead, either create a site profile to customize the service, or use the `svccfg` or `inetadm` command to manipulate the properties directly. The `/lib/svc/manifest/site` and `/var/svc/manifest/site` directories are also reserved for site-specific use. The Oracle Solaris release will never deliver manifests into those directories, which ensures that the customizations made by customers are never lost.

## SMF Manifest: Example

```
<?xml version="1.0"?>
<!DOCTYPE service_bundle SYSTEM
    "/usr/share/lib/xml/dtd/service_bundle.dtd.1">
<!--
<header and copyright content omitted>
<service_bundle type='manifest' name='SUNWcsr:rbac'>

  <service
name='system/rbac'
type='service'
version='1'>

    <create_default_instance enabled='true' />

    <single_instance />

    ---
    ---
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

This example presents an excerpt from `rbac.xml` manifest. A manifest file consists of the following basic entries:

- **<service\_bundle type=**: Identifies the name of the service. The type (manifest) indicates a simple service rather than a milestone, the package providing the service, and the service name.
- **<service=**: Identifies service category, type, name, and version
- **<create\_default\_instance=**: Creates the default instance
- **<single\_instance/>**: Identifies whether multiple instances of the service will run
- **<dependency=**: Identifies dependencies for this service
- **<dependent=**: Identifies what service has this service as a dependent
- **<exec\_method=**: Defines how the service is started and stopped
- **<property\_group name=**: Identifies the service model to use
- **<template>**: Creates information to describe the service



## SMF Repository Backups

- SMF automatically takes the following repository backups:
  - Boot backup: Is taken immediately before the first change to the repository is made during each system startup
  - `manifest_import` backups: Is taken after `svc:/system/early-manifest-import:default` or `svc:/system/manifest-import:default` completes
- The system maintains the latest four backups of each type.
- Backups are stored as `/etc/svc/repository-type-YYYYMMDD_HHMMSS` for the date and time when the backup was taken.
- Repository can be restored from these backups.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

SMF automatically takes the following backups of the repository:

- The boot backup is taken immediately before the first change to the repository is made during each system startup.
- The `manifest_import` backups occur after `svc:/system/early-manifest-import:default` or `svc:/system/manifest-import:default` completes, if the service imported any new manifests or ran any upgrade scripts.

Four backups of each type are maintained by the system. The system deletes the oldest backup when necessary. The backups are stored as `/etc/svc/repository-type-YYYYMMDD_HHMMSS`, where `YYYYMMDD` (year, month, day) and `HHMMSS` (hour, minute, second) are the date and time when the backup was taken. Note that the hour format is based on a 24-hour clock.

You can restore the repository from these backups, if an error occurs.

## SMF Repository Snapshots

- Snapshots are taken per service at the time when a service is successfully started.
- Standard snapshots include:
  - `initial`: Is taken on the first import of the manifest
  - `running`: Is taken when the service methods are executed
  - `start`: Is taken at the last successful start
- SMF service always executes with the `running` snapshot.
- Current property values for a service are incorporated into the `running` snapshot with the `svcadm refresh` command.
- Instance configurations can be viewed or reverted to in a previous snapshot by using the `svccfg` command.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered within a solid red rectangular bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The service configuration repository provides a per service snapshot at the time each service is successfully started so that fallback is possible. The standard snapshots that are stored in the SMF repository are listed in the slide.

The SMF service always executes with the `running` snapshot. This snapshot is automatically created if it does not exist.

When you change the property values of a service, the changes are incorporated into the `running` snapshot when you execute the `svcadm refresh` command. You can use the `svccfg` command to view or revert to instance configurations in a previous snapshot.

## Quiz

The booting process in Oracle Solaris 11 is similar to that in Oracle Solaris 10, except that the `init` process reads the `/etc/inittab` file primarily to start the ..... daemon, which in turn is responsible for initiating the rest of the services in Oracle Solaris 11.

- a. `svc.configd`
- b. `svc.startd`
- c. `inetd`
- d. `rc*.d`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

## Lesson Agenda

- Describing the features and capabilities of SMF
- **Administering SMF services**
- Configuring SMF services
- Securing SMF services
- Troubleshooting SMF services

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Administering SMF Services

- When provided with the appropriate Service Management privileges, a system administrator's tasks would include monitoring, modifying, and configuring services.
- SMF provides a set of command-line utilities that interact with SMF and can be used to perform administrative tasks.

Command	Description
svcs	Provides information about services, including their status
svcadm	Manages the state of service instances
svccprop	Retrieves information about service configuration properties
svccfg	Imports, exports, and modifies service configuration
inetadm	Transitions traditional <code>inetd</code> services and manages them in the SMF framework

**Note:** The following slides present some examples of using these SMF commands.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Note that some SMF commands require privilege. Users can gain this privilege by adopting the `root` role or by gaining the `solaris.smf.manage` and/or `solaris.smf.modify` authorizations. You will learn more about privileges in “Lesson 7: Administering Privileges and RBAC.”

## Listing Services Information

- To list all the services currently running on the system, run `svcs`.

```
# svcs
STATE          STIME      FMRI
legacy_run     1:25:08   lrc:/etc/rc2_d/S47pppd
legacy_run     1:25:08   lrc:/etc/rc2_d/S89PRESERVE
online         1:23:48   svc:/system/svc/restarter:default
<output omitted>
```

- To list all services defined on the system, run `svcs -a`.

```
# svcs -a
STATE          STIME      FMRI
legacy_run     1:25:08   lrc:/etc/rc2_d/S47pppd
legacy_run     1:25:08   lrc:/etc/rc2_d/S89PRESERVE
online         11:24:28   svc:/network/physical:default
<output omitted>
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To see what services are currently running on the system (including services that have been disabled temporarily), use the `svcs` command with no arguments, as shown in the first partial example. The `svcs` command displays information about service instances as recorded in the service configuration repository.

As in the first example, the `svcs` command output displays the state of the service, the time the service was started, and the service identifier (FMRI).

To see all the services that have been defined on the system, use the `svcs -a` command, as shown in the second partial example.

## Displaying the Status of a Service Instance

To display the status of a service, run `svcs -l FMRI`.

```
# svcs -l svc:/network/ssh:default
fmri          svc:/network/ssh:default
name          SSH server
enabled       true
state         online
next_state    none
state_time    November 11, 2012 01:25:05 AM UTC
logfile       /var/svc/log/network-ssh:default.log
restarter     svc:/system/svc/restarter:default
contract_id   113
manifest      /etc/svc/profile/generic.xml
manifest      /lib/svc/manifest/network/ssh.xml
dependency    require_all/none svc:/system/filesystem/local (online)
dependency    optional_all/none svc:/system/filesystem/autofs (online)
dependency    require_all/none svc:/network/loopback (online)
dependency    require_all/none svc:/network/physical:default (online)
dependency    require_all/none svc:/system/cryptosvc (online)
dependency    require_all/none svc:/system/utmp (online)
dependency    optional_all/error svc:/network/ipfilter:default (disabled)
dependency    require_all/restart file:///localhost/etc/ssh/sshd_config (online)
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To display the status information for an instance of a service, use the `svcs -l` command, followed by the service identifier, as shown in the example. The `-l` option, when used with the `svcs` command, displays all available information about the selected services and service instances, with one service attribute displayed for each line.

In this example, the status of the `ssh` service is displayed. The output displays:

- The FMRI for the service instance
- Whether the service is enabled or not
- The state of the service (In this case, the service is online.)
- The next state of the service
- The service that is used to restart the service
- The contract ID
- Associated manifests
- A list of service dependencies

## Displaying the Service Dependents and Dependencies

- To display all the services that depend on a specific service, run `svcs -D FMRI`.

```
# svcs -D svc:/network/ssh:default
STATE          STIME          FMRI
online         1:25:05        svc:/milestone/self-assembly-complete:default
online         1:25:09        svc:/milestone/multi-user-server:default
```

- To display all the services a specific service depends upon, run `svcs -d FMRI`.

```
# svcs -d svc:/network/ssh:default
STATE          STIME          FMRI
disabled       1:23:51        svc:/network/ipfilter:default
online         1:24:04        svc:/system/cryptosvc:default
online         1:24:09        svc:/network/loopback:default
online         1:24:11        svc:/system/utmp:default
online         1:24:28        svc:/network/physical:default
online         1:24:36        svc:/system/filesystem/local:default
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If a service fails, the dependent services are affected. Whenever an issue with a service occurs that requires administrative intervention (such as taking the service down for maintenance), one of the first actions that is taken is to see what dependents that service has. To determine which service instances depend on another service, use the `svcs -D` command followed by the service identifier, as shown in the example.

To understand why a particular service is not running, it is helpful to know the dependencies that the service has. To determine the services on which a service instance depends, use the `svcs -d` command followed by the service identifier, as shown in the second example.



## Disabling a Service

1. Identify service dependants by using `svcs -D FMRI`.
2. To disable the service, use `svcadm disable FMRI`.

```
# svcadm disable svc:/network/ssh:default
```

3. To verify that the service has been disabled, use `svcs -l FMRI`.

```
# svcs -l svc:/network/ssh:default
fmri          svc:/network/ssh:default
name          SSH server
enabled       false
state         disabled
<output omitted>
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The only reason to specifically disable and then enable a service is if changes need to be made before the service is enabled and after the service is disabled.

To stop a running service and prevent it from restarting, you must disable it. To disable a service, perform the steps outlined in the slide.

**Notes for step 1:** If this service has dependents that you need, you should not disable the service.

**Notes for step 2:** In this example, the SSH server is being disabled.

**Notes for step 3:** The enabled status has changed from `true` to `false`, indicating that the service is currently disabled.

When a service is disabled, the service status change is recorded in the service configuration repository. The disabled state persists across reboots. This means that the only way to get the service running again is to enable it.

**Note:** You can also temporarily disable a service with the `-t` option. With this option, the service returns to an `online` state on reboot.

## Enabling a Service

1. To determine whether service dependencies are satisfied, use `svcs -l FMRI/grep online`.
2. To enable the service, use `svcadm enable FMRI`.

```
# svcadm enable svc:/network/ssh:default
```

3. To verify that the service has been enabled, use `svcs FMRI`.

```
# svcs svc:/network/ssh:default
STATE          STIME          FMRI
online         1:24:28      svc:/network/ssh:default
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To enable a service, perform the steps outlined in the slide.

**Notes for step 1:** If the service is online, the service dependencies are satisfied. If the service is not online, use `svcadm enable -r FMRI` to recursively enable all dependencies.

**Notes for step 2:** In this example, the SSH service is being enabled.

**Notes for step 3:** The state should read `online`.

Again, the service status change is recorded in the service configuration repository. The enabled state persists across system reboots as long as the service dependencies are met.

## Refreshing and Restarting a Service

- To refresh a service, run `svcadm refresh FMRI`.

```
# svcadm refresh svc:/network/ssh:default
```

- To restart a service, run `svcadm restart FMRI`.

```
# svcadm restart svc:/network/ssh:default
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If a service is currently running but needs to be restarted due to a configuration change or some other reason, the service can be restarted without you having to enter separate commands to stop and start the service. Some configuration changes require only a refresh, while others require a restart.

- To refresh a service, use the `svcsadm refresh` command followed by the service identifier. In the first example, the `ssh` service is refreshed.
- To restart a service, use the `svcsadm restart` command followed by the service identifier. In the second example, the `ssh` service is restarted.

**Note:** When a refresh is done, the running snapshot is taken/updated. After this, properties can be queried from that snapshot to get a consistent picture. For example, if a service needs two properties to determine behavior, those two properties can be set individually and then refreshed into the service's running environment.

## Managing SMF Services Properties

To list and modify service properties, use the `svccprop`, `svccfg`, and `svcadm` commands.

```
# svccprop -p config/nodename identity:node  
sol11  
# svccfg -s identity:node setprop config/nodename=solaris11  
# svcadm refresh identity:node  
# svcadm restart identity:node  
# svccprop -p config/nodename identity:node  
# svccprop -p config/nodename identity:node  
solaris11
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The example in the slide captures the `svccprop`, `svcadm`, and `svccfg` commands as they are used to display and modify the `nodename` property from `sol11` to `solaris11` of the `identity:node` service.

## Modifying `inetd` Service Properties

- To modify a property of the `inetd` daemon, use the `inetadm` command.

```
# inetadm -M tcp_trace=<TRUE/FALSE>
```

- To modify a property of a particular service managed by `inetd` daemon, use `inetadm -m FMRI`.

```
# inetadm -m network/telnet tcp_trace=<TRUE/FALSE>
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

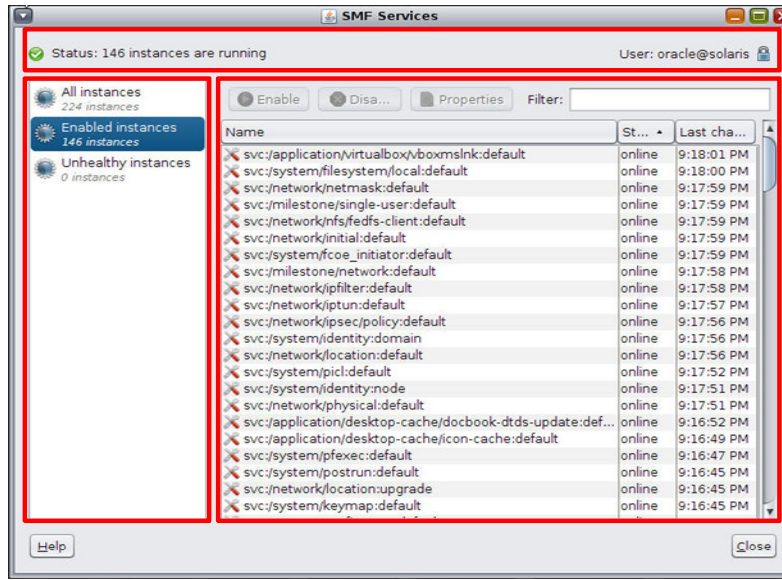
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the first example, the `tcp_trace` property is enabled (`TRUE`) to track the activities of the `inetd` daemon and in turn of all the services that are managed by the `inetd` daemon. This means that whenever all the services managed by `inetd` daemon are invoked, an appropriate entry is made in the log file mentioned against the `daemon.notice` field of the `/etc/syslog.conf` file.

In the second example, the `tcp_trace` property is enabled for an individual service (`network/telnet`) by using the `-m` option of the `inetadm` command. This gives more granular control on the service.

# Managing SMF Services by Using the GUI

Apart from using the CLI, you can use the SMF GUI to monitor, configure, and perform basic administration of SMF services or service instances on a local or remote system.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can access the SMF GUI by using the following menu options from the desktop of Oracle Solaris 11 or later systems: **System > Administration > SMF Service**.

The screenshot in the slide shows the default **SMF Services** window, which is divided into the following three panels:

- **Top area:** Consists of the following two components:
  - The **Status** field on the left shows the status of the services running on the local host.
  - The **User** field on the right shows the credential that is used by the tool. You can use this field to change the credentials.
- **Left panel:** Enables you to select the group of services that you want displayed
- **Right panel:** Displays a list of all the services in the group you select on the left panel

## Quiz

You have been asked to change the nameserver (dns/client) information of your system to 192.168.72.201. Which of the following SMF commands would you use?

- a. `svcs dns/client`
- b. `svccfg -s dns/client setprop  
config/nameserver=192.168.72.201`
- c. `svcprop -p config/nameserver dns/client`
- d. `inetadm -m dns/client  
config/nameserver=192.168.72.201`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

## Lesson Agenda

- Describing the features and capabilities of SMF
- Administering SMF services
- **Configuring SMF services**
- Securing SMF services
- Troubleshooting SMF services

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Configuring SMF Services

Configuring SMF services involves the following activities:

- Creating a service
- Modifying a service's manifest
- Changing an environment variable for a service
- Changing a property for an `inetd`-controlled service
- Creating and applying an SMF profile
- Changing services and their configurations by using the `netservices` command

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Creating a Service

Perform the following steps to create a service:

1. Create the script by using the following command:  
`vi /usr/local/svc/method/servicename`
2. Grant the execute permission on the script so it can be executed by using the following command:  
`chmod 544 /usr/local/svc/method/servicename`
3. Change directories to `/lib/svc/manifest/site` and edit the manifest `.xml` file for your new service.
4. Import the new service into the SMF by using the following command: `svccfg import \`  
`/lib/svc/manifest/site/servicename.xml`
5. Verify that the new service is available by using the `svcs servicename` command.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note for step 3:** An explanation of each of the entries in the file is provided on the following pages.

**Note for step 4:** When using the default manifest, `/lib/svc/manifest`, use the `import` command as shown in this step; otherwise, use the `manifest-import` command.

## Creating a Service: Example

```
# vi /usr/local/svc/method/newservice
#!/sbin/sh
#
# ident "@(#)newservice 1.14 04/08/30 SMI"
case "$1" in
'start')
/usr/bin/newservice &
;;
'stop')
/usr/bin/pkill -x -u 0 newservice
;;
*)
echo "Usage: $0 { start | stop }"
;;
esac
exit 0
# chmod 544 /usr/local/svc/method/newservice
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example shown in the slide, a new service called `newservice` is created. This is followed by steps for editing the new service script and granting execute permissions.

## Creating a Service: Example

```
# cd /var/svc/manifest/site
# vi newservice.xml
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE service_bundle SYSTEM
    '/usr/share/lib/xml/dtd/service_bundle.dtd.1'>
<service_bundle type='manifest' name='OPTnew:newservice'>
  <service name='site/newservice' type='service' version='1'>
    <create_default_instance enabled='true'/>
    <single_instance/>
    <exec_method name='start' type='method'
      exec='/usr/local/svc/method/newservice start'
      timeout_seconds='30'>
    </exec_method>
    <exec_method name='stop' type='method' exec=':true'
      timeout_seconds='30'>
    </exec_method>
    <property_group name='startd' type='framework'>
      <propval name='duration' type='astring' value='transient'/>
    </property_group>
  </service>
</service_bundle>
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Here you see how to change directories to `/var/svc/manifest/site` and edit the `manifest.xml` file entries for the new service. Take a closer look at each of the entries in the file. To begin, there is the standard header:

```
<?xml version='1.0' encoding='UTF-8' ?>
<!DOCTYPE service_bundle SYSTEM
    '/usr/share/lib/xml/dtd/service_bundle.dtd.1'>
```

Just below the header is the name of the service. The type (manifest) indicates a simple service rather than a milestone, the package providing the service, and the service name.

```
<service_bundle type='manifest' name='OPTnew:newservice'>
```

Next are the service category, type, name, and version.

```
<service name='site/newservice' type='service' version='1'>
```

The next entry creates the instance and the entry below that specifies whether multiple instances of the service will run.

```
<create_default_instance enabled='true'/>
<single_instance/>
```

The next entry is how the service is started and stopped.

```
<exec_method name='start' type='method'
exec='/usr/local/svc/method/newservice start'
timeout_seconds='30'>
</exec_method>
<exec_method name='stop' type='method' exec=':true'
timeout_seconds='30'>
</exec_method>
```

This is followed by the service model to use. The entry shows that the service will be started by `svc.startd`. Transient services are not continuously running services.

```
<property_group name='startd' type='framework'>
  <propval name='duration' type='astring'
value='transient' />
</property_group>
</service>
</service_bundle>
```

**Note:** If you need to define dependencies for the service, you can do so by using the following entry:

```
<dependent
  name='newservice'
  grouping='require_all'
  restart_on='none'>
  <service_fmri value='svc:/milestone/multi-user' />
</dependent>
```

This example ensures that the service is associated with the multiuser milestone and that the multiuser milestone requires this service.

After editing the manifest file and reviewing to ensure that no XML tags are missing, it is a good practice to validate the file by running the following command:

```
# svccfg validate /var/svc/manifest/site/newservice.xml
```

## Creating a Service: Example

```
# svccfg import /var/svc/manifest/site/newservice.xml
svccfg: Taking "previous" snapshot for svc:/site/newservice:default.
svccfg: Upgrading properties of svc:/site/newservice according to
       instance "default".
svccfg: svc:/site/newservice: Deleting property
       "general/entity_stability".
svccfg: svc:/site/newservice: Upgrading property "stop/exec".
svccfg: svc:/site/newservice: Deleting property group "tm_common_name".
svccfg: svc:/site/newservice: Deleting property group "tm_man_utmpd1M".
svccfg: svc:/site/newservice: Deleting property group "tm_man_utmpx4".
svccfg: Taking "last-import" snapshot for svc:/site/newservice:default.
svccfg: Refreshed svc:/site/newservice:default.
svccfg: Successful import.
# svcs newservice
STATE STIME FMRI
online 8:43:45 svc:/site/newservice:default
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Here the service is being imported into the SMF by using the `svccfg import` command. Know that SMF creates a snapshot of this service to be stored in the service configuration repository.

After the service has been imported into SMF, the final step is to verify that it is visible to the system by using the `svcs` command. Observe that the service is `online`.

## Creating a Service by Using `svcbundle`

- With the introduction of `svcbundle` in Oracle Solaris 11.1, the creation of manifests and profiles is easier.
- `svcbundle` enables you to take advantage of the benefits of automatic application restart without requiring you to have full knowledge of the XML file format that is used when integrating with SMF.
- You can use the `svcbundle` command to generate SMF manifests that can then be validated by using the `svccfg` command.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For more information, refer to <http://www.oracle.com/technetwork/articles/servers-storage-admin/howto-svcbundle-manifest-profile-1866525.html>, [http://docs.oracle.com/cd/E26502\\_01/html/E29003/eqbrs.html#smft-5](http://docs.oracle.com/cd/E26502_01/html/E29003/eqbrs.html#smft-5) and `svcbundle` (1M).

## Modifying a Service's Manifest

- There might be times when you need to modify a service's manifest due to structural changes that impact the execution method.
- To change the configuration of a service that is not managed by the `inetd` service, use the steps listed below:
  1. Modify the manifest.
  2. Re-import the manifest with `svcadm restart manifest-import` if in the standard location. If not in the standard location, run `svccfg import <manifest>`.
  3. Restart may be required although importing the service will refresh it.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Notes for step 1:** Many of the services have one or more configuration files that are used to define the startup or other configuration information. These files can be changed while the service is running. The contents of the files are checked only when the service is started.

**Notes for step 2:** The `svcadm` utility enables you to perform common service management tasks, such as enabling, disabling, or restarting service instances.



## Changing an Environment Variable of a Service

Perform the following steps to change an environment variable of a service:

1. Verify that the service is running by using `svcs FMRI`.
2. Set environment variables by using `svccfg -s FMRI setenv envar value`.
3. Refresh the service by using `svcadm refresh FMRI`.
4. Restart the service by using `svcadm restart FMRI`.
5. Verify that the change has been made by using `pargs -e `pgrep -f /usr/sbin/FMRI``.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Notes for step 1:** The `svcs` utility provides detailed views of the service state of all service instances in the service configuration repository.

**Notes for step 2:** The `-s` option selects the entity indicated by the FMRI before executing any subcommands. The modification subcommand `setenv` searches for the `start` property group in the currently selected entity and, if an instance is currently selected, its parent is also searched. After the property is located, all values that begin with `envvar` followed by a “=” are removed, and the value “`envvar=value`” is added.

**Notes for step 3:** The `svcadm` command is used to manipulate service instances. The command issues requests for actions on services executing within SMF. Actions for a service are carried out by its assigned service restarter agent. The `refresh` subcommand requests that the assigned restarter update the service’s running configuration snapshot with the values from the current configuration. Some of these values take effect immediately (for example, dependency changes). Other values do not take effect until the next service restart.

**Notes for step 5:** The `pargs -e` command prints the parameter arguments and environment variables that have been passed to the service.

## Changing a Property for an `inetd`-Controlled Service

If you need to impose more access controls on a particular Internet service, you can do so by modifying the service's property settings.

1. List the properties for the specific service by using `inetadm -l FMRI`.
2. Change the property for the service by using `inetadm -m FMRI property-name=value`.
3. Verify that the property has changed by using `inetadm -l FMRI`.
4. Confirm that the change has taken effect.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Notes for step 1:** The `inetadm` command enables you to observe or configure services controlled by `inetd`, which is the delegated restarter for Internet services for SMF. Its primary responsibilities are to manage service states in response to administrative requests, system failures, and service failures and, when appropriate, to listen for network requests for services.

The `inetadm -l` command displays all the properties for the service identified by the FMRI.

**Notes for step 2:** The `-m` option is used to change the values of the specified properties of the identified service instances. Each property for an `inetd`-controlled service is defined by a property name and an assigned value. Supplying the property name without a specified value resets the property to the default value.

## Creating and Applying an SMF Profile

- You can create an SMF profile that reflects the services that you want enabled or disabled on the current system.
- Perform the following steps to create an SMF profile:
  1. Create a profile by using `svccfg extract> profile.xml`.
  2. Edit the `profile.xml` file to make any required changes.
    - a. Change the name of the profile in the `service_bundle` declaration.
    - b. Remove any services that should not be managed by this profile.
    - c. Add any services that should be managed by this profile.
    - d. Change the enabled flag for selected services, if necessary.
  3. When required, apply the new profile by using `svccfg apply profile.xml`.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** Not all services need to be listed in a profile. Each profile needs to include only those services that need to be enabled or disabled to make the profile useful.

**Notes for step 1:** The `svccfg` utility enables you to display and manipulate the contents of the service configuration repository. The service profile subcommand `extract` prints a service profile that represents the enabled status of the service instances in the repository to standard output. You can redirect the output to a file by using `extract>` as is being done in step 1.

**Notes for step 2b:** For each service, remove the three lines that describe the service. Each service description starts with `<service` and ends with `</service>`.

**Notes for step 2c:** Each service needs to be defined by using the three-line syntax shown here:

```
<service name='network/ldap/client' version='1' type='service'>
  <instance name='default' enabled='true' />
</service>
```

**Notes for step 3:** Applying the service profile subcommand takes the properties, including general/enabled, that are specified in the file and modifies them in the SMF repository.

## Changing Services and Their Configurations by Using the `net services` Command

The `net services` command switches system services between open/traditional network exposure or limited network exposure.

- For open or traditional network exposure, run:  
`/usr/sbin/net services open`
- For limited network exposure, run:  
`/usr/sbin/net services limited`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `net services` command switches system services between open/traditional network exposure and limited network exposure. The switch is done with the `generic_limited.xml` and `generic_open.xml` profiles. In addition, some service properties are changed by the command to limit some services to a local-only mode or to the traditional mode, as appropriate.

**Note:** The `generic_limited_net` profile and the local-only mode service properties are applied by default.

- To have open or traditional network exposure, use the `/usr/sbin/net services open` command.
- To have limited network exposure, use the `/usr/sbin/net services limited` command. This command changes properties to run some services in local mode, as well as restricts the enabling of services with the `generic_limited_net` profile. The command should be used only if the `generic_open.xml` profile is applied.

# Setting Up Service State Transition Notifications

Perform the following steps to set up email notification:

1. Installing the `smtp-notify` package
2. Enabling the `smtp-notify:default` service
3. Configuring service state transition notifications

Monitored Transition States	
to-uninitialized	to-disabled
from-uninitialized	from-disabled
to-maintenance	to-online
from-maintenance	from-online
to-offline	to-degraded
from-offline	from-degraded

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

SMF supports an email notification feature that sends information about service state transitions and fault management events. The notification feature monitors the transition states presented in the table in the slide and uses a service called Simple Mail Transfer Protocol (SMTP) to send the email notification when a service changes states. This feature enables you to respond quickly to any changes in service states that might require immediate resolution.

To use the notifications feature, you must ensure that the `smtp-notify` package is installed, enable the service that controls the notification feature, and then configure the notifications.

# Setting Up Service State Transition Notifications

1. To install the SMF notification feature, run `pkg install`.

```
# pkg install system/fault-management/smtp-notify
```

2. To enable the SMF notification service, run `svcadm enable`.

```
# svcadm enable
# svc:/system/fm/smtp-notify:default
```

3. To configure notifications for all services and a single service, run the following commands, respectively.

```
# svccfg -s svc:/system/svc/global:default setnotify -g \
to-maintenance mailto:root@localhost
```

```
# svccfg -s svc:/network/http:apache22 setnotify \
to-maintenance mailto:root@localhost
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

1. Before you can use the SMF notification feature, ensure that the `smtp-notify` package is already installed on your system. The `smtp-notify` package contains the notification functionality and the daemon that controls it.  
To verify that the `system/fault-management/smtp-notify` package is installed, run the `pkg info system/fault-management/smtp-notify` command. If the package is not installed, install it by using the `pkg install` command, as shown in the example.
2. Next, you need to enable the service that controls the notification feature's daemon. Use the `svcadm enable` command, as shown in the second example.
3. The final step is to configure the notifications. You can configure notifications for all services or a single service.  
In the first example, a notification is set to occur if any service state changes from the `online` state to the `to-maintenance` state.  
In the second example, a notification is being set to alert you if the `apache22` service state changes from `online` to the `to-maintenance` state.

## Managing Notifications

- To view configured notifications, run the `svccfg listnotify` command.

```
# svccfg -s svc:/system/svc/global:default listnotify

Event: from-online (source: svc:/system/svc/global:default)
Notification Type: smtp
Active: true
to: root@localhost
```

- To stop all notifications, run the `svccfg delnotify` command.

```
# svccfg -s svc:/system/svc/global:default delnotify -g all
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can view the notifications that are configured for a system by using the `svccfg listnotify` subcommand, as shown in the first example. Here, you can see that a notification has been configured for all services to be sent to you if there is a service state change from `online` to any other state.

If you want to stop service state transition notifications, you can do so by using the `delnotify` subcommand, as shown in the second example.

## Lesson Agenda

- Describing the features and capabilities of SMF
- Administering SMF services
- Configuring SMF services
- **Securing SMF services**
- Troubleshooting SMF services

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Least Privilege and SMF

- The principle of least privilege requires that any user be given no more privilege than is required to do the job.
- Running a service with reduced privileges reduces the potential harm that a service can cause if it were to behave abnormally due to a bug, an accident, or even a malicious exploit.

**Note:** You will learn more about least privilege, rights profiles, and role-based access control (RBAC) in “Lesson 7: Administering Privileges and RBAC.”

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Service Privileges

Every service has four sets of privileges that determine whether a service can use a particular privilege.

- Services started thus have all privileges assigned to their effective ( $\mathbb{E}$ ), permitted ( $\mathbb{P}$ ), and limit ( $\mathbb{L}$ ) privilege sets.
- The basic set of privileges, available to all users by default, is assigned to the service's inherited ( $\mathbb{I}$ ) privilege set.
  - The first process spawned by SMF for a given service has essentially all privileges.
  - Any child process created by the parent process runs only with the basic set of privileges.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Many services simply do not need to be started with this much privilege, and doing so exposes system resources to potential risks that are easily prevented. By restricting how a service is started, you are in a better position to contain its set of available privileges, and also possibly eliminate the need for the service to ever start as `root`.

Further, if a service is never started as `root`, or if it never has the set of privileges of `root` available to it, then the window of exposure and criticality of impact is significantly reduced in the event that a flaw is found in that service. Essentially, running with reduced privileges is recommended because it reduces the potential harm that a service can cause if it were to behave abnormally due to a bug, accident, or even a malicious exploit.

For example, if a service is not running with the `proc_fork` and `proc_exec` privileges, it is unable to use those system calls. This is important in the event that an exploitable buffer overflow is found in that service, because an attacker would not easily be able to exploit that vulnerability to execute arbitrary code.

## SMF Rights Profile

- Using SMF rights profiles, it is possible to delegate access to core service management functions based on the concept of least privilege.
- If a user or service does not strictly need to have some degree of privilege, that privilege should not be granted.
- Oracle Solaris contains two primary SMF rights profiles:
  - Service Management: Enables manipulation of any service in any way
  - Service Operator: Provides the ability to enable or disable any service instance on the system, as well as to request that its restart or refresh method be executed

The Oracle logo, consisting of the word "ORACLE" in white capital letters on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Rights profiles are defined in the `/etc/security/prof_attr` database. The following is a list of the profile name and description for SMF-related rights profiles that are included in the Oracle Solaris 11 OS:

- Cron Management: Manage `at` and `cron` jobs
- File System Management: Manage, mount, and share file systems
- Inetd Management: Manage the `inetd` configuration parameters
- Mail Management: Manage `sendmail` and queues
- Maintenance and Repair: Maintain and repair a system
- Network IPsec Management: Manage IPsec and IKE
- Network Management: Manage the host and network configuration
- Network Security: Manage network and host security
- Process Management: Manage current processes and processors
- Service Management: Manage services
- Service Operator: Administer services
- Software Installation: Add application software to the system

## Authorizations and Rights

- SMF uses RBAC authorizations to limit the administration of a service facility.
- The authorizations for SMF are under the heading `solaris.smf` in the `/etc/security/auth_attr` database.
- The three primary SMF authorizations are:
  - `solaris.smf.manage`
  - `solaris.smf.modify`
  - `solaris.smf.value`

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The three primary SMF authorizations are:

- `solaris.smf.manage`: This authorization is typically reserved for service operators who are not given the authority to add or remove services, or to alter the way in which services are configured.
- `solaris.smf.modify`: This authorization is typically reserved for service administrators. Be sparing in how this authorization is granted, because it can be used to facilitate a privilege escalation attack by allowing a user (who has been granted this authorization) to effectively run arbitrary commands as `root` (or as any other user) by simply manipulating a service's configuration.
- `solaris.smf.value`: This authorization does not let you request state modifications of a service.

Note that many of the authorizations in the `etc/security/auth_attr` file are service-instance specific. With these authorizations, it is possible to delegate access to core service management functions based on the principle of least privilege.

## Service-Specific Property Groups

SMF provides a greater level of access control through service-specific property group authorizations:

- `action_authorization`: Permits a user to perform a state modification request for a given service
- `modify_authorization`: Permits the addition, deletion, or modification of properties within the property group
- `value_authorization`: Permits changing or retrieval of any property values of the property group except `modify_authorization`
- `read_authorization`: Permits the retrieval of property values within the property group

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

SMF provides a greater level of access control through the use of service-specific property group authorizations, which can be defined in specific property groups of a service. This technique can be used to further limit which property groups within which services can be created, modified, or deleted. To use service-specific property group authorizations, create one or more of the following properties by using the `svccfg` command, under the property group of the service to which access should be controlled.

The authorization type to use depends on the level of access that should be provided. For example, the following code captures the property group authorization for the `svc:/network/ntp:default` service.

```
<property_group name='general' type='framework'>
  <!-- to start stop ntpd -->
  <propval name='action_authorization' type='astring'
    value='solaris.smf.manage.ntp' />
  <propval name='value_authorization' type='astring'
    value='solaris.smf.value.ntp' />
</property_group>
```

## Quiz

The `/etc/security/auth_attr` file in Oracle Solaris 11.1 contains the authorization for managing SMF services?

- a. True
- b. False

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: a**

## Lesson Agenda

- Describing the features and capabilities of SMF
- Administering SMF services
- Configuring SMF services
- Securing SMF services
- Troubleshooting SMF services

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Troubleshooting SMF Services

Some troubleshooting situations include the following:

- Debugging a service that is not starting
- Restoring a service in maintenance state
- Reverting to an SMF snapshot
- Repairing a corrupt repository
- Debugging the services during a system boot
- Addressing `system/filesystem/local:default` service failures during boot

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Debugging a Service That is Not Starting

**Situation:** You have a service that is disabled and not starting. Debug the situation by using the following steps:

1. Request information about the hung service by using `svcs -xv servicename`.
2. Enable the service by using `svcadm enable serviceinstance`.
3. Verify that the service is online by using `svcs -a servicename`.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If you have a service that is disabled and not starting, you can debug it by using the steps shown in the slide.

**Notes for step 1:** The `-xv` option provides additional information about the service instances that are affected.

In the example, the `print` service is disabled. To find out more about the problem, run the `svcs -xv` command for the service. The output for the `svcs -xv` command provides the following information:

- **State:** The state of the service and the date and time stamp
- **Reason:** Why the service is disabled
- **See:** The URL to a knowledge article on the issue
- **See:** Man page references to help resolve the issue
- **Impact:** What services have been affected by the problem

## Debugging a Service That is Not Starting

```
# svcs -xv
svc:/application/print/server:default (LP Print Service)
  State: disabled since Thu 14 Nov 2013 02:20:37 PM PDT
Reason: Disabled by an administrator.
  See: http://sun.com/msg/SMF-8000-05
  See: man -M /usr/share/man -s 1M lpsched
Impact: 2 services are not running:
        svc:/application/print/rfc1179:default
        svc:/application/print/ipp-listener:default
# svcadm enable application/print/server
# svcs printer
online          11:06:14 svc:/application/print/server:default
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For example, the printer service appears disabled. Note that the disabled printer service is impacting two other services. The problem is corrected by enabling the service. The `svcs <servicename>` command helps verify if the service is back online.

## Restoring a Service in the Maintenance State

**Situation:** SMF places the `time-slider: default` service in the maintenance mode when it is unable to bring it up.

```
# svcs time-slider:default
STATE          STIME          FMRI
maintenance    8:22:10      svc:/application/time-slider:default
```

Perform the following steps to restore the service:

1. Determine whether any processes that are dependent on the service have not stopped by using `svcs -p FMRI`.
2. Kill any remaining processes as required by using `kill -9 PID`.
3. Repair the service configuration by using `svcs -x FMRI`, if necessary.
4. Restore the service by using `svcadm clear FMRI`.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

SMF places a service in maintenance mode when it is unable to bring it up. The steps for restoring a service in maintenance state are shown in the slide.

**Note for step 1:** Normally, when a service instance is in maintenance state, all processes associated with that instance are stopped. However, you should make sure before you proceed. The `svcs -p FMRI` command lists all the processes that are associated with a service instance as well as the PIDs for those processes.

**Note for step 2:** Repeat this step for all processes that are displayed by the `svcs` command.

**Note for step 3:** The `-x` option provides you with details that you might find useful for debugging the issue. You can also examine the appropriate service log files in `/var/svc/log` for a list of errors.

## Restoring a Service in the Maintenance State: Example

```
# svcs -p time-slider:default
STATE          STIME      FMRI
maintenance    8:23:06   svc:/application/time-slider:default
# svcs -x time-slider:default
svc:/application/time-slider:default (GNOME Desktop Snapshot
Management Service)
State: maintenance since Dec 15, 2011 08:22:41 AM MDT
Reason: Start method exited with $SMF_EXIT_ERR_FATAL.
See: http://sun.com/msg/SMF-8000-KS
See: zfs(1M)
See: /var/svc/log/application-time-slider:default.log
Impact: This service is not running.
# svccfg delete time-slider:default
# svcadm refresh time-slider:default
# svcadm enable time-slider:default
# svcadm clear time-slider:default
# svcs time-slider:default
STATE          STIME      FMRI
online          9:37:52   svc:/application/time-slider:default
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example shown in the slide, the `time-slider: default` service is in the maintenance state. As a first step, determine whether any processes that are dependent on the service have not stopped by using the `svcs -p` command. As the example indicates, no dependent processes are listed, so the next step is to repair the service by using the `svcs -x` command. The output from this command indicates that there is an issue with the `start` method. You can examine the log for further details.

Now, determine what in the execution method configuration in the `time-slide.xml` manifest file is causing the problem. However, before doing that, delete the corrupted service by using the `svccfg delete` command.

Then, assuming you opened the `time-slider.xml` manifest file, found the problem with the `start` method, addressed it, and imported the file into SMF, you are ready to bring the service back up. To do this, first refresh the service by using the `svcadm refresh` command to ensure that SMF is reading the new service manifest file, enable the service, and then restore the service by using the `svcadm clear` command. Finally, verify that the service is back online. If it is, the service has been successfully restored.

## Reverting to an SMF Snapshot

**Situation:** If the service's administrative customizations are wrong, you can address the problem by reverting to the last snapshot that started successfully. Perform the following steps to revert to a previous SMF snapshot:

1. Run the `svccfg` command.
  - a. Select the service instance that you want to address.
  - b. Generate a list of available snapshots by using `listsnap`.
  - c. Revert to the `start` snapshot by using `revert start`.
  - d. Quit `svccfg` by using `quit`.
2. Update the information in the service configuration repository by using `svcadm refresh FMRI`.
3. Restart the service instance by using `svcadm restart FMRI`.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If the service's administrative customizations are wrong, you can address the problem by reverting to the last snapshot that started successfully. The steps for reverting to a previous SMF snapshot are shown in the slide.

**Note for step 1a:** You must use an FMRI that fully defines the instance. Shortcuts are not allowed.

**Note for step 1c:** The `start` snapshot is the last snapshot in which the service successfully started.

**Note for step 2:** This step updates the repository with the configuration information from the `start` snapshot.

**Note:** None of the file-backed properties (that is, properties delivered via manifests or profiles) from the snapshot are restored. Instead, all the administrative customizations in the current configuration are removed, and then all the administrative customizations from the selected snapshot are propagated forward.

## Reverting to an SMF Snapshot: Example

```
# svccfg
svc:> select system/console-login:default
svc:/system/console-login:default> listsnap
initial
last-import
previous
running
start
svc:/system/console-login:default> revert start
svc:/system/console-login:default> quit
# svcadm refresh system/console-login:default
# svcadm restart system/console-login:default
# svcs console-login:default
online 18:15:32 svc:/system/console-login:default
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example shown in the slide, it is assumed that the `console-login:default` service is in the `maintenance` state. To resolve the issue, you can revert to a previous SMF snapshot, in this case the `start` snapshot, to bring the service back online.

**Note:** The version of the snapshot you choose to use is based on what you are trying to accomplish. When you have selected the type of snapshot you want, you quit the service configuration. You then refresh and restart the service. Your final step is to verify that the service is back online.

## Repairing a Corrupt Repository

**Situation:** When the repository daemon, `svc.configd`, is started, it does an integrity check of the configuration repository. If the integrity check fails, the `svc.configd` daemon writes a message to the console, then exits and starts `sulogin` to enable you to perform maintenance.

```
<MESSAGE DISPLAYED BY SMF>
svc.configd: smf(5) database integrity check of:
/etc/svc/repository.db
failed. The database might be damaged or a media error might have
prevented it from being verified. Additional information useful to
your service provider is in:
/etc/svc/volatile/db_errors
```

```
The system will not be able to boot until you have restored a working
database. svc.startd(1M) will provide a sulogin(1M) prompt for
recovery purposes. The command:
/lib/svc/bin/restore_repository
can be run to restore a backup version of your repository. See
http://sun.com/msg/SMF-8000-MY for more information.
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Repairing a Corrupt Repository

Perform the following steps to repair a corrupt repository:

1. Enter the `root` password at the `sulogin` prompt.
2. Run the following command:  
`/lib/svc/bin/restore_repository`
3. Enter the appropriate response for the backup option.
4. Enter `yes` to remedy the fault.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Notes for step 1:** `sulogin` enables the `root` user to enter the maintenance mode to repair the system.

**Notes for step 2:** Running this command takes you through the necessary steps to restore a non-corrupt backup. SMF automatically takes backups of the repository at key system moments. When started, the `/lib/svc/bin/restore_repository` command displays a message similar to the following:

Repository Restore utility

See <http://sun.com/msg/SMF-8000-MY> for more information on the use of this script to restore backup copies of the `smf(5)` repository.

If there are any problems that need human intervention, this script gives instructions and then exits back to your shell.

Note that upon full completion of this script, the system will be rebooted using `reboot(1M)`, which will interrupt any active services.



If the system that you are recovering is not a local zone, the script explains how to remount the `/` and `/usr` file systems with read and write permissions to recover the databases. The script exits after printing these instructions. Follow the instructions, paying special attention to any errors that might occur.

After the root (`/`) file system is mounted with write permissions, or if the system is a local zone, you are prompted to select the repository backup to restore, as follows:

The following backups of `/etc/svc/repository.db` exists, from oldest to newest:

*... list of backups ...*

Backups are given names, based on type and the time the backup was taken. Backups beginning with `boot` are completed before the first change is made to the repository after system boot. Backups beginning with `manifest_import` are completed after `svc:/system/manifest-import:default` finishes its process. The time of the backup is given in the `YYYYMMDD_HHMMSS` format.

**Notes for step 3:** Typically, you will select the most recent backup option. The list of options is as follows:

Please enter one of:

- 1) `boot`, for the most recent post-boot backup
- 2) `manifest_import`, for the most recent `manifest_import` backup.
- 3) a specific backup repository from the above list
- 4) `-seed-`, the initial starting repository. (All customizations will be lost.)
- 5) `-quit-`, to cancel.

Enter response [`boot`]:

If you press **Enter** without specifying a backup to restore, the default response, enclosed in `[ ]` is selected. Selecting `-quit-` exits the `restore_repository` script, returning you to your shell prompt.

Selecting `-seed-` restores the `seed` repository. This repository is designed for use during initial installation and upgrades. Using the `seed` repository for recovery purposes should be the last resort.

After the backup to restore has been selected, it is validated and its integrity is checked. If there are any problems, the `restore_repository` command prints error messages and prompts you for another selection.

When a valid backup is selected, the following information is printed and you are prompted for final confirmation.

After confirmation, the following steps will be taken:

```
svc.startd(1M) and svc.configd(1M) will be quiesced, if running.  
/etc/svc/repository.db  
    -- renamed --> /etc/svc/repository.db_old_YYYYMMDD_HHMMSS  
/etc/svc/volatile/db_errors  
    -- copied --> /etc/svc/repository.db_old_YYYYMMDD_HHMMSS_errors  
repository_to_restore  
    -- copied --> /etc/svc/repository.db  
and the system will be rebooted with reboot(1M).
```

Proceed [yes/no]?

**Notes for step 4:** The system reboots after the `restore_repository` command executes all the listed actions.

# Repairing a Corrupt Repository: Example

```
# cd /lib/svc/bin
# ./restore_repository
<output omitted>

The following backups of /etc/svc/repository.db exist, from oldest to
newest:
manifest_import-20111215_035411
boot-20111214_124026
boot-20111215_150206

Please enter either a specific backup repository from the above list to
restore it, or one of the following choices:
CHOICE          ACTION
-----
boot            restore the most recent post-boot backup
manifest_import restore the most recent manifest_import backup
-seed-         restore the initial starting repository (All
               customizations will be lost, including those
               made by the install/upgrade process.)
-quit-         cancel script and quit

Enter response [boot]: boot-20111215_150206
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example shown in the slide, the repository is being restored by using the most recent post-boot backup option. The confirmation for this selection is shown on the next page.

## Repairing a Corrupt Repository: Example

```
<output continued from previous page>
...
...
After confirmation, the following steps will be taken:

svc.startd(1M) and svc.configd(1M) will be quiesced, if running.
/etc/svc/repository.db
    -- renamed --> /etc/svc/repository.db_old_20111215_060922
/etc/svc/repository-boot-20111215_150206
    -- copied --> /etc/svc/repository.db
and the system will be rebooted with reboot(1M) .

Proceed [yes/no]? yes
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Here you are prompted for the final confirmation. Entering `yes` tells the system to remedy the fault. After the `restore_repository` command executes all the listed actions, the system reboots.

## Debugging Services During a System Boot

**Situation:** A system hangs during the booting process. Perform the following steps to address this problem.

1. Log in to the system as `root`.
2. Enable all services by using `svcadm milestone all`.
3. Determine where the boot process is hanging:
  - a. Run `svcs -a` to identify the services that are not running.
  - b. Look for error messages in `/var/svc/log`.
4. After fixing the issues, verify that all services have started:
  - a. Verify that all needed services are online by using `svcs -x`.
  - b. Verify that the console-login service dependencies are satisfied by using `svcs -l system/console-login:default`.
5. Continue the normal booting process.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

If problems occur with starting services, sometimes a system could hang during the booting process. You can use the steps shown in the slide to address this problem.

**Notes for step 2:** There is an additional system state associated with the `all` milestone. With the `all` milestone, all the services with a defined dependency on the `multiuser-server` milestone are started, as well as any services that do not have a defined dependency. If you have added services, such as third-party products, they may not be started automatically unless you use the `boot -m milestone=all` command.

**Notes for step 4b:** This command verifies that the login process on the console runs.

## Addressing `system/filesystem/local:default` Service Failures During Boot

**Situation:** Local file systems that are not required to boot the system are mounted by the `svc:/system/filesystem/local:default` service. When any of those file systems are unable to be mounted, the service enters the maintenance state. System startup continues, and any services that do not depend on `filesystem/local` are started. However, services that require `filesystem/local` to be online before starting through dependencies are not started.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered within a solid red rectangular bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Addressing system/filesystem/local:default Service Failures During Boot

Perform the following steps to change the configuration of the system so that a `sulogin` prompt appears immediately after the service fails instead of allowing system startup to continue:

1. Modify the `system/console-login` service as follows by using `svccfg -s svc:/system/console-login`.

```
# svc:/system/console-login> addpg site,filesystem-local dependency
# svc:/system/console-login> setprop site,filesystem-local/entities =
fmri: \ svc:/system/filesystem/local
# svc:/system/console-login> setprop site,filesystem-local/grouping =
astring: require_all
# svc:/system/console-login> setprop site,filesystem-local/restart_on =
astring: none
# svc:/system/console-login> setprop site,filesystem-local/type =
astring: service
# svc:/system/console-login> end
```

2. Refresh the service by using `svcadm refresh console-login`.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** When a failure occurs with the `system/filesystem/local:default` service, the `svcs -vx` command should be used to identify the failure. After the failure has been fixed, the following command clears the error state and allows the system boot to continue: `svcadm clear filesystem/local`.

## Summary

In this lesson, you should have learned how to:

- Describe the features and capabilities of SMF
- Administer SMF services
- Configure SMF services
- Secure SMF services
- Troubleshoot SMF services

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

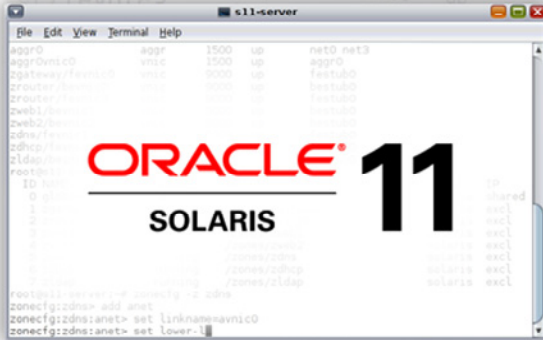


# 4

## Administering ZFS

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## System Administration for Experienced UNIX/Linux Administrators



Administering System  
Software by Using IPS



Administering Services  
by Using SMF



Administering ZFS



Configuring the Network



Administering Oracle Solaris  
Zones



Administering Privileges  
and RBAC



Installing the Oracle Solaris 11  
Operating System



Monitoring System Resources

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Objectives

After completing this lesson, you should be able to:

- Describe the merits of using ZFS in data management
- Administer ZFS storage pools
- Manage devices in ZFS storage pools
- Administer ZFS file systems
- Administer ZFS snapshots and clones
- Secure ZFS file systems

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Lesson Agenda

- Describing the merits of using ZFS in data management
- Administering ZFS storage pools
- Managing devices in ZFS storage pools
- Administering ZFS file systems
- Administering ZFS snapshots and clones
- Securing ZFS file systems

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Overview of ZFS

- Oracle Solaris 11 uses a very powerful and flexible technology known as ZFS to manage data storage.
- Some of the merits of using ZFS include:
  - Incorruptible file system
  - Scalability
  - Pooled storage
  - ZFS file system
  - Data integrity
    - Checksums
    - Self-healing data
  - Dynamic data striping
  - Disaster recovery
    - Snapshots
    - Clones

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Most importantly, ZFS provides a greatly simplified administration model. Through the use of a hierarchical file system layout, property inheritance, and automatic management of mount points and NFS share semantics, ZFS makes it easy to create and manage file systems without requiring multiple commands or the editing of configuration files. You can easily set quotas or reservations, turn compression on or off, or manage mount points for numerous file systems with a single command. You can examine or replace devices without learning a separate set of volume manager commands. You can send and receive file system snapshot streams.

## Transactional File System

- ZFS is a transactional file system, which means that the file system state is always consistent on disk.
- With a transactional file system, data is managed by using copy-on-write semantics.
- Data is never overwritten, and any sequence of operations is either entirely committed or entirely ignored.
- Thus, the file system can never be corrupted through accidental loss of power or a system crash.

**Note:** Although the most recently written pieces of data might be lost, the file system itself will always be consistent. In addition, synchronous data (written by using the `O_DSYNC` flag) is always guaranteed to be written before returning, so it is never lost.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ZFS is a transactional file system, in that the file system state is always consistent on disk. Traditional file systems overwrite data in place, which means that if the system loses power, for example, between the time a data block is allocated and when it is linked into a directory, the file system will be left in an inconsistent state. Historically, this problem was solved through the use of the `fsck` command. This command was responsible for reviewing and verifying the file system state, and repairing any inconsistencies during the process. This problem of inconsistent file systems caused great pain to administrators, and the `fsck` command was never guaranteed to address all possible problems. More recently, file systems have introduced the concept of **journaling**. The journaling process records actions in a separate journal, which can then be **replayed** safely if a system crash occurs. This process introduces unnecessary overhead because the data needs to be written twice, often resulting in a new set of problems, such as when the journal cannot be replayed properly.

# Scalability

- The ZFS file system is 128 bit, allowing for 256 quadrillion zettabytes of storage.
- Directories can have up to  $2^{48}$  (256 trillion) entries.
- There is no limit to the number of file systems or the number of files that can be contained within a file system.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

All metadata is allocated dynamically, so there is no need to pre-allocate inodes or otherwise limit the scalability of the file system when it is first created.

## Pooled Storage

- ZFS uses the concept of storage pools to manage physical storage.
- ZFS aggregates devices into a storage pool, instead of forcing you to create virtualized volumes.
- ZFS eliminates the need for an external volume management altogether.
- A ZFS storage pool can have the following components:
  - Disks
  - Slices
  - Files
  - Virtual devices

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A ZFS storage pool can consist of the following components:

- Disks: A disk is a piece of physical storage, which forms the most basic element of a storage pool. A disk can be:
  - Any block device of at least 128 MB in size
  - A hard drive visible to the system in the `/dev/dsk` directory, typically
  - A whole disk (`c1t0d0`) or an individual slice (`c0t0d0s7`)

To use whole disks:

- Use the `/dev/dsk/cXtXdX` naming convention.
- Specify using either the full path (`/dev/dsk/c1t0d0`) or a shorthand name consisting of the device name within the `/dev/dsk` directory (`c1t0d0`).

Disk device file syntax:

- `c#`: This device is attached to controller #.
- `t#`: This is the target controller number that is physically stored on the SCSI disk. On a SCSI controller, this is the SCSI target ID and is usually set via a switch on any external enclosure or by jumpers on the disk itself.
- `d#`: This is the disk number that is assigned by the system to a local disk.



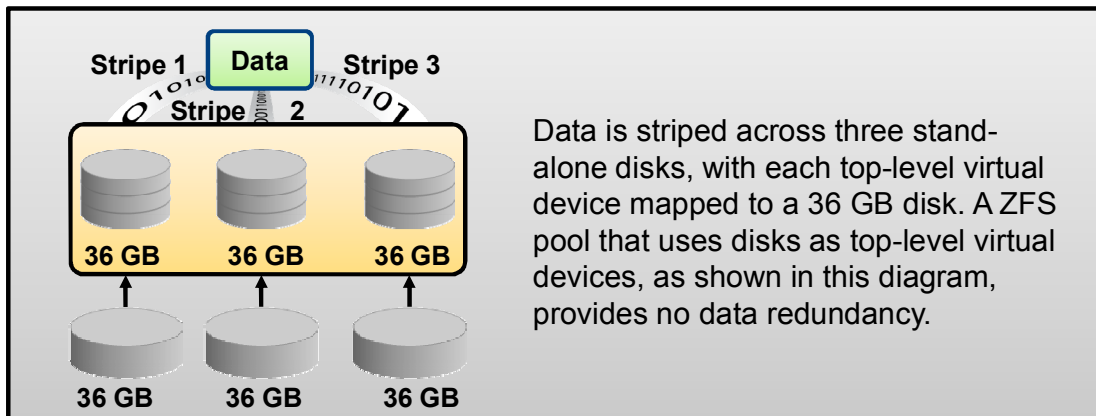
ZFS configurations become progressively more complex (from management, reliability, and performance perspectives) when you build pools from disk slices, Logical Unit Numbers (LUNs) in hardware RAID arrays, or volumes presented by software-based volume managers. In this course, you use whole disks only.

**Recommendation:** Using the whole disk is the simplest way to create ZFS storage pools because the disk does not need to be specially formatted. ZFS does the formatting by using an Extensible Firmware Interface (EFI) label to contain a single, large slice.

- **Slices:** Disks can be labeled with a traditional Solaris VTOC (SMI) label when you create a storage pool with a disk slice. For a bootable ZFS root pool, the disks must be labeled with an SMI label. The simplest configuration would be to put the entire disk capacity in slice 0 and use that slice for the root pool. For example, on a SPARC-based system with a 72 GB disk, you would need to have 68 GB of usable space located in slice 0. Similarly, on an x86-based system with a 72 GB disk, you would also need to allow 68 GB of usable space located in slice 0. A small amount of boot information is contained in slice 8. Slice 8 requires no administration and cannot be changed.
- **Files:** ZFS also allows you to use UFS files as virtual devices in your storage pool. This feature is not intended for production use. It is aimed primarily for testing and enabling simple experimentation. The reason is that any use of files relies on the underlying file system for consistency. If you create a ZFS pool backed by files on a UNIX file system (UFS), you are implicitly relying on UFS to guarantee correctness and synchronous semantics. However, files can be quite useful when you are first trying out ZFS or experimenting with more complicated layouts when not enough physical devices are present. All files must be specified as complete paths and must be at least 128 MB in size.
- **Virtual Devices:** Each ZFS storage pool is comprised of one or more virtual devices. A virtual device is a logical device in a pool that can be disks, disk slices, or files. A pool can have any number of virtual devices at the top of the configuration, known as top-level virtual devices or top-level vdevs. You can configure these virtual devices to stand alone within a pool (referred to as an unreplicated or non-redundant configuration) or combine them into a mirror or RAID-Z virtual device to provide data redundancy. Disks, disk slices, or files that are used in pools outside of mirrors and RAID-Z virtual devices, function as top-level virtual devices themselves.

## Dynamic Striping in a Storage Pool

- Data is dynamically striped across all top-level virtual devices.
- Data placement is done at write time.
- When a new virtual device is added, data is gradually allocated to the new device.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ZFS dynamically stripes data across all top-level virtual devices. The decision about where to place data is done at write time, so no fixed-width stripes are created at allocation time. When new virtual devices are added to a pool, ZFS gradually allocates data to the new device to maintain performance and space allocation policies.

**Note:** Although ZFS supports combining different types of virtual devices within the same pool, this practice is not recommended. This is because your fault tolerance is only as good as your worst virtual device. The recommended practice is to use top-level virtual devices of the same type with the same redundancy level in each device.

# Data Integrity

ZFS provides data integrity through checksums and self-healing data.

- Checksums:
  - All data and metadata is verified by using a user-selectable checksum algorithm.
  - All checksum verification and data recovery are performed at the file system layer, and are transparent to applications.
- Self-healing data:
  - ZFS provides for self-healing data through varying levels of data redundancy:
    - Mirrored storage pool configuration (RAID 1)
    - Parity storage pool configuration (RAID-Z)

ORACLE

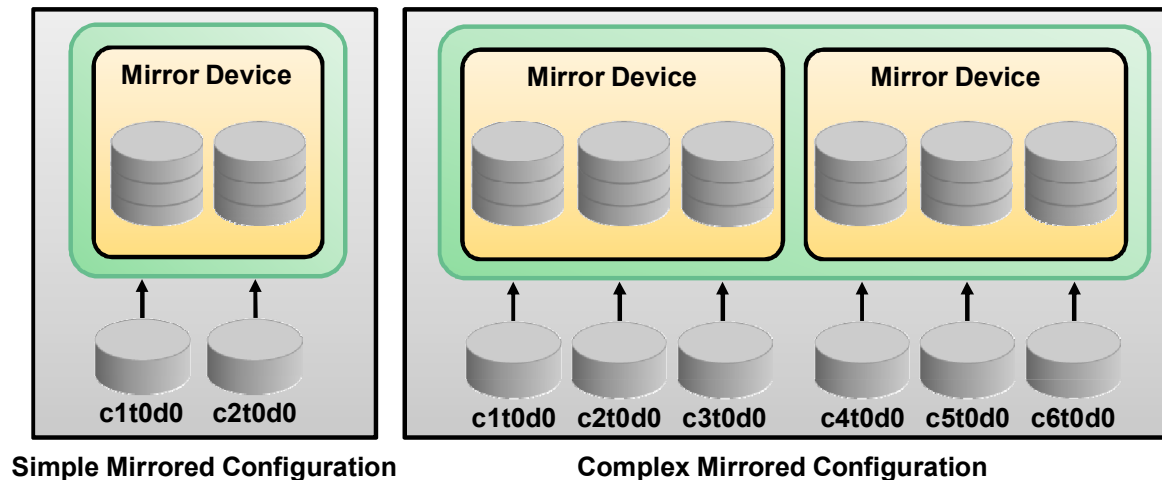
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

- **Checksums:** With ZFS, all data and metadata is verified by using a user-selectable checksum algorithm. Traditional file systems that do provide checksum verification have performed it on a per-block basis, out of necessity due to the volume management layer and traditional file system design. The traditional design means that certain failures, such as writing a complete block to an incorrect location, can result in data that is incorrect but has no checksum errors. ZFS checksums are stored in a way such that these failures are detected and can be recovered from gracefully. All checksum verification and data recovery are performed at the file system layer, and are transparent to applications.
- **Self-healing data:** ZFS supports storage pools with varying levels of data redundancy. When a bad data block is detected, not only does ZFS fetch the correct data from another replicated copy, but it also repairs the bad data by replacing it with the correct data.

## Mirrored Storage Pool Configuration (RAID-1)

Mirrored storage pool configuration:

- Requires at least two disks
- Allows more than one mirror to be created in each pool



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

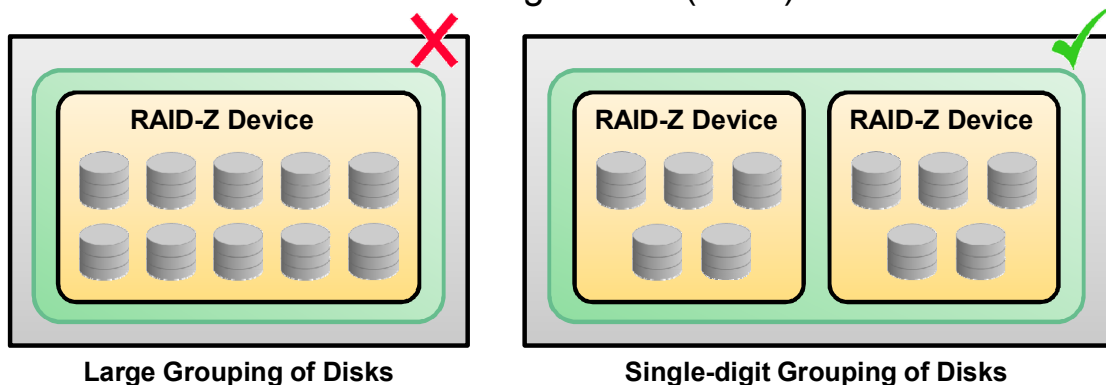
A mirrored storage pool configuration requires at least two disks, preferably on separate controllers. Many disks can be used in a mirrored configuration. In addition, you can create more than one mirror in each pool.

This slide shows examples of two mirrored storage pool configurations. The diagram on the left shows an example of a simple mirrored configuration. The storage pool contains one mirror with two disks. In this example, you could lose only one disk before you start to lose data.

An example of a more complex mirrored configuration is shown in the diagram on the right. The storage pool there contains two mirrors with three disks each. With the more complex mirrored configuration example, you could lose up to two disks in each mirror and not lose any data.

## Parity Storage Pool Configuration (RAID-Z )

- A RAID-Z storage pool can be a configuration with single-, double-, or triple-parity fault tolerance.
- Minimum disk usage recommendations by RAID-Z level:
  - `raidz` or `raidz1`: Use at least three disks (2 + 1).
  - `raidz2`: Use at least five disks (3 + 2).
  - `raidz3`: Use at least eight disks (5 + 3).



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A RAID-Z storage pool can be a configuration with single-, double-, or triple-parity fault tolerance, which means that one, two, or three device failures can be sustained respectively without any data loss.

Single-parity RAID-Z (`raidz` or `raidz1`) is similar to RAID-5. Double-parity RAID-Z (`raidz2`) is similar to RAID-6. Triple-parity RAID-Z (`raidz3`) is similar to `raidz2` with an additional parity protection level.

For a `raidz1` configuration, you must have at least three disks (2 + 1). For a `raidz2` configuration, use at least five disks (3 + 2). For a `raidz3` configuration, use at least eight disks (5 + 3).

RAID-Z configurations with single-digit groupings of disks should perform better. Therefore, it is recommended that if you are creating a RAID-Z configuration with many disks, try dividing the disks into smaller groupings. For example, if you have a RAID-Z configuration with 14 disks, it would be better to split these 14 disks into two 7-disk groupings.

# ZFS File System

- A ZFS file system is built on top of a storage pool.
- ZFS file systems can be dynamically created and destroyed without requiring you to allocate or format any underlying disk space.

The Oracle logo, consisting of the word "ORACLE" in white, uppercase letters on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Historically, file systems were constructed on top of a single physical device. To address multiple devices and provide for data redundancy, the concept of a *volume manager* was introduced. This design added another layer of complexity and ultimately prevented certain file system advancements because the file system had no control over the physical placement of data on the virtualized volumes.

With ZFS, file systems are no longer constrained to individual devices, allowing them to share disk space with all file systems in the pool. You no longer need to predetermine the size of a file system, as file systems grow automatically within the disk space allocated to the storage pool. When new storage is added, all file systems within the pool can immediately use the additional disk space without additional work.

# Snapshots

- A snapshot is a read-only copy of a file system or volume.
- Snapshots can be created quickly and easily.
- Initially, snapshots consume no additional disk space within the pool.
- As data within the active dataset changes, the snapshot consumes disk space by continuing to reference the old data.
- As a result, the snapshot prevents the space consumed by the data from being released back to the pool.
- When snapshots are destroyed, consumed space is released.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** ZFS allows you to take  $2^{64}$  instantaneous snapshots of file systems.

## ZFS Clones

- A clone is writable file system or volume.
- A clone is created from a snapshot.
- As with snapshots, creating a clone is nearly instantaneous and initially consumes no additional disk space.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Quiz

For a redundant configuration using parity, Oracle Solaris 11.1 ZFS file system supports:

- a. raidz/raidz1 only with single parity
- b. raidz/raidz1 and raidz2 with single and double parity, respectively
- c. raidz/raidz1, raidz2, and raidz3 with single, double, and triple parity, respectively

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: c**

## Lesson Agenda

- Explaining the role of ZFS in data management
- **Administering ZFS storage pools**
- Managing devices in ZFS storage pools
- Administering ZFS file systems
- Administering ZFS snapshots and clones
- Securing ZFS file systems

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Administering ZFS Storage Pools

- ZFS storage pools are administered by using the `zpool` command.
- Administering ZFS storage pools involves the following activities:
  - Creating ZFS storage pools
  - Displaying ZFS storage pool information
  - Destroying ZFS storage pools
  - Managing ZFS storage pool properties

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Creating ZFS Storage Pools

- You can create the following types of pools by using the `zpool create` command:
  - Basic storage pool
  - Mirrored storage pool
  - RAID-Z storage pool
  - ZFS storage pool with log devices
  - ZFS storage pool with cache devices
- However, before creating these pools, you must determine the following:
  - Local storage disk availability
  - Default mount point for storage pools

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Before creating ZFS storage pools, review the following precautions and observations:

- Do not repartition or relabel disks that are part of an existing storage pool. If you attempt to repartition or relabel a root pool disk, you might have to reinstall the OS.
- Do not create a storage pool that contains components from another storage pool, such as files or volumes. Deadlocks can occur in this unsupported configuration.
- A pool created with a single slice or single disk has no redundancy and is at risk of data loss. A pool created with multiple slices but no redundancy is also at risk of data loss. A pool created with multiple slices across disks is harder to manage than a pool created with whole disks.
- A pool that is not created with ZFS redundancy (RAIDZ or mirror) can only report but not repair data inconsistencies.
- Although a pool that is created with ZFS redundancy can help reduce downtime due to hardware failures, it is not immune to hardware failures, power failures, or disconnected cables. Ensure that you backup your data on a regular basis. Performing routine backups of pool data on non-enterprise grade hardware is important.
- A pool cannot be shared across systems. ZFS is not a cluster file system.

# Determining Local Storage Disk Availability

Prior to creating a pool, run `format` to display disk availability.

```
# format
Searching for disks...done
AVAILABLE DISK SELECTIONS:
  0. c7t0d0 <ATA-VBOX HARDDISK-1.0-40.00GB>
    /pci@0,0/pci8086,2829@d/disk@0,0
  1. c7t2d0 <ATA-VBOX HARDDISK-1.0 cyl 1022 alt 2 hd 64 sec 32>
    /pci@0,0/pci8086,2829@d/disk@2,0
  2. c7t3d0 <ATA-VBOX HARDDISK-1.0 cyl 1022 alt 2 hd 64 sec 32>
    /pci@0,0/pci8086,2829@d/disk@3,0
  3. c7t4d0 <ATA-VBOX HARDDISK-1.0 cyl 1022 alt 2 hd 64 sec 32>
    /pci@0,0/pci8086,2829@d/disk@4,0
  4. c7t5d0 <ATA-VBOX HARDDISK-1.0 cyl 1022 alt 2 hd 64 sec 32>
    /pci@0,0/pci8086,2829@d/disk@5,0
  5. c7t6d0 <ATA-VBOX HARDDISK-1.0 cyl 1022 alt 2 hd 64 sec 32>
    /pci@0,0/pci8086,2829@d/disk@7,0
  8. c7t7d0 <ATA-VBOX HARDDISK-1.0 cyl 1022 alt 2 hd 64 sec 32>
    /pci@0,0/pci8086,2829@d/disk@8,0
  7. c7t8d0 <ATA-VBOX HARDDISK-1.0 cyl 1022 alt 2 hd 64 sec 32>
    /pci@0,0/pci8086,2829@d/disk@9,0
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can determine the disks that are available for creating a pool by using the `format` command. The disks that you should use in each pool should be provided to you. When creating a storage pool, if you choose a disk that is unavailable, the system gives an error message like the following:

```
# zpool create u01 c12t0d0
cannot open 'c12t0d0': no such device in /dev/dsk
```

## Default Mount Point for Storage Pools

- When a pool is created, the default mount point is `/pool-name`.
- If the `/pool-name` directory exists, it must be empty.
- If the directory does not exist, ZFS automatically creates it.
- To create a pool with a different default mount point, use the `-m` option with the `zpool create` command.

```
# zpool create -m /export/zfs home c1t0d0
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** If the directory is not empty, you receive an error message stating that the mount point exists and it is not empty. The error message directs you to use the `-m` option to provide a different default mount point.

To create a pool with a different default mount point, use the `-m` option with the `zpool create` command.

For example, in the code sample shown in the slide, the default mount point of the pool named `home` located on disk `c1t0d0` is being changed to a different mount point, `/export/zfs`.

## Creating a Basic ZFS Storage Pool

To create a basic ZFS pool, enter `zpool create` followed by the pool name and any number of virtual devices.

```
# zpool create hrpool c1t0d0 c1t1d0
```

Both disks are:

- Found in `/dev/dsk`
- Labeled by ZFS to contain a single, large slice
- Dynamically striped across with data

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To create a basic storage pool, enter the `zpool create` command followed by the new pool name and the names of the disks that you want in the pool. In the example in the slide, a pool called `hrpool` is created by using the `zpool create` command and the pool consists of two disks: `c1t0d0` and `c1t1d0`. These whole disks are found in the `/dev/dsk` directory and are labeled appropriately by ZFS to contain a single large slice. Data is dynamically striped across both disks.

## Creating a Mirrored Storage Pool

- To create a ZFS mirrored storage pool, use `zpool create` followed by the pool name, the `mirror` keyword, and the storage devices that comprise the mirror.

```
# zpool create hrpool mirror c1t0d0 c2t0d0 mirror c3t0d0 c4t0d0
```

- Note that data is:
  - Dynamically striped across both mirrors
  - Redundant between each disk within a mirror

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Note that multiple mirrors can be specified by repeating the `mirror` keyword on the command line. The command shown in the example creates a pool called `hrpool` with two 2-way mirrors. The first mirror contains the devices `c1t0d0` and `c2t0d0`, and the second mirror contains the devices `c3t0d0` and `c4t0d0`.

Data is dynamically striped across both mirrors, with data being redundant between each disk within a mirror.



## Creating a RAID-Z Storage Pool

To create a ZFS RAID-Z storage pool, enter `zpool create` followed by the pool name, the `raidz` keyword, and the storage devices that will be part of each RAID-Z pool.

```
# zpool create hrpool raidz c1t0d0 c2t0d0 c3t0d0 c4t0d0
/dev/dsk/c5t0d0
```

```
# zpool create appool raidz2 c1t0d0 c2t0d0 c3t0d0 c4t0d0 c5t0d0
c6t0d0 raidz2 c8t0d0 c9t0d0 c10t0d0 c11t0d0 c12t0d0 c13t0d0
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Creating a RAID-Z storage pool is identical to creating a mirrored storage pool, except that the `raidz`, `raidz2`, or `raidz3` keyword is used instead of `mirror`.

In the first code example in the slide, a single-parity RAID-Z storage pool called `hrpool` is created by using the `zpool create` command. The pool consists of five disks: `c1t0d0`, `c2t0d0`, `c3t0d0`, `c4t0d0`, and `c5t0d0`. The `/dev/dsk/` path has been included for the `c5t0d0` disk to illustrate that disks can be specified by using their full paths. The `/dev/dsk/c5t0d0` device is identical to the `c5t0d0` device.

You could also use disk slices in this configuration, but you would need to preformat the disks to have appropriately sized slices.

In the second code example, a double-parity RAID-Z storage pool has been created called `appool`. The first `raidz2` virtual device contains six disks: `c1t0d0`, `c2t0d0`, `c3t0d0`, `c4t0d0`, `c5t0d0`, and `c6t0d0`. The second `raidz2` virtual device also contains six disks: `c8t0d0`, `c9t0d0`, `c10t0d0`, `c11t0d0`, `c12t0d0`, and `c13t0d0`.

## Creating a ZFS Storage Pool with Log Devices

- A log device keeps track of the write operations.
- When a faulty device is replaced, the writes from the log device are applied.
- A log device:
  - Can be added as part of, or after, pool creation
  - Can be removed from the pool
  - Is designated by the keyword `log`

```
# zpool create appool mirror c1t1d0 c1t2d0 log mirror c1t5d0  
c1t8d0
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can specify a separate log device when a pool is created or add it after the pool is created. You can also remove a log device from a pool. The keyword `log` is used to designate a device as the log device.

In the code example in the slide, a mirrored storage pool called `appool` is created that consists of two virtual devices. The first virtual device contains the disks `c1t1d0` and `c1t2d0`. The second virtual device, which is the log device, contains the disks `c1t5d0` and `c1t8d0`.

## Creating a ZFS Storage Pool with Cache Devices

- Cache devices provide an additional layer of caching between main memory and disk.
- You can create a storage pool with one or more cache devices to cache storage pool data.
- Designated with the keyword `cache`, cache devices can be added as part of, or after, pool creation.

```
# zpool create appool mirror c0t2d0 c0t4d0 cache c0t0d0
```

- You can remove cache devices from the pool after the pool is created.
- However, cache devices cannot be mirrored or be a part of a RAID-Z configuration.
- You can monitor cache statistics with `zpool iostat`.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Cache devices provide an additional layer of caching between main memory and disk. If the data is not in the adaptive replacement cache (ARC), ZFS attempts to serve the requests from the L2ARC, also called cache drives. These cache drives are physically multi-level cell (MLC) style solid-state drives (SSDs). These SSDs are slower than system memory, but much faster than hard drives.

In the code example in the slide, a mirrored pool called `appool` is created that consists of two disks, `c0t2d0` and `c0t4d0`, and the `c0t0d0` device in the pool is designated as the cache.

## Displaying ZFS Storage Pool Information

- You can use the `zpool list` command to display basic information about pools.

```
# zpool list
```

NAME	SIZE	ALLOC	FREE	CAP	HEALTH	ALTROOT
hrpool	80.0G	22.3G	47.7G	28%	ONLINE	-
appool	1.2T	384G	816G	32%	ONLINE	

- You can use `zpool status` to display virtual devices and the physical devices in a ZFS storage pool.

```
# zpool status appool
pool: appool
state: ONLINE
scrub: none requested
config:
  NAME                STATE      READ      WRITE     CKSUM
  appool              ONLINE    0         0         0
  mirror-0            ONLINE    0         0         0
    c0t1d0             ONLINE    0         0         0
```



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Destroying ZFS Storage Pools

To destroy a ZFS storage pool, enter `zpool destroy` followed by the pool name.

```
# zpool destroy testpool
```

**Caution:** Be cautious when you destroy a pool. Ensure that you are destroying the right pool and that you always have copies of your data. If you accidentally destroy the wrong pool, you can attempt to recover the pool.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To destroy a pool, use the `zpool destroy` command followed by the pool name. This command destroys the pool even if it contains mounted datasets. In the example shown here, the pool named `testpool` is being destroyed.

## Managing ZFS Storage Pool Properties

- How a storage pool behaves is determined by its pool properties.
- You can use the following commands to view the various properties of the storage pool.

Command	Description
<code>zpool get all</code>	Display all property information.
<code>zpool list</code>	Display usage information about all pools on the system.
<code>zpool iostat</code>	Display the accumulated statistics since boot for all pools on the system.
<code>zpool status -x</code>	Request a quick overview of pool health status.
<code>zpool status</code>	Examine the health of a specific pool.
<code>zpool history</code>	Display <code>zpool</code> commands that modify pool state information.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

How a storage pool behaves is determined by its pool properties. Properties determine the behavior of a pool feature, such as whether the pool is bootable or whether a particular property is enabled.

For example, the `autoreplace` property controls automatic device replacement. If it is set to `off`, device replacement must be initiated by using the `zpool replace` command. If it is set to `on`, any new device that is found in the same physical location as a device that previously belonged to the pool is automatically formatted and replaced. The default behavior is `off`.

A pool property can also identify a read-only attribute, such as the current pool size or the unique identifier for the pool, the GUID.

## Quiz

You have created a storage pool called `repopool` on disk `c1t0d0`. Which of the following commands would you use to change the default mount point to `/IPS/base`?

- a. `zpool create -m repopool /IPS/base c1t0d0`
- b. `zpool create -m /IPS/base repopool c1t0d0`
- c. `zpool create -m c1t0d0 repopool /IPS/base`
- d. `zpool create -m repopool c1t0d0 /IPS/base`

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered within a solid red rectangular bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

## Lesson Agenda

- Explaining the role of ZFS in data management
- Administering ZFS storage pools
- **Managing devices in ZFS storage pools**
- Administering ZFS file systems
- Administering ZFS snapshots and clones
- Securing ZFS file systems

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Managing Devices in ZFS Storage Pools

After you have created a pool, there are several tasks that you can perform to manage the devices within the pool, such as:

- Adding top-level virtual devices
- Attaching and detaching devices
- Taking a device offline
- Bringing a device online
- Replacing a device
- Designating hot spares

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Adding Devices to a Storage Pool

- To add a new virtual device to a pool, use the `zpool add` command.

```
# zpool add appool mirror c2t1d0 c2t2d0
```

- By adding a new top-level virtual device, space is:
  - Dynamically added to the pool
  - Immediately available to all the datasets within the pool

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the code example in the slide, a mirrored device is added to an existing pool called `appool`. The mirror consists of two disks: `c2t1d0` and `c2t2d0`.

**Note:** Use `zpool status` to determine the disks that are currently configured for a storage pool. Then, before adding a device to the pool, you must execute the `format` command to identify any additional disks configured in the system.

## Attaching Devices to a Storage Pool

To attach a new device to an existing mirrored or non-mirrored virtual device (vdev), use the `zpool attach` command.

```
# zpool attach appool c1t1d0 c2t1d0
# zpool status appool
  pool: appool
  state: ONLINE
  scan: resilver completed after 0h0m with 0 errors on Tue Dec
13 14:11:33 2013
config:
  NAME                STATE          READ    WRITE   CKSUM
  appool              ONLINE         0        0       0
    mirror-0         ONLINE         0        0       0
      c0t1d0          ONLINE         0        0       0
      c1t1d0          ONLINE         0        0       0
      c2t1d0          ONLINE         0        0       0 73.5K resilvered
```

**Resilvering:** The process of transferring data from one device to another device.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use the `zpool attach` command to transform a two-way mirrored configuration into a three-way mirrored configuration or convert a non-redundant storage pool into a redundant storage pool.

In the example in the slide, a new device, `c2t1d0`, is being attached to an existing device, `c1t1d0`, to create either a mirrored pool or a three-way mirror in an already mirrored pool.

You can run the `zpool status` command to verify that the device is attached successfully. Observe that the new device has already been resilvered. In ZFS, the new device begins to resilver immediately.

## Detaching Devices from a Storage Pool

- To detach a device from a mirrored storage pool, use the `zpool detach` command.

```
# zpool detach appool c2t1d0
```

- The device to be detached should be offline.
- The operation is refused if there are no other valid replicas of the data.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For example, if you want to detach the `c2t1d0` device that you just attached to the mirrored pool `appool`, you can do so by entering the command `zpool detach appool c2t1d0` as shown in the code example. You can verify that the device has been detached by running the `zpool status` command again.

## Taking Devices Offline in a Storage Pool

- When hardware is unreliable or not functioning properly, ZFS continues to read or write data to the device, assuming that the condition is only temporary.
- If the condition is not temporary, you can instruct ZFS to ignore the device by taking it offline.
- To take a device offline, use `zpool offline` followed by the pool name and the device name.

```
# zpool offline hrpool c1t0d0  
bringing device c1t0d0 offline
```

**Note:** When a device is taken offline, it is not detached from the storage pool. This means that you cannot use the device in another pool.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the code example in the slide, the `c1t0d0` device is taken offline. This device is located in the pool named `hrpool`.

When a device is taken offline, ZFS does not send any requests to that device. By default, the offline state is persistent; consequently, the device remains offline even after the system is rebooted. If you want to take a device offline temporarily and have it automatically returned to the `ONLINE` state after the system is rebooted, use the `zpool offline -t` command instead.

## Bringing Devices Online in a Storage Pool

- To bring back a device online, use `zpool online` followed by the pool name and the device name.

```
# zpool online hrpool c1t0d0  
bringing device c1t0d0 online
```

- Data that was added to the storage pool while the device was offline resilveres to the device after it is brought online.

**Note:** You cannot use `zpool online` to replace a disk.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** You cannot use device onlining to replace a disk. If you take a device offline, replace the drive, and try to bring it online, it remains in the faulted state.

## Replacing Devices in a Storage Pool

- To replace a failed device with another device in the same location, use `zpool replace` followed by the pool name and the device name.

```
# zpool replace hrpool c1t1d0
```

- If the device is in a different location, specify both devices.

```
# zpool replace hrpool c1t1d0 c1t2d0
```

**Note:** The replacement device must be greater than or equal to the minimum size of all the devices in the configuration.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

There are two reasons why you may want or need to replace a device. You may want to replace a device with a larger device, or you may need to replace a failing or failed device.

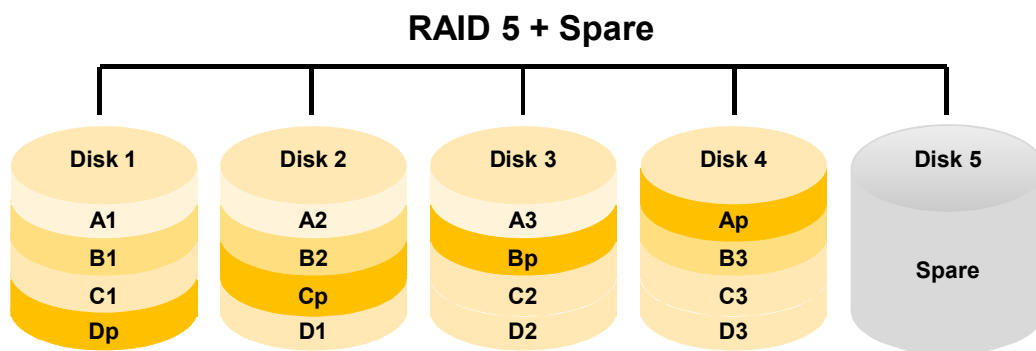
To replace a device, use the `zpool replace` command followed by the pool name and the device name.

- If you are physically replacing a device with another device in the same location in a redundant pool, you need to identify only the replaced device. ZFS recognizes that it is a different disk in the same location. In the first example in the slide, the disk `c1t1d0` is replaced in the pool named `hrpool`.
- If you are replacing a device in a storage pool with a disk in a different location, you must specify both devices. This is shown in the second example, where disk `c1t1d0` in the pool named `hrpool` is replaced with disk `c1t2d0`.

**Note:** For the replacement operation to be successful, the replacement device must be greater than or equal to the minimum size of all the devices in a mirror or RAID-Z configuration.

## Designating Hot Spares in a Storage Pool

- The ZFS hot spares feature enables you to identify disks that can be used to replace a failed or faulted device in one or more storage pools.
- The hot spare device is not an active device in a pool, but if an active device in the pool fails, the hot spare automatically replaces the failed device.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** The device or devices that you designate as spares must be equal to or larger than the size of the largest disk in the pool.

After a failed device has been replaced and resilvered, the spare is automatically detached and made available. An in-progress spare replacement can be cancelled by detaching the spare. If the original faulted device is detached, the spare assumes its place in the configuration and is removed from the spare's list of all active pools.



## Creating Hot Spares in a Storage Pool

To designate hot spares to a pool, use `zpool create` followed by the pool name, the configuration, the keyword `spare`, and the names of the spares.

```
# zpool create appool mirror c1t1d0 c2t1d0 spare c1t2d0 c2t2d0
# zpool status appool
pool: appool
state: ONLINE
scan: none requested
config:
    NAME           STATE      READ    WRITE  CKSUM
    appool          ONLINE     0        0      0
    mirror-0        ONLINE     0        0      0
    c1t1d0           ONLINE     0        0      0
    c2t1d0           ONLINE     0        0      0
    spares
    c1t2d0           AVAIL
    c2t2d0           AVAIL
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example in the slide, a pool called `appool` is being created. Within this pool is a mirror that contains two disks: `c1t1d0` and `c2t1d0`. Disks `c1t2d0` and `c2t2d0` have been designated as spares.

Looking at the status of `appool`, note that the spares are part of the pool and that they are available. In this scenario, if either or both of the mirrored disks were to fail, ZFS automatically replaces them with one or both of the available spares.

## Adding Hot Spares to a Storage Pool

To add hot spares to an existing pool, use `zpool add` followed by the pool name, the keyword `spare`, and the name of the hot spares.

```
# zpool add appool spare c1t3d0 c2t3d0
# zpool status appool
pool: appool
state: ONLINE
scan: none requested
config:
  NAME          STATE      READ  WRITE  CKSUM
  appool        ONLINE     0     0     0
  mirror-0      ONLINE     0     0     0
    c1t1d0       ONLINE     0     0     0
    c2t1d0       ONLINE     0     0     0
  spares
    c1t3d0       AVAIL
    c2t3d0       AVAIL
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example in the slide, disks `c1t3d0` and `c2t3d0` are designated as spares and added to the pool named `appool`.

Running the `zpool status` command for the pool shows that the spares have been added successfully. Notice that both spares are available (AVAIL).

Again, as you saw in the previous example, if either or both of the mirrored disks were to fail, ZFS automatically replaces them with one or both of the available spares.

Next, you look at an example in which one of the active devices in `appool` has faulted and ZFS automatically replaces the faulted device with one of the available spares.

# Replacing a Faulted Device With a Hot Spare

Example of a hot spare replacing a faulted device:

```
# zpool status appool
pool: appool
state: DEGRADED
status: One or more devices could not be opened.  Sufficient replicas
       exist for the pool to continue functioning in a degraded state.
action: Attach the missing device and online it using 'zpool online'.
       see: http://www.sun.com/msg/ZFS-8000-2Q
       scan: resilvered completed 0h12m with 0 errors on Tue Dec 13 14:16:04
           2011
config:
  NAME           STATE      READ    WRITE   CKSUM
  appool         DEGRADED    0        0        0
  mirror-0      DEGRADED    0        0        0
  c1t1d0         ONLINE      0        0        0
  spare-1        UNAVAIL     0        0        0
  c2t1d0         UNAVAIL     0        0        0 cannot open
  c1t3d0         ONLINE      0        0        0 58.5K resilvered
spares
  c1t3d0         INUSE              currently in use
  c2t3d0         AVAIL
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In this example, disk `c2t1d0` has faulted and is replaced automatically by the hot spare `c1t3d0`, which has been resilvered and now appears as an active device in the mirrored pool. Notice also that the status of the hot spare `c1t3d0` has changed from available (AVAIL) to in use (INUSE).

Next, is an example of removing a hot spare.

## Removing Hot Spares in a Storage Pool

To remove a hot spare, use `zpool remove` followed by the pool name and the name of the hot spare.

```
# zpool remove appool c1t2d0
# zpool status appool
pool: appool
(output omitted)
  spares
    c1t3d0      AVAIL
```

**Note:** You cannot remove a hot spare if it is currently being used by the storage pool as an active device.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In this example, the hot spare `c1t2d0` is being removed from the pool named `appool`, leaving just one hot spare in the pool: `c1t3d0`.

As you can see in this example, after it is removed, the hot spare no longer appears in the spares set.

## Quiz

The `c1t2d0` device in `hrpool` is not being quite used. You wish to take `c1t2d0` offline and use it in `appool`, where you could do with some additional disk. Is this exercise possible?

- a. Yes
- b. No

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

## Lesson Agenda

- Explaining the role of ZFS in data management
- Administering ZFS storage pools
- Managing devices in ZFS storage pools
- **Administering ZFS file systems**
- Administering ZFS snapshots and clones
- Securing ZFS file systems

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Administering ZFS File Systems

- ZFS file systems are administered by using the `zfs` command.
- Administering ZFS file systems involves the following activities:
  - Creating, renaming, and destroying a ZFS file system
  - Mounting and unmounting ZFS file systems
  - Getting and setting ZFS properties

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered within a solid red rectangular bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `zfs` command provides a set of subcommands that perform specific operations on file systems. Snapshots, volumes, and clones are also managed by using the `zfs` command.

## Creating a ZFS File System

- To create a file system, enter `zfs create` followed by the file system path name.

```
# zfs create hrpool/home/reports
```

- The file system that you create is mounted by default under a mount point that has the same name as the file system.
  - For example, if you create a file system called `abc/one`, it will be mounted under `/abc/one`.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use the `zfs create` command to create ZFS file systems. The `create` subcommand takes a single argument: the name of the file system to create. In the example shown in the slide, a file system named `reports` is being created in the `hrpool/home` file system.

The file system name is specified as a path name starting from the name of the pool: `pool-name/[filesystem-name/]filesystem-name`

The pool name and initial file system names identify the location in the hierarchy where a new file system will be created. The last name identifies the file system to be created.

**Note:** You can create any non-existent, intermediate file system names automatically by using the `-p` option with the `zfs create` command.



# Renaming a ZFS File System

You can use the `rename` subcommand to:

- Change the name of a file system

```
# zfs rename hrpool/home/reviews hrpool/home/reviews_2011
```

- Relocate the file system to a new location within the ZFS hierarchy

```
# zfs rename hrpool/home/jobdesc hrpool/ws/jobdesc
```

- Change the name of a file system and relocate it within the ZFS hierarchy

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The rename operation attempts an unmount/remount sequence for:

- The file system
- Any descendent file systems

If unable to unmount an active file system:

- Rename operation fails
- Forced unmount is required

In the first example shown in the slide, the `reviews` file system that resides in `hrpool/home` is being renamed to `reviews_2011`.

In the second example, the `jobdesc` file system is being relocated from `hrpool/home` to `hrpool/ws`.

**Note:** When you relocate a file system through `rename`, the new location must be within the same pool and it must have enough space to hold this new file system. If the new location does not have enough space, possibly because it has reached its quota, the rename fails.

## Destroying a ZFS File System

- To destroy a file system, enter `zfs destroy` followed by the file system path name.

```
# zfs destroy hrpool/home/oldreports
```

- If the `zfs destroy` command fails, use one of the options shown here.

Condition	Option	Results
File system is busy.	-f	This option can unmount, unshare, and destroy active file systems, causing unexpected application behavior.
File system has children.	-r	Recursively destroys a file system and all its descendents. This option also destroys snapshots.
File system has indirect dependents.	-R	Forces the destruction of all dependents, including cloned file systems outside the target hierarchy.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example shown here, the file system named `oldreports` is being destroyed in `hrpool/home`.

**Caution:** No confirmation prompt appears with the `destroy` subcommand; therefore, use it with extreme caution. Unlike the `zpool destroy` command, the `zfs destroy` command cannot be reversed and the file system cannot be recovered.

If the `zfs destroy` command fails, there are several options that you can use to force the destruction of the file system. If the file system to be destroyed is busy and thus cannot be unmounted, you can use the `-f` option. Use this option with caution because it can unmount, unshare, and destroy active file systems, causing unexpected application behavior.

The `zfs destroy` command also fails if a file system has children. To recursively destroy a file system and all its descendents, use the `-r` option. Note that a recursive destroy also destroys snapshots, so you should use this option with caution.

If the file system to be destroyed has indirect dependents, even the recursive destroy command described above fails. To force the destruction of *all* dependents, including cloned file systems outside the target hierarchy, the `-R` option must be used.

## Mounting ZFS File Systems

- By default, all ZFS file systems are mounted by ZFS at boot by using the Service Management Facility (SMF) `svc:/system/filesystem/local` service.
- File systems are mounted under the mount point mentioned in the `mountpoint` property of the file system.
- File systems are mounted under a mount point that has the same name as the file system only if the `mountpoint` property of ZFS file system is not set.
- You can use the `zfs mount` command to:
  - View the currently mounted ZFS managed file systems
  - Change mount options
  - Explicitly mount a file system

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Mounting ZFS File Systems

- To view all file systems currently mounted and managed by ZFS, use `zfs mount` with no arguments.

```
# zfs mount
hrpool /hrpool
hrpool/home /hrpool/home
hrpool/home/reports /hrpool/home/reports
```

- To mount all ZFS managed file systems, use the `zfs mount -a` command.

```
# zfs mount -a
```

**Note:** A mounted file system uses a set of mount options based on the property values associated with the dataset.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `zfs mount` command with no arguments shows all currently mounted file systems that are managed by ZFS. Note that the legacy managed mount points are not displayed.

You can use the `-a` option with the `zfs mount` command to mount all ZFS managed file systems. Legacy managed file systems are not mounted.

**Note:** When a file system is mounted, it uses a set of mount options based on the property values associated with the dataset. If any of these mount options are set explicitly by using the `-o` option with the `zfs mount` command, the associated property value is temporarily overridden. These property values are reported as temporary by the `zfs get` command and revert to their original settings when the file system is unmounted. If a property value is changed while the dataset is mounted, the change takes effect immediately, overriding any temporary setting.

## Unmounting a ZFS File System

To unmount a ZFS file system, use `zfs unmount` followed by either the file system name or mount point.

```
# zfs unmount hrpool/home/qarpts
```

```
# zfs unmount /export/home/qarpts
```

**Note:** If the file system is active or busy, `zfs unmount` fails. You can use `-f` to force the unmount, but you should use this option with caution.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the first example, the file system `hrpool/home/qarpts` is being unmounted by file system name.

In the second example, same file system is being unmounted by mount point name (`/export/home/qarpts`).

## ZFS File System Properties

- Properties allow you to control the behavior of:
  - File systems
  - Volumes
  - Snapshots
  - Clones
- There are two types of ZFS file system properties:
  - Native properties
    - Export internal statistics
    - Control ZFS file system behavior
  - User-defined properties
    - No effect on ZFS file system behavior
    - Can be used to annotate datasets

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The primary focus in this course is on native properties. For more information about user properties, refer to the “ZFS User Properties” section in the *Oracle Solaris Administration: ZFS File Systems* course.

## ZFS File System Native Properties

There are two types of native properties:

- Settable properties
  - Can be both retrieved and set
  - Most are inheritable (exceptions: `quota` and `reservation`)
- Read-only properties
  - Can be retrieved but not set
  - Are not inherited

**Note:** An inheritable property is a property that, when set on a parent, is propagated to all its descendents.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## ZFS File System Native Properties

Property Name	Type	Default Value	Description
<code>compression</code>	String	<code>off</code>	Enables or disables compression for a dataset
<code>mountpoint</code>	String	N/A	Controls the mount point used for this file system
<code>quota</code>	Number (or none)	<code>none</code>	Limits the amount of disk space that a dataset and its descendents can consume
<code>readonly</code>	Boolean	<code>off</code>	Controls whether a dataset can be modified. When it is set to <code>on</code> , no modifications can be made.
<code>sharenfs</code>	String	<code>off</code>	Controls whether a ZFS dataset is published as an NFS share



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide displays a sampling of the native ZFS properties, both settable and read-only. The next two slides explain how to set the `quota` property. For a complete list and full descriptions of the native ZFS properties and how to set them, refer to the “Introducing ZFS Properties” section of *Oracle Solaris Administration: ZFS File Systems* course.



## Setting Quotas for ZFS File Systems

- The `quota` property in ZFS allows you to set a limit on the amount of space a file system can use.
- To set a quota on a file system, use `zfs set`, followed by `quota=`, the space amount, and the file system name.

```
# zfs set quota=10g rpool/export/home/jjones
```

- To display the quota setting for a file system, use `zfs get`, followed by `quota` and the file system name.

```
# zfs get quota rpool/export/home/jjones
```

NAME	PROPERTY	VALUE	SOURCE
rpool/export/home/jjones	quota	10.0G	local



Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ZFS quotas can be set and displayed by using the `zfs set` and `zfs get` commands. In the first example, a quota of 10 GB is set on `rpool/export/home/jjones`. In the second example, the results of space allocation is displayed.

**Note:** You cannot set a quota amount that is less than what is currently being used by a dataset.

## Setting Quotas for Users

- You can set a user or group quota on the amount of space consumed by the files that are owned by a user or group.
- To set a user quota on a file system, use `zfs set` followed by `userquota@<name>=`, the amount of space, and the file system name.

```
# zfs create students/compsci
# zfs set userquota@student1=10g students/compsci
```

- To display the user quota setting for a file system, use `zfs get` followed by `userquota@<name>` and the file system name.

```
# zfs get userquota@student1 students/compsci
```

NAME	PROPERTY	VALUE	SOURCE
students/compsci	userquota@student1	10g	local

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `quota` property in ZFS allows you to set a limit on the amount of space a file system can use. With user disk quotas, you can control the amount of space that a user can use on a file system. Given the bulk of user accounts, user quotas provide a way to easily manage disk space usage.

You can set a user quota by using the `zfs set userquota` command followed by the amount of space that you want to allocate to the file system and the file system name. In the first example shown in the slide, the file system `students/compsci` is created. Next, the user quota is set to 10 GB.

Note that the amount of space allocated for a home directory depends on the kind of files the user creates, their size, and the number of files that are created.

**Note:** ZFS also supports group quotas. To learn more about configuring group quotas, see the section titled “Managing Oracle Solaris ZFS File Systems” in *Oracle Solaris Administration: ZFS File Systems*.

## Displaying User Space Usage

- To display general user space usage, use `zfs userspace` followed by the file system name.

```
# zfs userspace students/compsci
TYPE          NAME      USED      QUOTA
POSIX User    root       227M     none
POSIX User    student1   455M     10g
```

- To display individual user space usage, use `zfs get userused@<name>` followed by the file system name.

```
# zfs get userused@student1 students/compsci
NAME                PROPERTY              VALUE      SOURCE
students/compsci    userused@student1     455M      local
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can identify individual user space usage by using the `zfs get` command followed by `userused@<name>` and the file system name as shown in the second example in the slide. The individual user space usage for the `students/compsci` file system is 455 MB.

**Note:** The user quota properties are not displayed by using the `zfs get all dataset` command that displays a listing of all file system properties.

## Removing User Quotas

To remove a user quota, use `zfs set userquota@<name>=none` followed by the file system name.

```
# zfs set userquota@student1=none students/compsci
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Quiz

You have downloaded the IPS repository images and have made it accessible from the `/mnt` directory. Given that the repository is bulky and likely to incrementally grow over time, which of the following commands would you use to optimize the space consumption of the target ZFS dataset `IPS/base`?

- a. `zfs set quota=10g IPS/base`
- b. `zfs set userquota@root=none IPS/base`
- c. `zfs set compression=on IPS/base`
- d. `zfs userspace IPS/base`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: c**

## Lesson Agenda

- Explaining the role of ZFS in data management
- Administering ZFS storage pools
- Managing devices in ZFS storage pools
- Administering ZFS file systems
- **Administering ZFS snapshots and clones**
- Securing ZFS file systems

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Administering ZFS Snapshots and Clones

Administering ZFS snapshots and clones involves the following activities:

- Creating and destroying ZFS snapshots
- Renaming, displaying, and rolling back ZFS snapshots
- Determining ZFS snapshot differences
- Sending and receiving ZFS snapshot data
- Replicating ZFS snapshot data remotely
- Creating and destroying ZFS clones

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Creating a ZFS Snapshot

- To create a snapshot, enter `zfs snapshot` followed by the snapshot name.

```
# zfs snapshot hrpool/home/reports@friday
```

- To create snapshots for all descendent file systems, use `zfs snapshot -r` and the snapshot name.

```
# zfs snapshot -r hrpool/home@now
# zfs list -t snapshot
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
hrpool/home@now	0	-	29.5K	-
hrpool/home/reports@now	0	-	2.15M	-
hrpool/home/reviews@now	0	-	1.89M	-
hrpool/home/jobdesc@now	0	-	1.89M	-
hrpool/home/bonus@now	0	-	2.15M	-

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

`zfs snapshot` takes as its only argument the name of the snapshot to create, which is specified as follows:

- `filesystem@snapname`
- `volume@snapname`

In the example shown in the slide, a `hrpool/home/reports@friday` snapshot is being created of `hrpool/home/reports`. This snapshot can now serve as a backup.

You can create snapshots for all descendent file systems by using the `zfs snapshot` command with the `-r` option followed by the snapshot name, as displayed in the example.

Using the `zfs list -t snapshot` command, you can display the snapshot information.

**Note:** Snapshots have no modifiable properties, and, therefore, dataset properties cannot be applied to a snapshot.



## Displaying a ZFS Snapshot

- To display snapshots, enter `zfs list -t snapshot`.

```
# zfs list -t snapshot
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
hrpool/home/reports@tuesday	18K	-	21K	-
hrpool/home/reports@wednesday	19K	-	280K	-
hrpool/home/reports@thursday	0	-	538K	-

- To list snapshots created for a specific file system, enter `zfs list -r -t snapshot` followed by the file system name.

```
# zfs list -r -t snapshot -o name,creation hrpool/home
```

NAME	CREATION
hrpool/home/reports@tuesday	Tue Nov 29 10:08 2013
hrpool/home/reports@wednesday	Wed Nov 30 08:05 2013
hrpool/home/reports@thursday	Thu Dec 1 07:03 2013
hrpool/home/bonus@now	Fri Dec 2 06:15 2013

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** You can enable or disable the display of snapshot listings in the `zfs list` output by using the `listsnapshots` pool property. This property is disabled by default. To enable, use `zpool set listsnapshot=on <poolname>`. If you disable this property, you must use the `zfs list -t snapshot` command to display snapshot information.

## Viewing Snapshot Space Accounting

- To display space consumed by snapshots and descendent file systems, use `zfs list -o space`.

```
$ zfs list -o space
```

NAME	AVAIL	USED	USED SNAP	USED DDS	USED REFRESERV	USED CHILD
rpool	25.4G	7.79G	0	64K	0	7.79G
rpool/ROOT	25.4G	6.29G	0	18K	0	6.29G
rpool/ROOT/sol11	25.4G	6.29G	0	6.29G	0	0
rpool/dump	25.4G	1.00G	0	1.00G	0	0
rpool/export	25.4G	38K	0	20K	0	18K
rpool/export/home	25.4G	18K	0	18K	0	0
rpool/swap	25.8G	512M	0	111M	401M	0

- When a snapshot is created, its space is:
  - Initially shared between the snapshot and the file system
  - Possibly shared with previous snapshots
- As the file system changes, previously shared space:
  - Becomes unique to the snapshot
  - Is counted in the snapshot's `used` property

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The output in this example shows available space in each file system, space being used, space being consumed by snapshots of each dataset (`USED SNAP`), space being used by the dataset itself (`USED DDS`), space being used by a reservation set on the dataset (`USED REFRESERV`), and space being used by the children of this dataset (`USED CHILD`).

**Note:** A snapshot's space `referenced` property is the same as the file system's was when the snapshot was created. Deleting snapshots can increase the amount of space that is unique to (and thus used by) other snapshots.

## Destroying a ZFS Snapshot

- To destroy a snapshot, use `zfs destroy` followed by the snapshot name.

```
# zfs destroy hrpool/home/reports@now
```

- When attempting to destroy a snapshot, know that:
  - The dataset cannot be destroyed if snapshots of it exist.
  - The clones created from a snapshot must be destroyed before the snapshot can be destroyed.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example in the slide, snapshot `hrpool/home/reports@now` is being destroyed.

## Renaming a ZFS Snapshot

- To rename a snapshot, use `zfs rename`.

```
# zfs rename hrpool/home/report@121011 hrpool/home/report@today
```

- To recursively rename snapshots, use `zfs rename -r` followed by the snapshot name.

```
# zfs list
NAME                                USED    AVAIL REFER  MOUNTPOINT
users                              270K    16.5G   22K      /users
users/home                          76K     16.5G   22K      /users/home
users/home@yesterday                0        -      22K      -
users/home/jjones                   18K     16.5G   18K      /users/home/jjones
users/home/jjones@yesterday         0        -      18K      -
# zfs rename -r users/home@yesterday @2daysago
# zfs list -r users/home
NAME                                USED    AVAIL REFER  MOUNTPOINT
users/home                          76K     16.5G   22K      /users/home
users/home@2daysago                0        -      22K      -
users/home/jjones                   18K     16.5G   18K      /users/home/jjones
users/home/jjones@2daysago         0        -      18K      -
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can rename a snapshot by using the `zfs rename` command followed by the snapshot name. In the example in the slide, the snapshot `report@121011` that resides in `hrpool/home` is renamed to `report@today`. Note that snapshots must be renamed within the same pool and data set from which they were created.

You can recursively rename snapshots with the `zfs rename -r` command followed by the snapshot name. In the example in the slide, only those snapshots that are named `@yesterday` are renamed to `@2daysago`.

You can see a before-and-after picture with the list of file systems before the `zfs rename -r` command is run in the first half of the example, and then the change that occurs in the second half of the example. In short, any snapshot with the `@yesterday` name is changed to `@2daysago` after the `rename` command is run.

## Rolling Back a ZFS Snapshot

- To discard all changes made since a specific snapshot or to restore the data of a specific snapshot, enter `zfs rollback` followed by the snapshot name.

```
# zfs rollback hrpool/home/qarpt@thursday
```

- By default, `zfs rollback` rolls back only to the most recent snapshot.
- To destroy more recent snapshots, enter `zfs rollback` with `-r`, followed by the snapshot name.

```
# zfs rollback -r hrpool/home/qarpt@tuesday
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use the `zfs rollback` command followed by the snapshot name to discard all the changes made since a specific snapshot. The `zfs rollback` command causes the file system to revert to its state at the time the snapshot was taken. In the example in the slide, the `hrpool/home/qarpt` file system is rolled back to the `thursday` snapshot.

By default, the `zfs rollback` command cannot roll back to a snapshot other than the most recent snapshot. To roll back to an earlier snapshot, you must destroy all intermediate snapshots. To do this, you must specify the `-r` option with the `zfs rollback` command followed by the snapshot name, as shown in the second example. Here, the `hrpool/home/qarpt` file system is rolled back to the `tuesday` snapshot. For this operation to take place, the `wednesday` and `thursday` snapshots must be destroyed.

# Identifying ZFS Snapshot Differences

To determine ZFS snapshot differences, use `zfs diff` followed by the snapshot names.

```
# zfs snapshot datapool/hrdata@before
# touch /datapool/hrdata/newfile
# rm /datapool/hrdata/oldfile
# zfs snapshot datapool/hrdata@after
# zfs list -r -t snapshot -o name,creation
NAME                                CREATION
datapool/hrdata@before              Thu Dec 13 14:54 2012
datapool/hrdata@after               Thu Dec 13 14:59 2012
rpool/ROOT/solaris@install          Tue Dec 18 22:33 2012
# zfs diff datapool/hrdata@before datapool/hrdata@after
M      /datapool/hrdata/
+      /datapool/hrdata/newfile
-      /datapool/hrdata/oldfile
#
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To determine the differences between ZFS snapshots, you can use the `zfs diff` command. The output of this command provides a high-level description of the differences between a snapshot and a descendant dataset. The descendant can be either a snapshot of the dataset or the current dataset. For each file that has undergone a change between the original snapshot and the descendant, the type of change is described along with the name of the file. In the case of a rename, both the old and new names are shown. The type of change follows any timestamp displayed and is described with a single character, M, -, +, and R. The definition of each of these characters is provided in the next slide.

In the example in the slide, a `before` snapshot of the `datapool/hrdata` ZFS file system was taken. A new file (`newfile`) was then created in `/datapool/hrdata`. Then another snapshot `after` of the same ZFS file system was taken. The `zfs list` command is used to list the snapshots based on name and creation date. The `zfs diff` command is then run to determine the differences between the `before` and `after` snapshots. The `M` in the `zfs diff` command output indicates that the `/datapool/hrdata/` directory has been modified. The `+` character indicates that a file `/datapool/hrdata/newfile` exists in the later snapshot. The `-` character indicates that a file or directory is present in the older snapshot but not in the newer snapshot. Finally, `R` indicates that the file or directory is renamed.

## Sending ZFS Snapshot Data

- To send a ZFS snapshot, enter `zfs send` followed by the snapshot name and destination.
- For example:
  - To send the snapshot stream on a different pool to the same system, use the following command:

```
# zfs send hrpool/data@snap1 | zfs recv spool/ds01
```

- To send the snapshot stream to a different system, pipe the `zfs send` output through the `ssh` command.

```
# zfs send hrpool/data@snap1 | ssh host2 zfs recv newpool/data
```

- To send incremental ZFS snapshot data, use `zfs send -i`.

```
# zfs send -i hrpool/data@snap1 hrpool/data@snap2 | ssh host2 \
zfs recv newpool/data
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use the `zfs send` command to send a copy of a snapshot and its underlying file system and receive the snapshot stream in another pool on the same system or in another pool on a different system that is used to store backup data. The `zfs send` command creates a stream representation of a snapshot that is written to standard output. By default, a full stream is generated. You can redirect the output to a file, to a different system, or to a device (for example, a magnetic tape).

To send a ZFS snapshot, use the `zfs send` command followed by the snapshot name and destination, as shown in the first example.

You can also send incremental data by using the `-i` option with the `zfs send` command, as shown in the second example. Note that the first argument is the earlier snapshot (`snap1`) and the second argument is the later snapshot (`snap2`).

## Receiving ZFS Snapshot Data

To receive a ZFS file system snapshot, use `zfs receive` followed by the snapshot name and the location from which you want to retrieve the file system.

```
# zfs send hrpool/jobdesc@1215 > /bkups/jobdesc.121511
# zfs receive hrpool/jobdesc2@today < /bkups/jobdesc.121511
# zfs rename hrpool/jobdesc hrpool/jobdesc.old
# zfs rename hrpool/jobdesc2 hrpool/jobdesc
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can receive ZFS data by using the `zfs receive` command. This command receives snapshot data from another pool on the same system or from another pool on a different system. It can also receive snapshot data from a file or device.

Consider the following points when a file system snapshot is received:

- The snapshot and the file system are received.
- The file system and all the descendant file systems are unmounted.
- The file systems are inaccessible while they are being received.
- The original file system to be received must not exist while it is being transferred.
- If a conflicting file system name exists, rename the file system.

In the example in the slide, the `hrpool/jobdesc@0930` snapshot of the `hrpool/jobdesc` file system is sent to the destination backup system called `/bkups/jobdesc.093011`.

Next, the `hrpool/jobdesc2@today` snapshot of the `hrpool/jobdesc2` file system is retrieved from the backup system. Then the `hrpool/jobdesc` file system is renamed to `hrpool/jobdesc.old` and the `hrpool/jobdesc2` file system is renamed to `hrpool/jobdesc`.



# Replicating ZFS Snapshot Data Remotely

To remotely copy (replicate) snapshot data from one system to another system, use `zfs send` and `zfs receive`.

```
# zfs send hrpool/report@today | ssh newsys zfs recv \  
sandbox/restfs
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

You can use the `zfs send` and `zfs receive` commands to remotely copy (replicate) a snapshot stream representation from one system to another system.

In the example in the slide, the `hrpool/report@today` snapshot data is sent, and it is received in the `sandbox/restfs` file system. This command also creates a `restfs@today` snapshot on the `newsys` system. In this example, the user has been configured to use `ssh` on the remote system.

ZFS also supports sending and receiving complex snapshot streams. For more information about remote replication of ZFS snapshot data and about sending and receiving ZFS data in general, see the appropriate sections of *Oracle Solaris Administration: ZFS File Systems*.

## Creating a ZFS Clone

- To create a clone, enter `zfs clone` followed by the snapshot name from which the clone is to be created, and the name of the new file system or volume.

```
# zfs snapshot hrpool/ws/gate@yesterday
# zfs clone hrpool/ws/gate@yesterday hrpool/home/reports/bug123
```

- The new file system or volume:
  - Can be located anywhere in the ZFS hierarchy
  - Has the same dataset type (for example, file system or volume) as the snapshot from which the clone was created

**Note:** A clone of a file system must be created in the same pool where the original file system snapshot resides.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the example in the slide, a clone named `hrpool/home/reports/bug123` is created with the same initial contents as the snapshot `hrpool/ws/gate@yesterday`. A clone, being a writable copy, can be used for testing purposes so that you do not affect the production version of the file system.

Note that a clone can be created only from a snapshot. An implicit dependency exists between the clone and the snapshot. The original snapshot cannot be destroyed as long as the clone exists.

- The `origin` property exposes this dependency.
- The `zfs destroy` command lists any such dependencies (if they exist).

A clone does not inherit the dataset properties of the dataset from which it was created. You can use the `zfs get` and `zfs set` commands to view and change the properties of a cloned dataset.

## Destroying a ZFS Clone

To destroy a clone, use `zfs destroy` followed by the clone name.

```
# zfs destroy hrpool/home/reports/bug123
```

**Note:** A clone must be destroyed before the parent snapshot can be destroyed.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Replacing a ZFS File System with a ZFS Clone

The clone replacement process is a two-step operation.

1. Replace an active ZFS file system by promoting the clone by using `zfs promote` followed by the clone name.

```
# zfs snapshot hrpool/reviews/q4@today
# zfs clone hrpool/reviews/q4@today hrpool/reviews/q4sum
# zfs promote hrpool/reviews/q4sum
# zfs list -r hrpool/reviews
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
hrpool/reviews	316K	8.24G	27.5K	/hrpool/reviews
hrpool/reviews/q4	0	8.24G	288K	/hrpool/reviews/q4
hrpool/reviews/q4sum	288K	8.24G	288K	/hrpool/reviews/q4sum
hrpool/reviews/q4@today	0	-	288K	-

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

ZFS provides the clone replacement (or promotion) feature, which enables you to:

- Replace an active ZFS file system with a clone of that file system
- Clone and replace file systems so that the original file system becomes the clone of the newly created file system
- Destroy the file system from which the clone was originally created

**Note:** Without clone promotion, you cannot destroy the original file system of active clones.

The clone replacement process is a two-step operation:

1. Replace an active ZFS file system by promoting the clone. In the example, the original `hrpool/reviews/q4` file system is replaced with the cloned file system, `hrpool/reviews/q4sum`, by promoting the clone.

## Replacing a ZFS File System with a ZFS Clone

2. Rename the promoted file systems to the original name, by using `zfs rename` followed by the current file system name and a new file system name.

```
# zfs rename hrpool/reviews/q4 hrpool/review/q4legacy
# zfs rename hrpool/reviews/q4sum hrpool/reviews/q4
# zfs list -r hrpool/reviews
```

NAME	USED	AVAIL	REFER	MOUNTPOINT
hrpool/reviews	316K	8.24G	27.5K	/hrpool/reviews
hrpool/reviews/q4	288K	8.24G	288K	/hrpool/reviews/q4
hrpool/reviews/q4@today	0	-	288K	-
hrpool/reviews/q4legacy	0	8.24G	288K	/hrpool/reviews/q4legacy

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

2. Rename the file systems. Use the `zfs rename` command followed by the current file system name and a new file system name. In the example in the slide, the original `hrpool/reviews/q4` file system is renamed to `hrpool/reviews/q4legacy` and the original clone file system, `hrpool/reviews/q4sum`, to `hrpool/reviews/q4`, which was the name of the original file system. In the `zfs list` output, observe that the name changes have taken effect. The original file system and snapshot can now be deleted.

## Quiz

Jack has a home directory, which is a mount point to the ZFS dataset `data/home/jack`. He is testing some scripts and is about to modify his home directory by running those scripts. Which of the following commands should Jack run to take a point-in-time backup of his home directory to which he can revert should the need for a rollback arise?

- a. `zfs backup data/home/jack@s_test`
- b. `zfs snapshot data/home/jack@s_test`
- c. `zfs rollback data/home/jack@s_test`
- d. `zfs send data/home/jack`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

## Lesson Agenda

- Explaining the role of ZFS in data management
- Administering ZFS storage pools
- Managing devices in ZFS storage pools
- Administering ZFS file systems
- Administering ZFS snapshots and clones
- **Securing ZFS file systems**

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Securing ZFS File Systems

You can secure ZFS storage pools and file systems by using the following two options:

- Delegated administration
- Data encryption

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Delegated Administration

- ZFS allows delegated administration, which enables you to distribute refined permissions to specific users, groups, or everyone.
- ZFS supports two types of delegated permissions:
  - Individual permissions such as `create`, `destroy`, `mount`, `snapshot`, and so on
  - Groups of permissions called permission sets

The Oracle logo, consisting of the word "ORACLE" in white, uppercase, sans-serif font, centered on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** A permission set can later be updated, and all of the consumers of the set automatically get the change. Permission sets begin with the @ symbol and are limited to 64 characters in length. After the @ symbol, the remaining characters in the set name have the same restrictions as normal ZFS file system names.

ZFS delegated administration provides features similar to the Role-Based Access Control (RBAC) security model. ZFS delegation provides the following advantages for administering ZFS storage pools and file systems:

- Permissions follow the ZFS storage pool whenever a pool is migrated.
- Dynamic inheritance is provided, where you can control how the permissions propagate through the file systems.
- It can be configured so that only the creator of a file system can destroy the file system.
- You can delegate permissions to specific file systems. Newly created file systems can automatically pick up permissions.
- Simple NFS administration is provided. For example, a user with explicit permissions can create a snapshot over NFS in the appropriate `.zfs/snapshot` directory.

For more information about Oracle Solaris ZFS delegated administration, refer to [http://docs.oracle.com/cd/E23824\\_01/html/821-1448/gbchv.html#scrolltoc](http://docs.oracle.com/cd/E23824_01/html/821-1448/gbchv.html#scrolltoc).

## Delegating ZFS Permissions

Use the `zfs allow` command to delegate permissions to non-root users for administering ZFS file systems in a ZFS storage pool, in the following ways:

- Delegate individual permissions to a user, group, or everyone.
- Delegate groups of individual permissions as a *permission set* to a user, group, or everyone.
- Delegate permissions either locally to the current file system only or to all descendents of the current file system.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Delegating ZFS Permissions

The following example shows how to set permissions so that the user `sandbox` can create, destroy, mount, and take snapshots on `mypool/home/sandbox`. The permissions on `mypool/home/sandbox` are also displayed.

```
# zfs allow sandbox create,destroy,mount,snapshot
mypool/home/sandbox
# zfs allow mypool/home/sandbox
-----
Local+Descendent permissions on (mypool/home/sandbox)
      user sandbox create,destroy,mount,snapshot
-----
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** The `mypool/home/sandbox` mount point permission is set to 755 by default. The user `sandbox` will be unable to mount file systems under `mypool/home/sandbox`. Set an ACL similar to the following syntax to provide mount point access:

```
# chmod A+user:sandbox:add_subdirectory:allow mypool/home/sandbox
```

## Disabling ZFS Delegated Permissions

You can control the delegated administration features by using a pool's `delegation` property, for example:

```
$ zpool get delegation users
NAME PROPERTY VALUE SOURCE
users delegation on default
$ zpool set delegation=off users
$ zpool get delegation users
NAME PROPERTY VALUE SOURCE
users delegation off local
```

**Note:** By default, the `delegation` property is enabled.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Removing ZFS Delegated Permissions

Use the `zfs unallow` command to remove previously delegated permissions, for example:

```
$ zfs unallow -s @pset snapshot pool/home
$ zfs allow pool/home
---- Permissions on pool/home -----
Permission sets:
@pset create,destroy,mount
Create time permissions:
Destroy
Local+Descendent permissions:
group staff @pset,create,mount
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The example provided in the slide shows how to remove the snapshot permission from the `@pset` permission set for the `staff` group on the `pool/home` file system. The permissions on `pool/home` are also displayed.

# Data Encryption

- Encryption is the process of encoding data for privacy and requires a key or a passphrase by the data owner to access the encoded data.
- Oracle Solaris 11 supports the following ZFS encryption features:
  - ZFS encryption is integrated with the ZFS command set.
  - ZFS encryption uses the Cryptographic Framework feature.
  - ZFS encryption is inheritable to descendent file systems.
  - Data is encrypted by using Advanced Encryption Standard (AES).

The Oracle logo, consisting of the word "ORACLE" in white, uppercase, sans-serif font, centered on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

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.
- 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.
- ZFS encryption is inheritable to descendent file systems. Key management can be delegated through ZFS delegated administration.
- Data is encrypted by using Advanced Encryption Standard (AES) with key lengths of 128, 192, and 256 in the CCM and GCM operation modes.

## Encrypting a ZFS Storage Pool and ZFS File System

- To encrypt a ZFS storage pool or a ZFS file system, change the `encryption` property to `on`.
  - For example, to encrypt the storage pool `hrpool`, use the following command:

```
# zpool create -O encryption=on hrpool c1t0d0
Enter passphrase for 'hrpool': <passphrase>
Enter again: <passphrase>
```

- For example. To encrypt the `hrpool/data` file system, use the following command:

```
# zfs create -o encryption=on hrpool/data
Enter passphrase for 'hrpool/data': <passphrase>
Enter again: <passphrase>
```

- The default encryption policy is to prompt for a *passphrase*, which must be a minimum of eight characters.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

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 more information about ZFS file encryption, refer to

[http://docs.oracle.com/cd/E23824\\_01/html/821-1448/gkkih.html#scrolltoc](http://docs.oracle.com/cd/E23824_01/html/821-1448/gkkih.html#scrolltoc).

## Summary

In this lesson, you should have learned how to:

- Describe the merits of using ZFS in data management
- Administer ZFS storage pools
- Manage devices in ZFS storage pools
- Administer ZFS file systems
- Administer ZFS snapshots and clones
- Secure ZFS file systems

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

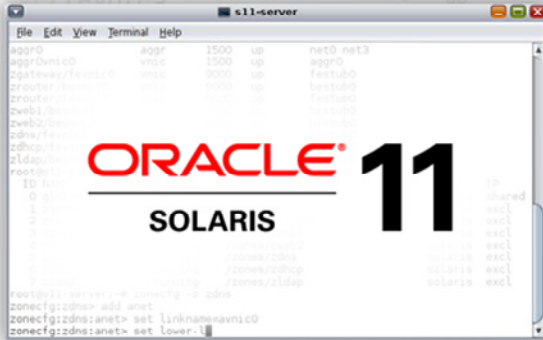


# 5

## Configuring the Network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## System Administration for Experienced UNIX/Linux Administrators



Administering System  
Software by Using IPS



Administering Services  
by Using SMF



Administering ZFS



Configuring the Network



Administering Oracle Solaris  
Zones



Administering Privileges  
and RBAC



Installing the Oracle Solaris 11  
Operating System



Monitoring System Resources

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Objectives

After completing this lesson, you should be able to:

- Describe the network stack in Oracle Solaris 11
- Configure a network interface
- Implement network virtualization
- Configure network high availability
- Implement resource management
- Secure the network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Agenda

- Describing the network stack in Oracle Solaris 11
- Configuring a network interface
- Implementing network virtualization
- Configuring network high availability
- Implementing resource management
- Securing the network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Networking in Oracle Solaris 11

- The network stack in Oracle Solaris 11 has been significantly re-architected to unify, simplify, and enhance the datacenter networking experience for large enterprises.
- The network stack in Oracle Solaris 11 makes network administration more flexible in the following ways:
  - The network configuration is insulated from any changes that might occur in the hardware layer.
  - The separation of the network configuration from the network hardware configuration also allows the use of customized link names in the datalink layer.
  - With the abstraction of the datalink layer, multiple networking abstractions or configurations are unified into a common administrative entity, which is the datalink.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a red rectangular background.

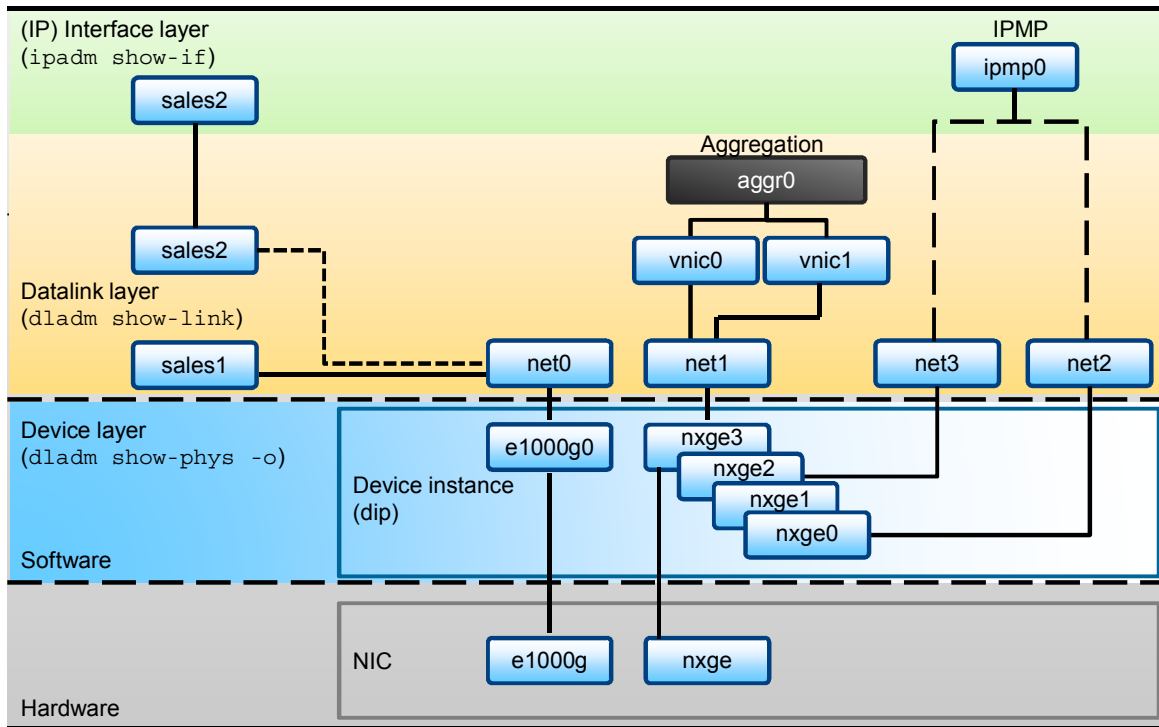
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A network stack refers to the network, Internet, and transport layers of the TCP/IP protocol stack. The network stack in Oracle Solaris 11 is different from the network stack in the previous Oracle Solaris implementation.

The network stack in Oracle Solaris 11 makes network administration more flexible in the following ways:

- The network configuration is insulated from any changes that might occur in the hardware layer. This implies that the link and interface configurations are preserved even if the underlying hardware is removed.
- The separation of the network configuration from the network hardware configuration also allows the use of customized link names in the datalink layer.
- With the abstraction of the datalink layer, multiple networking abstractions or configurations are unified into a common administrative entity, which is the datalink.

# Network Stack in Oracle Solaris 11



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

While the one-to-one relationship between the hardware, datalink, and interface layers remains intact, in Oracle Solaris 11, the software layer is decoupled from the hardware layer. With this separation, network configuration in the software layer is no longer bound to the chipset or the network topology in the hardware layer.

**Note:** For simplicity and uniformity, Oracle Solaris 11 uses `netx` as a vanity naming scheme for the Network Interface Card (NIC). The datalink has a corresponding IP interface, `net0`. This interface can be configured with IPv4 or IPv6 addresses to host both types of network traffic.

# Agenda

- Describing the network stack in Oracle Solaris 11
- **Configuring a network interface**
- Implementing network virtualization
- Configuring network high availability
- Implementing resource management
- Securing the network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Prerequisites for Configuring a Network

- Before you set out to configure a network, gather the following prerequisite information:
  - Host name
  - IP addresses and netmasks
  - IP address range
  - Domain name
  - Gateway address
  - Default router IP address
- Once you have the necessary system details and have established a preliminary connection with the server, the next task is to configure the network configuration profile.

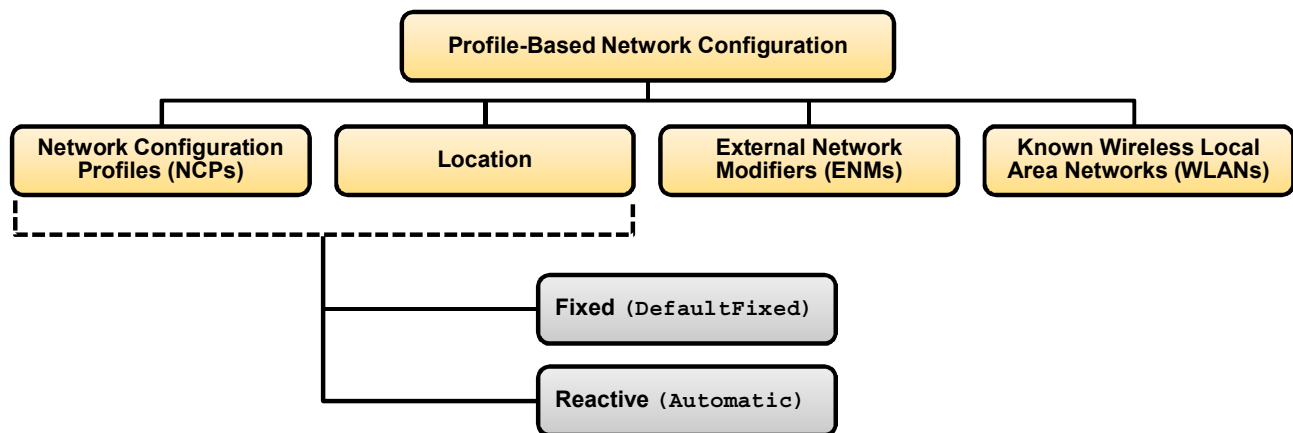
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



# Profile-Based Network Configuration

- Network configuration in Oracle Solaris 11 is based on profiles, which provides a more seamless and persistent network experience.
- The principal profile types are displayed in this chart:



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The principal profile types are:

- **Network Configuration Profiles (NCPs):** An NCP specifies the configuration of network links and interfaces.
- **Network Configuration Units (NCUs):** NCUs are the individual configuration objects that contain all of the properties that make up an NCP. The NCP is essentially a container that stores the NCUs that define it. Each NCU correlates to an individual link or interface in the system.
- **Location:** The location profile specifies system-wide network configuration (for example, the naming services, domain, IP Filter, and IPsec configuration).
- **External Network Modifiers (ENMs):** ENMs are profiles that are used to manage external applications, such as the VPN application.
- **Known Wireless Local Area Networks (WLANs):** Known WLANs are configuration objects that monitor and store information about wireless networks that are known to your system.

**Note:** Only one pair of NCP and location profile can be active at any given time to manage a system's network configuration. All other existing NCPs on the system are nonoperational.

## NCPs

- An NCP specifies the configuration of network links and interfaces.
- There are two types of NCPs in Oracle Solaris 11:
  - Fixed
  - Reactive
- The type of NCP that is operative on a specific system determines that system's network configuration.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Essentially, fixed configuration is designed for servers, whereas reactive configuration is designed for mobility, such as in the configuration of laptops. With the primary focus on mobility, a reactive configuration policy allows the system's configuration to be changed dynamically in response to different network events or at a user's request.

## Fixed NCP

- Fixed network configuration refers to the configuration mode where the network daemon instantiates a specific network configuration on the system, but does not automatically adjust that configuration to varying conditions.
- If changes in that environment occur, such as an addition of interfaces, you must manually reconfigure the system's network setup to have the system adopt to the new environment.
- To implement fixed networking, a fixed profile must be activated on the system.
- While a system can have multiple reactive profiles, only one fixed profile exists on a system.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Caution:** Do not confuse fixed network configuration with simply configuring static IP addresses. In fixed network configuration, you can assign a DHCP address to an interface. Likewise, in reactive network configuration, you can create NCPs where interfaces are configured with static IP addresses.

## Reactive NCP

- Reactive network configuration, formerly known as Network Auto-Magic (NWAM), refers to the configuration mode where the network daemon, `nwamd` allows a system to automatically adapt to any change in network conditions and network configuration without requiring manual reconfiguration.
- For example, if your wired network interface is unplugged or a new wireless network becomes available, the system adapts accordingly.
- You can create multiple reactive NCPs on a system to match the varying network setups for the system.
- However, only one NCP together with a corresponding location profile can be active at a time.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered within a solid red rectangular bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The system always defines an NCP called the Automatic NCP, which is the default reactive NCP. The Automatic NCP is created and maintained by the system and cannot be modified or removed. You can also create additional user-defined NCPs, as needed.

**Note:** The NWAM facility is provided by the `network/physical:default` SMF service. NWAM configuration management is enabled if the `netcfg/active_ncp` property is set to the name of a reactive NCP. Alternatively, traditional network configuration is performed if the `netcfg/active_ncp` property is set to `DefaultFixed`.

## Comparison Between Fixed and Reactive NCPs

Features	Reactive Network Configuration	Fixed Network Configuration
Automatically adapts to changes in the system's network environment	Supported by means of multiple NCPs that can be configured	Not supported. Requires manual reconfiguration as needed.
Type of NCP operative on the system	Reactive (Automatic, or some other user-created NCP)	Fixed (DefaultFixed)
Multiple NCPs	Supported (but only one NCP can be active at a time)	Not supported
User-created NCPs	Supported	Only one fixed NCP (DefaultFixed) exists, which is generated by the system. However, the contents of DefaultFixed is entirely determined by the user.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide presents a comparison between the two types of network configuration.

## Network Configuration and Administration Commands

The following four network-related commands in Oracle Solaris 11 allow you to configure and administer the network:

- `netcfg`
- `netadm`
- `dladm`
- `ipadm`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For more information about each of the commands listed, refer to the respective `man` pages.

## netcfg Command

- The `netcfg` command is used to configure the properties and values of network profiles.
- You use the `netcfg` command to create and configure profiles that implement reactive network configuration: NCPs, location profiles, ENMs, and WLANs.
- However, on a system with a fixed network configuration, you can use the `netcfg` command only to view the `DefaultFixed` profile.
- You can use the `ipadm` command to administer fixed NCPs.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## netadm Command

- The `netadm` command is used to administer all the profiles on the system, particularly to list the system's network profiles as well as to replace one active NCP with another.
- In addition, you can use the `netadm` command to interact with the network management daemon, `nwamd`, in the absence of a GUI.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Commands to Configure Profiles

Task	Command
Create reactive or user-defined profiles.	<code>netcfg create [ -t <i>template</i> ] <i>object-type</i> [ <i>class</i> ] <i>object-name</i></code>
Create a location profile.	<code>netcfg create loc office</code>
Enable or activate a profile.	<code>netadm enable</code>
Disable or deactivate a profile.	<code>netadm disable</code>
Remove all user-defined profiles or a specific user-defined profile.  <b>Note:</b> You cannot remove system-defined profiles. System-defined profiles include the <code>Automatic</code> and <code>DefaultFixed</code> NCPs and the <code>NoNet</code> , <code>Automatic</code> , and <code>DefaultFixed</code> location profiles.	<code>netcfg destroy [ -a ] <i>object-type</i> [ <i>class</i> ] <i>object-name</i></code>

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide contains a summary of commands that you can use for configuring network profiles.

For more information about the `netcfg` and `netadm` commands, refer to the `netcfg(1M)` and `netadm(1M)` man pages, respectively.

## Configuring and Administering Datalink and Network Interfaces

- You can use the `dladm` and `ipadm` commands to configure and administer the datalinks and network interfaces in a physical network.
- Note that the `dladm` and `ipadm` commands replace the `ifconfig` command for interface configuration and the `ndd` command for configuring protocol properties.
- During the Oracle Solaris 11 installation, the system undergoes a one-time upgrade to convert any existing `/etc` network configuration files to their respective `ipadm` and `dladm` configurations.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `ifconfig` command has been the customary tool to configure network interfaces. However, this command does not implement persistent configuration settings. Over time, `ifconfig` underwent enhancements for added capabilities in network administration. Consequently, the command became complex and confusing to use.

Another issue with interface configuration and administration is the absence of simple tools to administer the TCP/IP properties or tunables. The `ndd` command has been the prescribed customization tool for this purpose. However, like the `ifconfig` command, `ndd` does not implement persistent configuration settings.

Previously, persistent settings could be simulated for a network scenario by editing the boot scripts. With the introduction of Service Management Facility (SMF), using such workarounds can become risky because of the complexities of managing SMF dependencies.

## dladm Command

- The `dladm` command is used to administer the datalinks.
- The `dladm` command is implemented as a set of subcommands with corresponding options.
- Each `dladm` subcommand operates on one of the following objects:
  - Link
  - Bridge
  - Network device
  - Etherstub
  - InfiniBand (IB) partition link
  - Secure object

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A datalink is represented in the system as a STREAMS DLPI (v2) interface that can be plumbed under protocol stacks such as TCP/IP. Each datalink relies on either a single network device or an aggregation of devices to send packets to or receive packets from a network. The datalink layer is responsible for a number of activities, such as:

- **Framing:** The datalink layer divides the stream of bits received from the network layer into manageable data units called frames.
- **Physical addressing:** The datalink layer then adds a header to the frame to define the addresses of the sender and receiver of the frame.
- **Media access control and error control:** The datalink layer further controls how data is placed onto the media and is received from the media using techniques such as media access control and error control.
- **Flow control:** Finally, if the rate at which the data is received by the receiver is less than the rate at which data is transmitted by the sender, the datalink layer imposes a flow-control mechanism to avoid overwhelming the receiver.

## dladm Types/Classes

- Some of the subcommands of the `Link` object operate only on certain types or classes of datalinks.
- For such cases, the following object names are used:
  - `aggr-link`: Aggregation datalink
  - `ether-link`: Physical Ethernet datalink
  - `iptun-link`: IP tunnel link
  - `part-link`: IB partition data link
  - `phys-link`: Physical datalink
  - `vlan-link`: VLAN datalink
  - `vnic-link`: Virtual network interface that is created on a link or an etherstub
  - `wifi-link`: A WiFi datalink

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Administering Datalinks with `dladm` Commands

Task	Command
Show the physical device and attributes of all physical links.	<code>dladm show-phys</code>
Delete a datalink.	<code>dladm delete-phys</code>
Change the name of a datalink.	<code>dladm rename-link</code>
Display existing datalinks.	<code>dladm show-link</code>
Display the properties that are associated with the datalink.	<code>dladm show-linkprop</code>
Set specified datalink properties.	<code>dladm set-linkprop</code>
Restore properties to their default settings.	<code>dladm reset-linkprop</code>
Display Ethernet parameter settings of a datalink.	<code>dladm show-ether</code>

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide displays some of the `dladm` commands that are used for administering datalinks. For more information about the `dladm` command, refer to the `dladm(1M)` man page.

## ipadm Command

- The `ipadm` command is used to configure link and IP interfaces.
- The `ipadm` subcommands are used for:
  - Managing interfaces
    - Creating and deleting interfaces
    - Modifying interface properties
    - Displaying interface configuration
  - Managing addresses
    - Creating and deleting addresses
    - Modifying address properties
    - Displaying address configuration
  - Managing TCP/IP protocol properties
    - Modifying TCP/IP properties
    - Displaying TCP/IP properties

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

As a tool for configuring interfaces, the `ipadm` command offers the following advantages:

- Manages IP interfaces and IP addresses more efficiently
- Provides an option to implement persistent interface and address configuration settings

As a tool to set protocol properties, the `ipadm` command provides the following benefits:

- Sets temporary or persistent protocol properties for IP, Address Resolution Protocol (ARP), Stream Control Transmission Protocol (SCTP), and Internet Control Message Protocol (ICMP), as well as upper-layer protocols such as TCP and User Datagram Protocol (UDP).
- Provides information about each TCP/IP parameter, such as a property's current and default setting, as well as the range of possible settings. Thus, debugging information is more easily obtained.
- Has a consistent command syntax and, therefore, is easier to use.

## Administering Network Interfaces with the `ipadm` Command

Task	Command
Display network interface information.	<code>ipadm show-if</code>
Display IP address assignments to network interfaces.	<code>ipadm show-addr</code>
Create a network interface.	<code>ipadm create-ip interface</code>
Assign a static IP address to a network interface.	<code>ipadm create-addr -T address-type -a address/prefixlen addrobj</code>
Take down a network interface.	<code>ipadm down-addr addrobj</code>
Bring up a network interface.	<code>ipadm up-addr addrobj</code>
Delete an IP address assigned to a network interface.	<code>ipadm delete-addr addrobj</code>
Delete a network interface.	<code>ipadm delete-ip interface</code>

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide contains a summary of the `ipadm` commands that are used for configuring and administering the network interfaces. For more information about the `ipadm` command, refer to the `ipadm(1M)` man page.

Here is a simple example of using the network configuration and administration commands to configure a network interface.

```
# netadm enable -p ncp DefaultFixed
```

```
Enabling ncp 'DefaultFixed'
```

```
# netadm list
```

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

```

# dladm show-phys
LINK          MEDIA          STATE      SPEED  DUPLEX    DEVICE
net1          Ethernet        unknown   1000   full     e1000g1
net2          Ethernet        unknown   1000   full     e1000g2
net3          Ethernet        unknown   1000   full     e1000g3
net0          Ethernet        unknown   1000   full     e1000g0

# dladm show-link
LINK          CLASS      MTU      STATE    OVER
net1          phys      1500     unknown  --
net2          phys      1500     unknown  --
net3          phys      1500     unknown  --
net0          phys      1500     unknown  --

# ipadm show-if
IFNAME      CLASS      STATE      ACTIVE  OVER
lo0         loopback  ok         yes     --
# ipadm create-ip net0
# ipadm create-ip net1
# ipadm show-if
IFNAME      CLASS      STATE      ACTIVE  OVER
lo0         loopback  ok         yes     --
net0        ip        down       no      --
net1        ip        down       no      --

# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
lo0/v6       static    ok         ::1/128
# ipadm create-addr -T static -a 192.168.1.2/24 net0/v4
# ipadm create-addr -T dhcp net1/v4dhcp
ipadm: warning: : Communication with dhcpagent timed out
# ipadm show-addr
ADDROBJ      TYPE      STATE      ADDR
lo0/v4       static    ok         127.0.0.1/8
net0/v4       static    ok         192.168.1.2/24
net1/v4dhcp   dhcp      ok         ?
lo0/v6       static    ok         ::1/128

```



## Quiz

You need to switch from an existing automatic network configuration on your Oracle Solaris 11.1 server to the 192.168.172.201 IP address. Which of the following values would you set for the `netcfg/active_ncp` network service property?

- a. Automatic
- b. DefaultFixed
- c. Manual
- d. Reactive

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

# Agenda

- Describing the network stack in Oracle Solaris 11
- Configuring a network interface
- **Implementing network virtualization**
- Configuring network high availability
- Implementing resource management
- Securing the network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Network Virtualization

- Introduced in Oracle Solaris 11, network virtualization is an OS-provisioned mechanism that decouples the virtual network from the underlying physical network.
- A virtual network is, therefore, a pseudo network that only uses the physical network as an IP backplane.
- Although a pseudo network, the virtual network offers the same capabilities as that of a physical network and much more, such as hardware independence and scaling.
- Being an OS-supported technique, virtual networks are programmatically created and configured.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The following are the advantages of network virtualization:

- Network virtualization provides systems and users with efficient, controlled, and secure sharing of resources.
- Virtual networks offer a huge degree of flexibility and control irrespective of the geography or interface type.

## Components of a Virtual Network

- The following are the building blocks of a virtual network or network-in-a-box setup in Oracle Solaris 11:
  - Network Interface card (NIC)
  - Virtual NIC (VNIC)
  - Etherstub
- While network virtualization can be implemented in a single instance of an Oracle Solaris 11 OS, it is within the following infrastructures that one realizes the full potential of network virtualization:
  - Oracle Solaris Zones
  - Oracle VM for SPARC (formerly called LDoms)

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a solid red rectangular background.

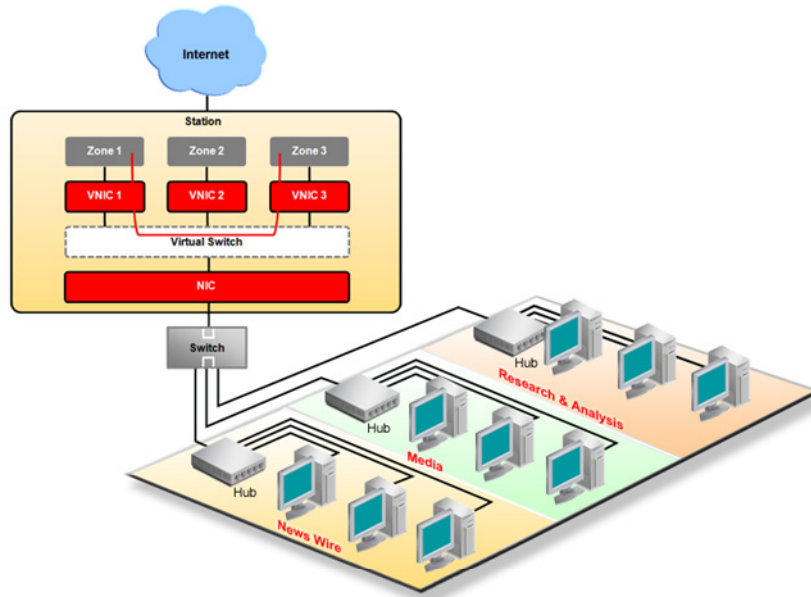
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The following are the building blocks of a virtual network in Oracle Solaris 11:

- **NIC:** A NIC refers to a physical network interface card.
- **VNIC:** A VNIC is a virtual network device with the same datalink interface as a physical NIC. You configure a VNIC over an underlying datalink. When a VNIC is configured, it behaves like a physical NIC. In addition, the system's resources treat a VNIC as if it were a physical NIC. A VNIC has an automatically generated MAC address. Note that you can also manually assign a MAC address.
- **Etherstubs:** Etherstubs are pseudo Ethernet NICs that are managed by the system administrator. You can create VNICs over etherstubs instead of over physical links. VNICs over an etherstub become independent of the physical NICs in the system. With etherstubs, you can construct a private virtual network that is isolated both from the other virtual networks in the system and from the external network.

## Network Virtualization in Zones

By combining NICs, VNICs, and etherstubs and deploying them within zones, you can create a network-in-a-box setup.



ORACLE

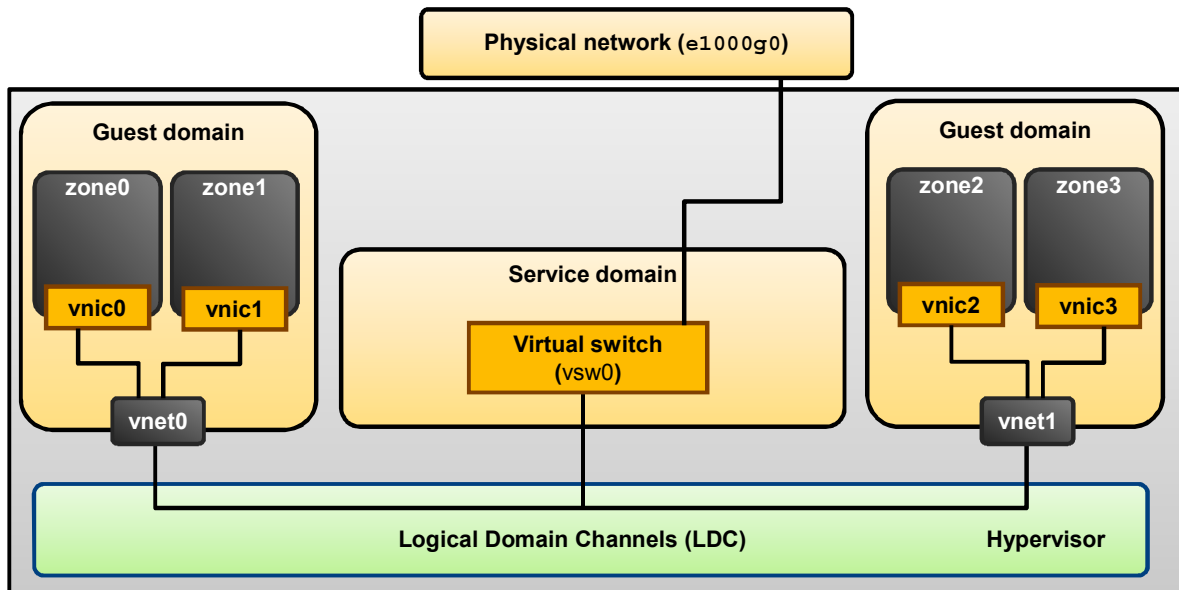
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Oracle Solaris Zones benefited the most from the introduction of VNICs in Oracle Solaris 11. The default `ip-type` configuration of zones was changed from a shared IP stack in Oracle Solaris 10 to an exclusive IP stack in Oracle Solaris 11. Now, with the capability of creating VNICs from physical interface cards, and each VNIC being as good as a physical interface card, you can apply the principles of networking at the zone level.

As the diagram indicates, the single system has one physical NIC. The NIC is configured with three VNICs. Each VNIC supports a single zone. Therefore, Zone 1, Zone 2, and Zone 3 are configured to use VNIC 1, VNIC 2, and VNIC 3, respectively. The three VNICs are virtually connected to a virtual switch. This switch provides the connection between the VNICs and the physical NIC on which the VNICs are built. The physical interface provides the system with its external network connection.

## Network Virtualization in LDOMs

A virtual network allows domains to communicate with each other without using any external physical networks.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The virtual switch (vsw0) created from a physical interface (e1000g0) and the virtual networks (vnet0 and vnet1) are the inherent components of network virtualization in the Oracle VM for SPARC (LDoms) infrastructure.

Network virtualization as a feature of the Oracle Solaris OS becomes manifest when a virtual network (vnet) can spawn VNICs that can be assigned to zones created inside the guest domains.

# Configuring and Administering Virtual Networks

- After the network configuration profile has been set, you can proceed with configuring and administering a virtual network.
- Much of the network virtualization features are configured and managed by using the `dladm` and `ipadm` commands.
  - Objects that are in the link layer (Layer 2) of the network stack, such as VLANs, tunnels, link aggregations, and VNICs, are configured by using the `dladm` command.
  - Interfaces that are on the IP layer (Layer 3) are configured by using the `ipadm` command.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Creating a Virtual Network

There are essentially two parts to creating a virtual network. One pertaining to the creation of VNICs and the other about assigning the VNICs to the zone.

1. Create the VNIC by using `dladm create-vnic`.
2. Create the zone by using `zonecfg -z <zonename>`.
3. Install the zone by using `zoneadm install <zonename>`.
4. Start the zone by using `zoneadm boot <zonename>`.
5. After the zone completely boots up, connect to the zone's console by using `zlogin -C`.
6. Provide the information as you are prompted, after which the zone is restarted.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

As you configure a virtual network, ensure that you have planned your zone configuration. As an example, this procedure illustrates how the VNIC is implemented inside a zone. Oracle Solaris Zones is covered in detail in “Lesson 6: Administering Oracle Solaris Zones.” However, network virtualization in LDoms is not in the scope of this course. For more information about implementing network virtualization in LDoms, refer to documentation.

**Note for Step 2:** The `ip-type` parameter is set to `exclusive` by default.

**Note for Step 5:** After you connect to the zone's console by using `zlogin -C`, you are prompted for the terminal type, region, language, and so on. Most of the information is supplied by selecting from a list of choices. Typically, the default options suffice unless your system configuration requires otherwise. After you have provided the required information for the zone, the zone is restarted.

**Note:** You can add the IP address information in the `/etc/hosts` file, if the network configuration profile is `DefaultFixed`, for local hostname-to-IP resolution.



# Administering Virtual Networks

Task	Command
Modify a single VNIC.	<code>dladm modify-vnic -v vid -L datalink</code>
Change the unique VIDs of multiple VNICs over a single datalink.	<code>dladm modify-vnic -v vid vnic</code>
Obtain information about VNICs.	<code>dladm show-vnic</code>
Obtain property information about the VNIC.	<code>dladm show-linkprop [-p property] vnic</code>
Set the link property of a VNIC.	<code>dladm set-linkprop -p &lt;prop=value&gt; vnic</code>
Determine the links used by a zone.	<code>dladm show-link</code>
Remove or detach the VNIC from the zone.	<code>zonecfg -z zone remove net physical=vnic</code>
Remove the VNIC from the system.	<code>dladm delete-vnic vnic</code>

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide contains a summary of commands that are used for administering virtual networks. For more information about the `dladm` command, refer to the `dladm(1M)` man page.

## Migrating a VNIC

- Migrating a VNIC is a new feature in Oracle Solaris 11.1 that enables you to move one or more VNICs from one underlying datalink to another without deleting and reconfiguring the VNICs.
- You can migrate VNICs globally or selectively.
  - In global migration, you migrate all the VNICs. This example moves all the VNICs from `ether0` to `net1`:

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

- In selective migration, you migrate a specific VNIC. This example moves the selected VNICs from their existing datalink to `net1`:

```
# dladm modify-vnic -l net1 vnic0,vnic1,vnic2
```

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The underlying link can be a physical link, a link aggregation, or an etherstub. To successfully migrate VNICs, the underlying datalink to which the VNICs are moved must be able to accommodate the datalink properties of the VNICs. If those properties are not supported, then the migration fails and the user is notified. After a successful migration, all the applications that use the VNICs continue to operate normally.

Know that certain hardware-dependent properties might change after a VNIC migration, such as datalink state, link speed, and MTU size. The values of these properties are inherited from the datalink to which the VNICs are migrated. You further have the option of migrating VNICs globally or selectively.

**Caution:** You cannot migrate a VNIC after it is assigned to a zone.

## Quiz

The `ip-type` property of Oracle Solaris Zones is set to shared by default in Oracle Solaris 11.1.

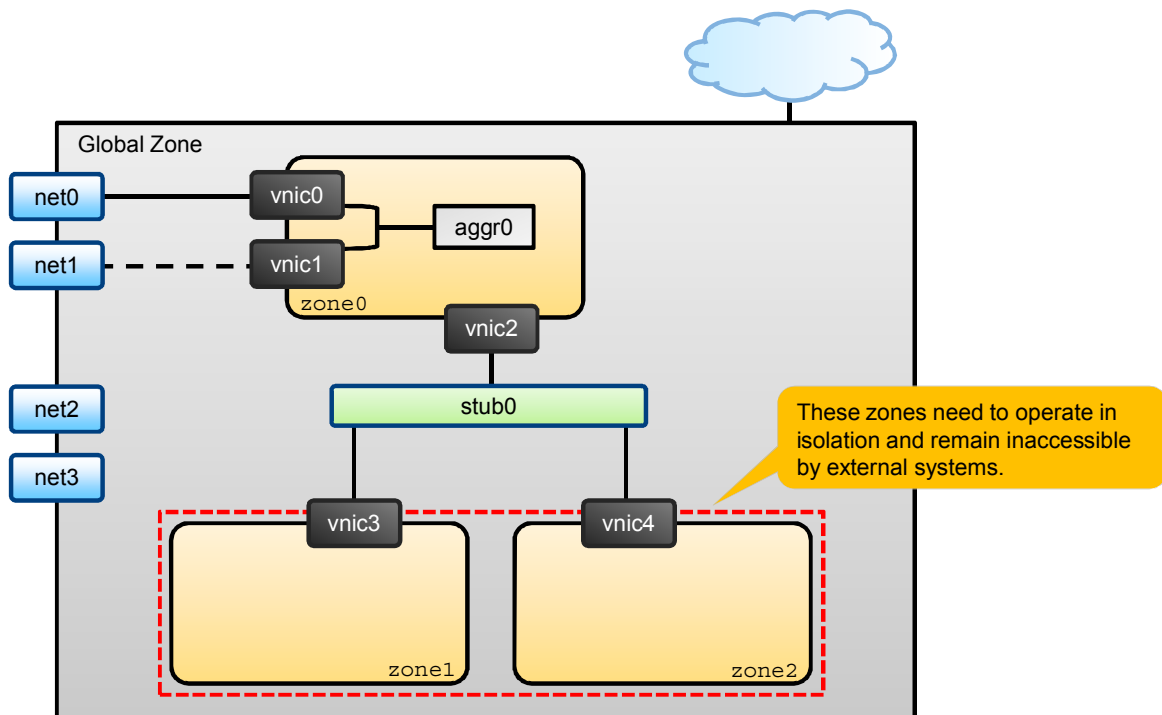
- a. True
- b. False

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**

# Private Virtual Network



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In nearly all enterprises, portions of a larger network must be shielded both from the other virtual networks in the system and from the external network. Consider a Market Research sub-department within the larger Research and Analysis department in an organization. Given the criticality of the roles and responsibilities of the employees who work in such a sub-department, it would be strategically befitting for this sub-department to operate in isolation. Private virtual networks do exactly that.

## Features of a Private Virtual Network

- A private virtual network provides another level of isolation within the scope of an internal virtual network.
- A private virtual network can be isolated both from the other virtual networks in the system and from the external network.
- A private virtual network can be created by using etherstubs, also called private virtual switches.
- Etherstubs are pseudo Ethernet NICs that are managed by the system administrator.
- You can create VNICs over etherstubs instead of creating them over physical links.
- VNICs over an etherstub become independent of the physical NICs in the system.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Use of etherstubs constitutes the main difference between an internal virtual network and a private virtual network. In a private virtual network, the VNICs that are assigned to the zones are configured over an etherstub. Thus, they are isolated from network traffic that flows into the system.

However, if you want the isolated network to send network traffic beyond the system, then you must use network address translation (NAT). NAT translates the VNIC's private IP addresses to routable IP addresses of the physical network interface, but without exposing the private IP addresses to the external network.

**Caution:** Although NAT is one way, it requires you to create an additional VNIC in the global zone. VNICs plumbed in the global zone can also use the routing table, if forwarding is turned on, to access the physical network.

## Creating a Private Virtual Network

This procedure creates a private virtual network:

1. Create the etherstub by using the `dladm create-etherstub <etherstub>` command.
2. Create a VNIC over the etherstub by using the `dladm create-vnic -l <etherstub> <vnic>` command.
3. Create the zone by using `zonecfg -z <zonenumber>`.
4. Install the zone by using `zoneadm install <zonenumber>`.
5. Start the zone by using `zoneadm boot <zonenumber>`.
6. Log in to the zone by using the `zlogin zone` command.
7. Provide the information as you are prompted, after which the zone is restarted.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Establishing Communication Between Networks

- To establish communication from the zones in the private network to the outside world, you require router functionality.
- Oracle Solaris 11 provides a network protocol attribute that implements the router functionality in the zone designated to assume the role of a router.

```
# ipadm set-prop -p forwarding=on ipv4
# ipadm show-prop -p forwarding ipv4
```

PROTO	PROPERTY	PERM	CURRENT	PERSISTENT	DEFAULT
ipv4forwarding	rw	on	--	off	on,off

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Quiz

You would like to establish a private communication channel for the four zones running in your system. Which of the following commands would you use to create a private virtual switch that can provision network interfaces for the zones?

- a. `dladm create-vnic -l net0`
- b. `dladm create-etherstub <etherstub>`
- c. `ipadm set-prop -p forwarding=on ipv4`
- d. `dladm create-vnic -l <etherstub>`

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b**



# Agenda

- Describing the network stack in Oracle Solaris 11
- Configuring a network interface
- Implementing network virtualization
- **Configuring network high availability**
- Implementing resource management
- Securing the network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# High Availability

- High availability (HA) refers to a state where a point of failure is taken over by a redundant system to ensure business continuity.
- HA is architected into various domains:
  - Server
  - Storage
  - Applications
  - Network
- Network-level failover and load sharing mechanisms are:
  - IP Multipathing (IPMP)
  - Link aggregation
  - Integrated Load Balancer (ILB)
  - Virtual Router Redundancy Protocol (VRRP)

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered on a red rectangular background.

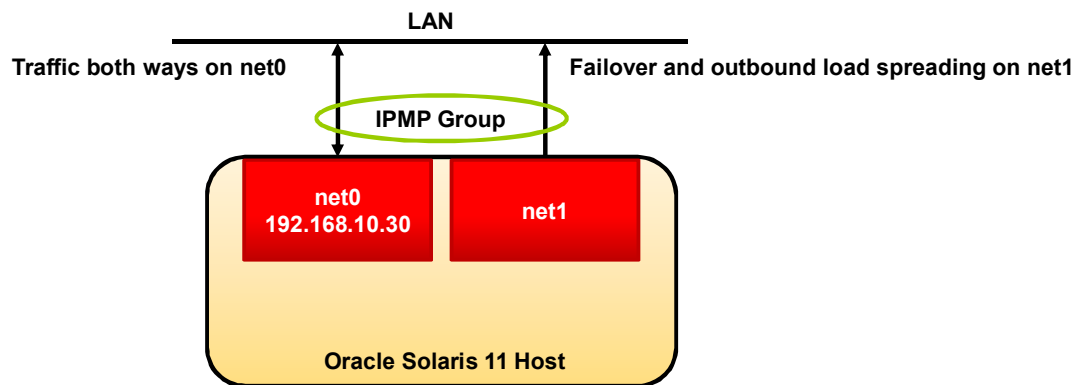
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

- ILB provides Layer 3 and Layer 4 load-balancing capabilities on SPARC and x86-based Oracle Solaris systems. ILB intercepts incoming requests from clients, decides which back-end server should handle the request based on load-balancing rules, and then forwards the request to the selected server.
- VRRP helps provide failure recovery for a router by introducing virtual routers into the network. VRRP is an Internet standard protocol specified in Virtual Router Redundancy Protocol Version 3 for IPv4 and IPv6 and is supported in Oracle Solaris 11 to provide high availability at the router level.

For more information about ILB and VRRP, refer to the *Oracle Solaris 11 Network Administration* course. This lesson introduces you to IPMP and link aggregation.

## Overview of IPMP

- IPMP is an Internet or Layer 3 technology that enables you to group multiple IP interfaces into a logical group.
- The group functions similar to an IP interface with data addresses to send or receive network traffic.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

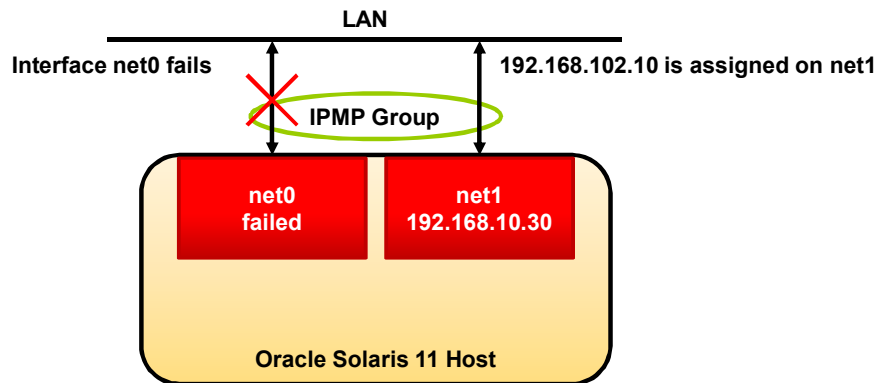
Different factors can cause an IP interface to become unusable, such as an interface failure. In such situations, the system can no longer be contacted by using any of the associated IP addresses.

IPMP enables you to configure multiple IP interfaces into a single group called an IPMP group. As a whole, the IPMP group with its multiple underlying IP interfaces is represented as a single IPMP interface. This interface is treated like any other interface on the IP layer of the network stack. All IP administrative tasks, routing tables, Address Resolution Protocol (ARP) tables, firewall rules, and other IP-related procedures work with an IPMP group by referring to the IPMP interface.

**Note:** The IPMP interface can be assigned a customized name to identify the IPMP group more easily.

## Overview of IPMP

- If an underlying interface in the group fails, the data addresses are redistributed among the remaining underlying active interfaces in the group.
- This address redistribution helps the group maintain network connectivity despite an interface failure.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

IPMP provides the following benefits:

- With IPMP, network connectivity is always available, provided at least one interface is usable for the group.
- With features such as failure detection, transparent access failover, and packet load spreading, IPMP improves network performance by ensuring that the network is always available to the system.
- IPMP improves overall network performance by automatically spreading outbound network traffic across the set of interfaces in the IPMP group, a process called outbound load spreading.
- The system also indirectly controls inbound load spreading by performing source address selection for packets whose IP source address was not specified by the application.
- However, if an application has explicitly chosen an IP source address, then the system does not vary that source address.
- With the ability to customize link names, link configuration is no longer bound to the physical NIC, which means greater flexibility in administering IP interfaces and IPMP itself.

# IPMP Components

IPMP consists of the following software components:

- IPMP daemon (`in.mpathd`)
- IPMP service (`svc:/network/ipmp`)
- Configuration file (`/etc/default/mpathd`)
- IPMP administration command (`ipadm`)
- IPMP display information command (`ipmpstat`)

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

IPMP consists of the following software components:

- **IPMP daemon:** Detects failure by monitoring the `RUNNING` flag on the interface, and can be configured to detect failures by sending ICMP echo probes through the interface.
- **IPMP service:** Manages the IPMP daemon. The IPMP service also sets IPMP properties, such as enabling or disabling transitive probing.
- **Configuration file:** Specifies the daemon's default behavior. This file can be used to set parameters, such as specifying which interfaces to probe for failure and the time duration. This file can also be used to specify what the status of a failed interface should be after it is repaired, or whether to monitor all interfaces, including those not belonging to an IPMP group.
- **IPMP administration command:** Administers IP interfaces of the IPMP group.
- **IPMP display information command:** Provides information about the status of IPMP.

## Types of IPMP Configurations

- An IPMP configuration consists of two or more physical interfaces on the same system that are attached to the same network.
- These interfaces can belong to an IPMP group in either of the following configurations:
  - Active-active
  - Active-standby

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

An IPMP configuration consists of two or more physical interfaces on the same system that are attached to the same LAN. These interfaces can belong to an IPMP group in either of the following configurations:

- **Active-active:** An IPMP group in which all underlying interfaces are active. An active interface is an IP interface that is currently available for use by the IPMP group. By default, an underlying interface becomes active when you configure the interface to become a part of an IPMP group.
- **Active-standby:** An IPMP group in which at least one interface is administratively configured as a standby interface. Although idle, the standby interface is monitored by the multipathing daemon to track the interface's availability, depending on how the interface is configured. If link-failure notification is supported by the interface, link-based failure detection is used. If the interface is configured with a test address, probe-based failure detection is used. If an active interface fails, the standby interface is automatically deployed as needed. You can configure as many standby interfaces as you want for an IPMP group.

## Failure and Repair Detection in IPMP

- To ensure continuous availability, IPMP performs failure detection on the IPMP group's underlying IP interfaces.
- Failed interfaces remain unusable until they are repaired.
- Remaining active interfaces continue to function while any existing standby interfaces are deployed as needed.
- The `in.mpathd` daemon handles the following types of failure detection:
  - Probe-based failure detection
    - No test addresses are configured (transitive probes)
      - ICMP probes
      - Sends ICMP probes to the other interfaces in the ICMP group
    - Test addresses are configured
      - Sends ICMP probes to targets on the local network
  - Link-based failure detection

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `in.mpathd` daemon handles the following types of failure detection:

### Probe-based failure detection:

- **No test addresses are configured:**
  - **ICMP probes:** The ICMP probe uses the data address as the probe's source address. ICMP probes are sent by the active interfaces in the IPMP group to probe targets that are defined in the routing table.
  - **Transitive probes:** Transitive probes are sent by the alternate interfaces in the group to probe the active interface. An alternate interface is an underlying interface that does not actively receive any inbound IP packets.
- **Test addresses are configured:** This failure detection method involves sending and receiving ICMP probe messages that use test addresses. These messages, also called probe traffic or test traffic, go out over the interface to one or more target systems on the same local network. The daemon probes all the targets separately through all the interfaces that have been configured for probe-based failure detection. If no replies are received in response to five consecutive probes on a given interface, `in.mpathd` considers the interface to have failed.

**Link-based failure detection:** The network drivers that support link-based failure detection monitor the interface's link state and notify the networking subsystem when that link state changes.

# Configuring and Administering an IPMP Group

Configuring and administering an IPMP group involves the following tasks:

- Creating an IPMP group
- Adding IP addresses to an IPMP group
- Moving an interface from one IPMP group to another
- Deleting or disabling an IPMP group

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Prior to Oracle Solaris 11, IPMP was configured by using the `/etc/hostname.<interface>` file and the `/usr/sbin/ifconfig` command. With the introduction of the `ipadm` command in Oracle Solaris 11, both the file and the command are obsolete. The `ipadm` command is delivered by the `system/network` package.



## Creating an IPMP Group

1. Create IP interfaces for the datalinks to use in the IPMP group by using the `ipadm create-ip` command.
2. Create the IPMP group by using the `ipadm create-ipmp` command.

```
# dladm rename-link net0 link0_ipmp0
# dladm rename-link net1 link1_ipmp0
# ipadm create-ip link0_ipmp0
# ipadm create-ip link1_ipmp0
# ipadm create-ipmp ipmp0
# ipadm add-ipmp -i link0_ipmp0 -i link1_ipmp0 ipmp0
# ipmpstat -g
```

GROUP	GROUPNAME	STATE	FDT	INTERFACES
ipmp0	ipmp0	ok	--	link1_ipmp0 link0_ipmp0

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

To create an IPMP group, follow the steps listed in the slide. In this example, two IP interfaces are being created for datalinks `net0` and `net1`.

The datalinks `net0` and `net1` are renamed to `link0_ipmp0` and `link1_ipmp0` just to make it look more obvious that they are part of an IPMP group. This is done by using the `dladm rename-link` command.

Then an IPMP group is created called `ipmp_group0` with the `ipadm create-ipmp` command. Finally, the specified network interfaces are added to the IPMP group. Finally, running the `ipmpstat -g` command displays the group information.

## Commands to Administer an IPMP Group

After you have created an IPMP group, you can use the commands listed below to administer the IPMP group.

Task	Command
Add addresses to an IPMP group.	<code>ipadm create-addr</code>
Display address information of the IPMP group.	<code>ipadm show-addr</code>
Remove the interface from the IPMP group.	<code>ipadm remove-ipmp</code>
Add network interfaces to the IPMP group.	<code>ipadm add-ipmp</code>
Delete an IPMP group.	<code>ipadm delete-ipmp</code>
Disable an IPMP group.	<code>ipadm disable-if</code>
Monitor an IPMP group.	<code>Ipmstat</code>

**ORACLE**

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Quiz

Which of the following comprise the IPMP software components?

- a. NWAM daemon, `nwamd`
- b. Multipathing daemon, `in.mpathd`
- c. IPMP configuration file, `/etc/default/mpathd`
- d. `dladm` command
- e. `ipmpstat` command

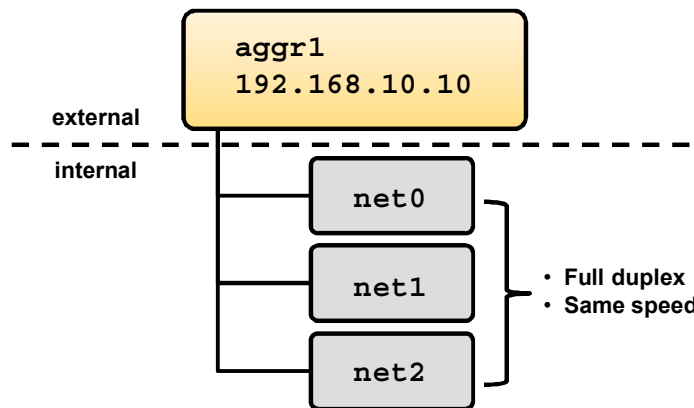
ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: b, c, e**

## Overview of Link Aggregation

- Link aggregation, also called trunking, allows multiple NICs to be grouped together into a single logical interface.
- Link aggregations are useful for increasing bandwidth as well as providing HA.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In the diagram, a link aggregation named `aggr1` has an exposed IP address of `192.168.10.10`, but the internal interfaces `net0`, `net1`, and `net2` are not seen externally. The Link Aggregation Standard states that all links must be full-duplex and operate at identical speeds. Link aggregation provides the following benefits:

- **Increased bandwidth:** The capacity of multiple links is combined into one logical link.
- **Automatic failover and fallback:** By supporting link-based failure detection, traffic from a failed link is failed over to the other working links in the aggregation.
- **Improved administration:** All underlying links are administered as a single unit.
- **Link protection:** You can configure the `datalink` property that enables link protection for packets flowing through the aggregation.
- **Resource management:** `Datalink` properties for network resources as well as flow definitions enable you to regulate the applications' use of network resources.

## Link Aggregation Types

Based on single or multiple switch capability, link aggregation can be of the following two types:

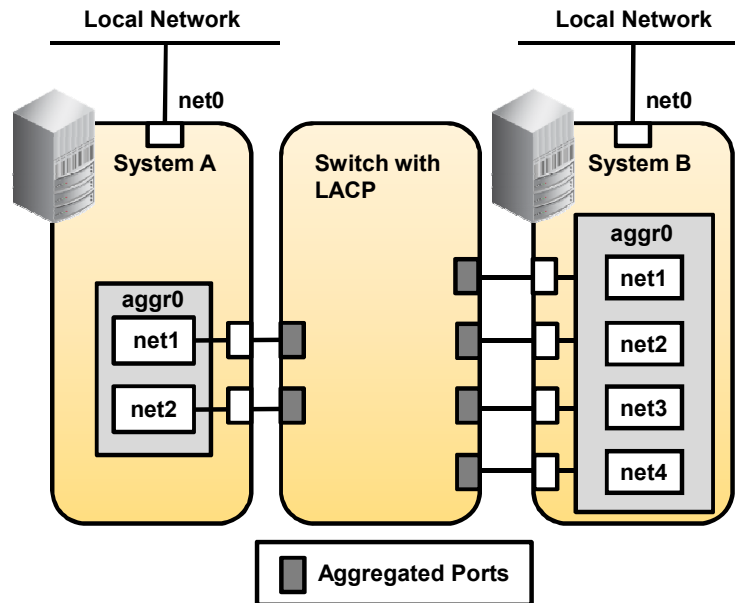
- Trunk aggregation: Works only with a single switch
- Datalink Multipathing (DLMP) aggregation: Spans multiple switches
  - For DLMP aggregation, no switch side configurations are required.
  - Switches are, therefore, unaware of the link aggregation and treat each port individually.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Trunk Aggregation

In Oracle Solaris 11, trunk aggregations are configured by default when you create an aggregation.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The figure in the slide depicts a local network with two systems, and each system has an aggregation configured. The two systems are connected by a switch on which the link aggregation control protocol (LACP) is configured. System A has an aggregation that consists of two interfaces, `net1` and `net2`. These interfaces are connected to the switch through aggregated ports. System B has an aggregation of four interfaces, `net1` through `net4`. These interfaces are also connected to aggregated ports on the switch. In this link aggregation topology, the switch must support IEEE 803.2ad. Accordingly, its switch ports must be configured to manage the traffic from the systems.

## LACP for Trunk Aggregation

- If the switch in your trunk aggregation setup supports LACP, you must configure LACP for the switch and the aggregation.
- LACP provides a method to group multiple ports to form a single virtual link.
- By distributing the traffic load over multiple physical links, LACP helps increase the bandwidth between network devices.
- The aggregation's LACP can be set to one of the following values:
  - off
  - active
  - passive

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The aggregation's LACP can be set to one of the three values:

- **off:** The default mode for aggregations. In this mode, LACP packets, which are called Link Aggregation Control Protocol Data Units (LACPDU), are not generated.
- **active:** The system generates LACPDUs at regular intervals, which you can specify.
- **passive:** The system generates an LACPDU only when it receives an LACPDU from the switch. When both the aggregation and the switch are configured in the passive mode, they cannot exchange LACPDUs.

## Policies in Trunk Aggregation

- If you plan to use a trunk aggregation, consider defining a policy for outgoing traffic.
- This policy specifies how you want packets to be distributed across the available links of an aggregation, thus establishing load balancing.
- You can define the following policies:
  - L2: Determines the outgoing link by hashing the MAC (L2) header of each packet
  - L3: Determines the outgoing link by hashing the IP (L3) header of each packet
  - L4: Determines the outgoing link by hashing the TCP, UDP, or other ULP (L4) header of each packet

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

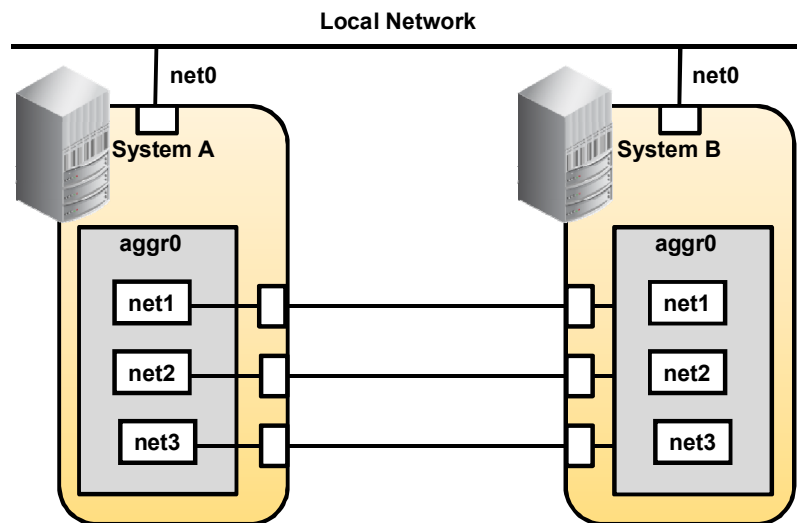
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** Any combination of these policies is also valid. The default policy is L4.



## Back-to-Back Configuration in Trunk Aggregation

Trunk aggregations also support back-to-back configuration, where instead of using a switch, two systems can be directly connected to run parallel aggregations.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The figure shows link aggregation `aggr0` on System A directly connected to link aggregation `aggr0` on System B by using the corresponding links between their respective underlying datalinks. This way, Systems A and B provide redundancy and high availability, as well as high-speed communications between both systems. Each system also has `net0` configured for traffic flow within the local network.

The most common application for back-to-back link aggregations is the configuration of mirrored database servers in data centers. Both servers must be updated together and, therefore, require significant bandwidth, high-speed traffic flow, and reliability.

**Note:** Back-to-back configurations are not supported in DLMP aggregations.

## Quiz

If you have a trunk aggregation setup, you must configure LACP.

- a. True
- b. False

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: a**

## DLMP Aggregation

- Unlike in a trunk aggregation, where the switch becomes the single point of failure (SPOF), a DLMP aggregation can fail over to an alternative switch.
- This is because a DLMP aggregation can span multiple switches.
- In DLMP, no switch configuration is required.
- You can switch between a trunk aggregation and a DLMP aggregation by using the `dladm modify-aggr` command.

**Note:** If you switch from a trunk aggregation to a DLMP aggregation, you must remove the switch configuration that was previously created for the trunk aggregation.

ORACLE

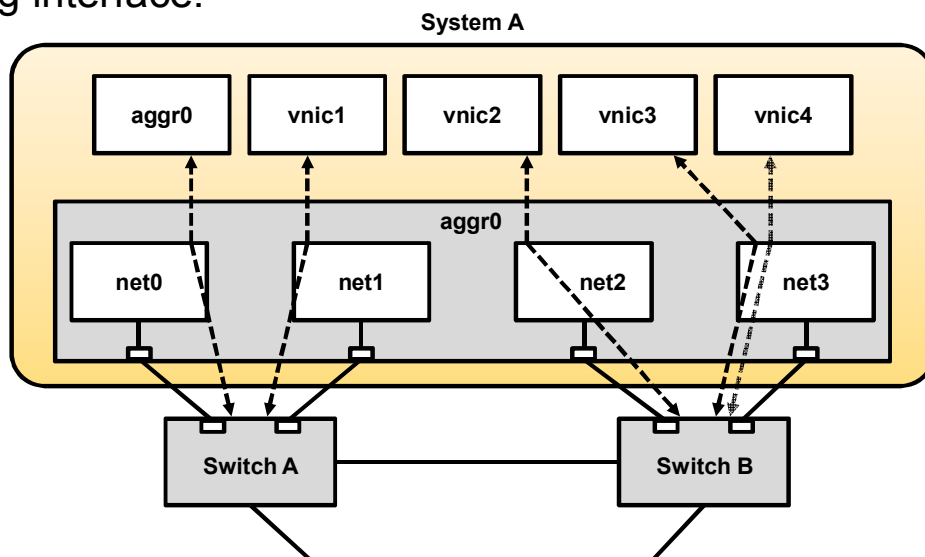
Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

A trunk aggregation is limited to work with only one switch. Thus, the switch becomes a single point of failure for the system's aggregation. Past solutions to enable aggregations to span multiple switches had their own disadvantages:

- Switch-based solutions were vendor-specific and not standardized. If multiple switches from different vendors were used, a particular vendor's solution might not be applicable to the products of the other vendors.
- Combining link aggregations with IPMP was very complex, especially in the context of network virtualization that involved global and non-global zones. The complexity increased as you scaled the configurations.
- Even if a combination of link aggregation and IPMP was implemented, that configuration would not benefit from other advantages of working on the datalink layer alone, such as link protection, user-defined flows, and the ability to customize datalink properties such as bandwidth.

## DLMP at Work

In a DLMP configuration, VNICs created over an aggregation and the aggregation itself have a one-to-one mapping with the underlying interface.



ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In a trunk aggregation, the VNICs configured over an aggregation get a consolidated bandwidth of the underlying links. That is because each of the network port is associated with every configured datalink over the aggregation. However, in a DLMP configuration, VNICs configured over an aggregation can only provide high availability and unlike trunk aggregation not the benefit of consolidated bandwidth. This is because in DLMP, the VNICs created over an aggregation and the aggregation itself have a one-to-one mapping with the underlying interface.

The figure in the slide shows how a DLMP aggregation works. The figure shows System A with link aggregation `aggr0`. The aggregation consists of four underlying links, from `net0` to `net3`. In addition to `aggr0`, the primary interface, VNICs are also configured over the aggregation: `vnic1` through `vnic4`. The aggregation is connected to Switch A and Switch B.

If the number of VNICs exceeds the number of underlying links, then an individual port is associated with multiple datalinks. As an example, the figure shows that `vnic4` shares a port with `vnic3`.

Similarly, if an aggregation's port fails, then all the datalinks that use that port are distributed among the other ports. For example, if `net0` fails, then `aggr0` shares a port with one of the other datalinks. The distribution among the aggregation ports occurs transparently and independently of the external switches connected to the aggregation.

If a switch fails, the aggregation continues to provide connectivity to its datalinks by using the other switches.

## Comparison Between Trunk Aggregation and DLMP Aggregation

Feature	Trunk Aggregation	DLMP Aggregation
Link-based failure detection	Supported	Supported
LACP	Supported	Not Supported
Use of standby interfaces	Not Supported	Supported
Multiple switches capability	Not supported unless using vendor proprietary solution	Supported
Switch configuration	Required	Not Required
Policies for load balancing	Supported	Not Applicable
Load spreading across all the aggregation's ports	Supported	Limited
User-defined flows for resource management	Supported	Supported
Link protection	Supported	Supported
Back-to-back parallel configuration	Supported	Not Supported

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide displays a comparative summary between trunk aggregation and DLMP aggregation.

## Preconfiguration Requirements for Link Aggregation

Before you begin configuring and administering link aggregations, ensure that your aggregation setup conforms to the following requirements:

- No IP interface is configured over the datalinks that is to be configured into an aggregation.
- All the datalinks in the aggregation must run at the same speed and in full-duplex mode.
- For a trunk aggregation, ensure that the following tasks have been performed on the switch:
  - Ports are configured to be used as an aggregation
  - LACP is configured in either active or passive mode, if the switch supports LACP

**Note:** These prerequisites do not apply for DLMP aggregations.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Preconfiguration Requirements for Link Aggregation

- For DLMP aggregations, you must have at least one switch to connect the aggregation to the ports in other systems.
- Devices must support link state notification (generic LAN driver version (gldv) compliant devices) as defined in the IEEE 802.3ad Link Aggregation Standard for a port to attach to an aggregation or to detach from an aggregation.

**Note:** Devices that do not support link state notification can be aggregated only by using the `-f` option of the `dladm create-aggr` command. For such devices, the `link state` is always reported as `UP`.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered within a solid red rectangular bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** On SPARC-based systems, each datalink must have its own unique MAC address.

## Creating a Link Aggregation

Link aggregation can be created by using the `dladm create-aggr` command.

```
# dladm create-aggr [-t] [-R <root-dir >] [-m <mode >]
[-P <policy >] [-L <lacpmode >] [-T <time >]
[-u <address >] -l <link >[-l <link >...] <link >
```

Where:

- `mode` is an optional argument, and it can be either `trunk` or `DLMP`
- If unspecified, `mode = trunk`, means trunk aggregation

**Note:** The previous `-L` mode is now renamed to `-L lacpmode`, because `mode` now refers to the aggregate creation mode.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The following procedure helps create link aggregations:

1. Display datalink information by using the `dladm show-phys` command, to determine which datalinks to aggregate.
2. To remove the IP interface, use the `ipadm delete-ip interface` command, where `interface` specifies the IP interface over the link.
3. Create a link aggregation by using one of the following commands:
  - To create a trunk aggregation, issue the following command:  

```
# dladm create-aggr [-f] [-P policy] [-L lacpmode] [-T time] [-u address] -l link1 -l link2 [...] aggr
```
  - To create a DLMP aggregation, use the following command:  

```
# dladm create-aggr -m DLMP -l link1 -l link2 [...] aggr ,
```

 where `-l linkn` specifies the datalinks that you want to aggregate and `aggr` specifies the name of the aggregation.
4. Check the status of the aggregation that you just created by using the `dladm show-aggr` command.



## Commands to Administer Link Aggregations

After you have created a link aggregation, you can use the commands listed below to administer it.

Task	Command
Switch between link aggregation types or modify a trunk aggregation.	<code>dladm modify-aggr</code>
Add a link to an aggregation.	<code>dladm add-aggr</code>
Display link aggregation details.	<code>dladm show-aggr -x</code>
Remove a link from an aggregation.	<code>dladm remove-aggr</code>
Delete a link aggregation.	<code>dladm delete-aggr</code>

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Quiz

You can switch between a trunk aggregation and DLMP aggregation by using the `dladm modify-aggr` command.

- a. True
- b. False

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: a**

# Agenda

- Describing the network stack in Oracle Solaris 11
- Configuring a network interface
- Implementing network virtualization
- Configuring network high availability
- **Implementing resource management**
- Securing the network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

# Overview of Network Resource Management

- In Oracle Solaris 11, Quality of Service (QoS) is obtained more easily and dynamically by managing network resources.
- Network resource management is helpful for network provisioning, establishing service-level agreements, billing clients, and diagnosing security problems.
- Network resource management consists of setting datalink properties that pertain to network resources.
- Setting these datalink properties allows you to determine the amount of resource that should be allocated for networking processes.
  - For example, a link can be associated with a specific number of CPUs that are reserved exclusively for networking processes.

The Oracle logo, consisting of the word "ORACLE" in white, uppercase letters on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

In Oracle Solaris releases prior to Oracle Solaris 11, implementing Quality of Service (QoS) is a complicated process. The process consists of defining queuing disciplines, classes, and filter rules, and indicating the relationships among all of these components.

## Datalink Properties

You can allocate network resources by setting the following datalink properties:

- **Bandwidth:** Depending on the need and priority of an interface, bandwidth can be controlled.
- **pool:** The `pool` link property enables you to bind network processing to a pool of CPUs.
- **cpu:** In a system with multiple CPUs, you can dedicate a given number of CPUs for specific network processing.
- **rxrings, txrings:** If a NIC supports ring allocation, its receive and transmit rings can be assigned for dedicated use by datalinks.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** You can set resource properties when you create the link. Alternatively, you can set these datalink properties later as well (for example, after analyzing your resource usage over a period of time). The procedures for allocating resources apply to both the virtual network environment as well as the traditional physical network.

# Flows

- The datalink properties can be applied to all traffic that passes through a VNIC or other datalinks, such as physical NICs, link aggregations, and IP over IB (InfiniBand).
- However, using flows, these properties can also be applied at a much granular level, such as to a subset of traffic.
- A flow enables identifying a subset of traffic based on a range of Layer 3 or Layer 4 attributes contained within a network packet header, such as source or destination IP addresses, subnets, transport protocols, and ports.
- Network flows is a key element of service virtualization and can be based on only one of the attributes to achieve bandwidth and priority control.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For example, you can create a flow according to the port that is being used, such as port 21 for FTP, or according to IP addresses, such as packets from a specific source IP address. However, you cannot create a flow for packets from a specified IP address that are received on port number 21.

Likewise, you cannot create a flow for all traffic from IP address 192.168.1.10, and then create a flow for transport layer traffic on 192.168.1.10. Thus, you can configure multiple flows on a system, with each flow based on a different attribute.

**Note:** You can also use flows for accounting purposes, such as to track the history of certain types of traffic. Flows also support monitoring capabilities. By enabling history logging, the amount of traffic on a link can be monitored for capacity and other planning purposes.

## Commands for Network Resource Management

- The command for allocating network resources depends on whether you are directly working on datalinks or flows.
- Use:
  - `dladm` for datalinks
  - `flowadm` for flows
- Although the commands to set properties are different for datalinks and flows, the syntax is similar.

**Note:** Currently, however, only the bandwidth property can be associated with flows. Flows can control bandwidth on a per-application, per-port, per-protocol, and per-address basis.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on the right side of a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## dladm for Allocating Datalink Properties

- For allocating datalink properties, use the appropriate `dladm` subcommand depending on whether you are setting the property while creating the link or setting the property of an existing link.
- To simultaneously create a link and allocate resources to it or set the property of an existing link, use the following syntaxes, respectively:

```
# dladm create-vnic -l link -p property=value[,property=value] vnic
# dladm set-linkprop -p property=value[,property=value] link
```

- For example, to set the maximum bandwidth of the `vnic0` datalink, use the following command:

```
# dladm set-linkprop -p maxbw=100m vnic0
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For more details about the `dladm` command and the properties that this command manages, refer to the `dladm (1M)` man page.



## flowadm for Managing Flows

- Oracle Solaris 11 provides a new capability to administer network flows by using the `flowadm` and `flowstat` commands from within exclusive-IP non-global zones.
- To simultaneously create a flow and add resources to it or set the property of an existing flow, use the following syntaxes, respectively:

```
# flowadm add-flow -l link -a attribute=value\  
[,attribute=value] -p property=value[,property=value] flow
```

- For example, to set the maximum bandwidth for the network traffic to port 80 on `vnic0`, use the following command:

```
# flowadm add-flow -l vnic0 -a transport=tcp,local_port=80 \  
-p maxbw=8M http-1
```

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red horizontal bar.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

For more details about the `flowadm` command and the properties that this command manages, refer to the `flowadm (1M)` man page.

## Quiz

In Oracle Solaris 11.1, only the bandwidth property can be associated with flows.

- a. True
- b. False

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Answer: a**

## Managing Network Resources

- Network resource management implements quality of service to enhance performance.
- The essential tasks here involve the following:
  - Configuring the virtual speed or bandwidth of VNICs
  - Configuring a CPU pool for datalinks
  - Allocating CPUs to datalinks

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Configuring Virtual Speed

- Network bandwidth control involves implementing a limit on how much bandwidth can be consumed.
- The bandwidth limit on a datalink defines the virtual link speed of a VNIC.
  - For example, you could make a VNIC on a 10 GB network connection perform like a 1 GB connection by allowing it to take only 10% of the bandwidth, and then share the rest of the bandwidth among other VNICs.
- This bandwidth limit can be applied in two ways:
  - To a datalink, such as a VNIC
  - To a user-defined flow

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Network bandwidth control is generally done to split up large aggregated network pipes or to prevent a specific application or zone from taking all the network bandwidth, usually to the detriment of applications on the rest of the system.

## Configuring CPU Pools for Datalinks

- `pool` is a link property that enables you to bind network processing to a pool of CPUs and finally assign the pool to a zone.
- This procedure configures a CPU pool for a VNIC:
  1. Set the link's `pool` property to the pool of CPUs that you created for the zone. Perform one of the following steps:
    - If the VNIC has not yet been created, create one by using the `dladm create-vnic -l link -p pool=pool vnic` command, where `pool` refers to the name of the pool that was created for the zone.
    - If the VNIC exists, use the `dladm setlinkprop -p pool=pool vnic` command.
  2. Set a zone to use the VNIC by using the `zonecfg>zoneid:net> set physical=vnic` command.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font on a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The `pool` property can be set for a datalink either when the link is created or later when the link requires further configuration. However, before you begin configuring CPU pools for datalinks, ensure that you have completed the following prerequisite tasks:

- Created a processor set (pset) with its assigned number of CPUs
- Created a pool
- Associated the pool with the pset

## Allocating CPUs to Datalinks

The following procedure explains how to assign specific CPUs to process traffic traversing a datalink by configuring the `cpu` property.

1. Check CPU assignments for the interface by using the `dladm show-linkprop -p cpus link` command.
2. List the interrupts and the CPUs with which the interrupts are associated by using the `echo ::interrupts | mdb -k` command.
  - The output of the above command lists parameters for each link in the system, including the CPU number.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** By default, no CPUs are assigned to any specific interface. Thus, the parameter `VALUE` in the `dladm show-linkprop -p cpus link` command output does not contain any entry.

## Allocating CPUs to Datalinks

3. Assign CPUs to the link by using the `dladm set-linkprop -p cpus=cpu1,cpu2,... link` command, where *cpu1* is the CPU number to be assigned to the link.
  - The CPUs can include those with which the link's interrupts are associated.
  - You can dedicate multiple CPUs to the link.
4. Check the link interrupt to verify the new CPU assignments by using the `echo ::interrupts | mdb -k` command.
  - Observe the CPU number now.
5. Optionally, display the CPUs that are associated with the link by using the `dladm show-linkprop -p cpus link` command.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** The two properties, `cpu` and `pool`, are mutually exclusive. You cannot set both properties for a given datalink. Another area of difference is that the pool property is dynamic in nature. Zone pools can be configured with a range of CPUs, and the kernel determines which CPUs are assigned to the pool's CPU set. Changes to the pool are automatically implemented for the datalink, which simplifies pool administration for that link.

By contrast, assigning specific CPUs to the link by using the `cpu` property requires you to specify the CPU to be assigned. You have to set the `cpu` property every time you want to change the CPU components of the `link`.

# Agenda

- Describing the network stack in Oracle Solaris 11
- Configuring a network interface
- Implementing network virtualization
- Configuring network high availability
- Implementing resource management
- Securing the network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.



## Need for Network Security

- To secure the network, you must enforce security measures in both the physical and virtual networks.
- The physical network is ably serviced through firewalls.
- However, a virtualized network presents a different scenario and a firewall is not quite sufficient there.
  - In virtualized network setups, it is common for the host administrator to grant exclusive access of a physical link or a VNIC to a zone.
  - Exclusive access means traffic isolation and improved performance for the zone.
  - However, the downside is that the zones can generate any type of packet, including harmful ones, and release it into the network.

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

## Overview of Link Protection

- Introduced in Oracle Solaris 11, link protection is a security mechanism that prevents potentially malicious or misbehaving zones from sending harmful packets to the network.
- Link protection primarily provides protection against the basic spoofing threats.
- Spoofing refers to the act of tricking computer systems or computer users and is done usually through impersonation.

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is positioned on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

**Note:** Link protection does not replace the deployment of a firewall, particularly for configurations with complex filtering requirements. Unlike a traditional firewall, link protection does not support inbound filtering or customizable filtering rules. For such requirements, you must use a firewall instead, such as Oracle Solaris IP Filter.

## Link Protection Types

- Spoofing occurs in some of the following ways:
  - Email spoofing is one of the most common threats, which involves sending messages from a fake email address.
  - Various other similar threats, such as IP, DHCP, MAC, and L2 frame spoofing can compromise the network.
- The link protection mechanism in Oracle Solaris 11 provides the following protection types against the above-mentioned threats:
  - `ip-nospoof`
  - `dhcp-nospoof`
  - `mac-nospoof`
  - `restricted`

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, centered within a red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The link protection mechanism provides the following protection types:

- **`ip-nospoof`:** Enables protection against IP spoofing, which involves faking a computer's IP address. IP spoofing is commonly used in denial-of-service attacks that overwhelm a server.
- **`dhcp-nospoof`:** Enables protection against spoofing of the Dynamic Host Control Protocol (DHCP) client. DHCP spoofing is a man-in-the-middle attack. The attacker responds to a DHCP request message by providing the client with an IP address and setting its own IP address as the gateway address of the affected client machine.
- **`mac-nospoof`:** Enables protection against spoofing the system's MAC address. MAC spoofing involves changing a factory-assigned MAC address of a network interface on a networked device in an attempt to gain access to your machine.
- **`Restricted`:** Provides protection against L2 frame spoofing, such as Bridge Protocol Data Unit (BPDU) attacks. This protection type is designed to prevent the link from generating potentially harmful L2 control frames.

# Configuring and Administering Link Protection

Task	Command
View the available link protection types.	<code># dladm show-linkprop -p protection</code>
Enable link protection by specifying one or more protection types.	<code># dladm set-linkprop -p protection=value[,value,...] link</code>
Verify that the link protections are enabled.	<code># dladm show-linkprop -p protection link</code>
Disable link protection by resetting the protection property to its default.	<code># dladm reset-linkprop -p protection link</code>
Add IP addresses to the list of default values for the <code>allowed-ips</code> link property.	<code># dladm set-linkprop -p allowed-ips=IP-addr[,IP-addr,...] link</code>

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide contains a summary of the `dladm` commands that are used for configuring and administering link protection.

# Configuring and Administering Link Protection

Task	Command
Specify an ASCII phrase for the allowed-dhcp-cids link property.	<pre># dladm set-linkprop -p allowed-dhcp-cids=<i>CID-or-DUID[,CID-or-DUID,...]</i> link</pre>
View the link protection property values.	<pre># dladm show-linkprop -p protection,allowed- ips,allowed-dhcp-cids link</pre>
View the link protection statistics.	<pre># dlstat -A</pre>

The Oracle logo, consisting of the word "ORACLE" in a white, sans-serif font, is centered on a solid red rectangular background.

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The table in the slide contains a summary of the dladm commands that are used for configuring link protection.

## Summary

In this lesson, you should have learned how to:

- Describe the network stack in Oracle Solaris 11
- Configure a network interface
- Implement network virtualization
- Configure network high availability
- Implement resource management
- Secure the network

ORACLE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.