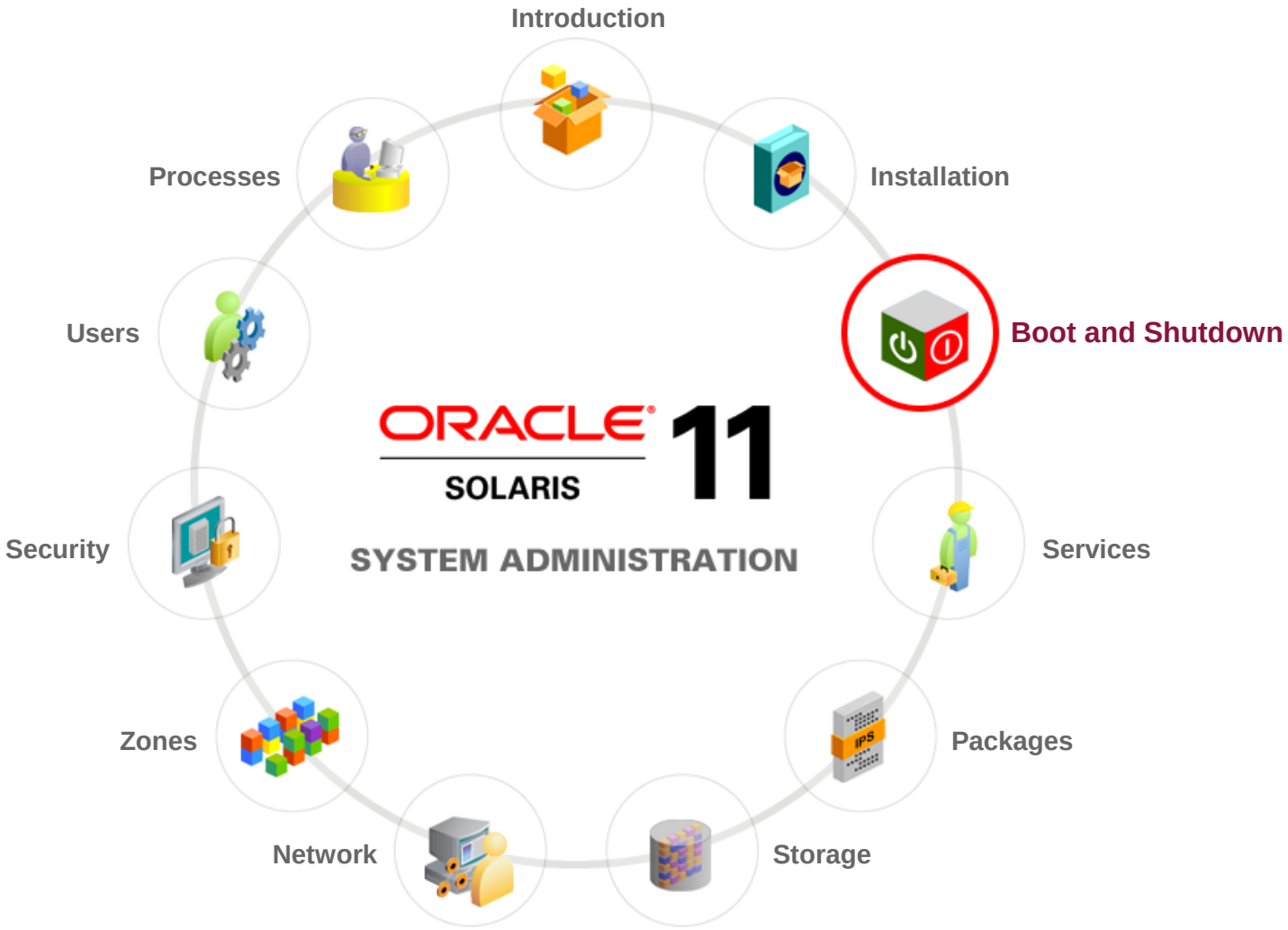
3

Managing Boot and Shutdown of a System

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Workflow Orientation



Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Objectives

After completing this lesson, you should be able to:

- Analyze the boot design and boot process
- Boot a SPARC-based system
- Boot an x86-based system
- Shut down a system

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Agenda

- **Analyzing the Boot Design and Boot Process**
- Booting a SPARC-Based System
- Booting an x86-Based System
- Shutting Down a System

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Reasons to Shut Down and Boot a System

- Turning off system power due to anticipated power outage
- Changing kernel parameters in the `/etc/system` file
- Performing file system maintenance, such as backing up or restoring system data
- Repairing a system configuration file, such as `/etc/system`
- Adding or removing hardware from the system
- Booting a system for recovery purposes due to a lost `root` password, or to fix a file system or a similar problem
- **(x86 only)** Recovering from a problem with the GRUB configuration
- Recovering from a hung system by forcing a crash dump
- Booting the system to track down a system problem

ORACLE

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Oracle Solaris Boot Architecture: Overview

- The SPARC and x86 boot design architectures are similar.
- The differences are in how the boot device and arguments are selected at boot time.
- SPARC uses OpenBoot PROM (OBP) and its commands.
- x86 uses the GRUB menu.
- By default, the SPARC and x86 platforms have one primary boot archive.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

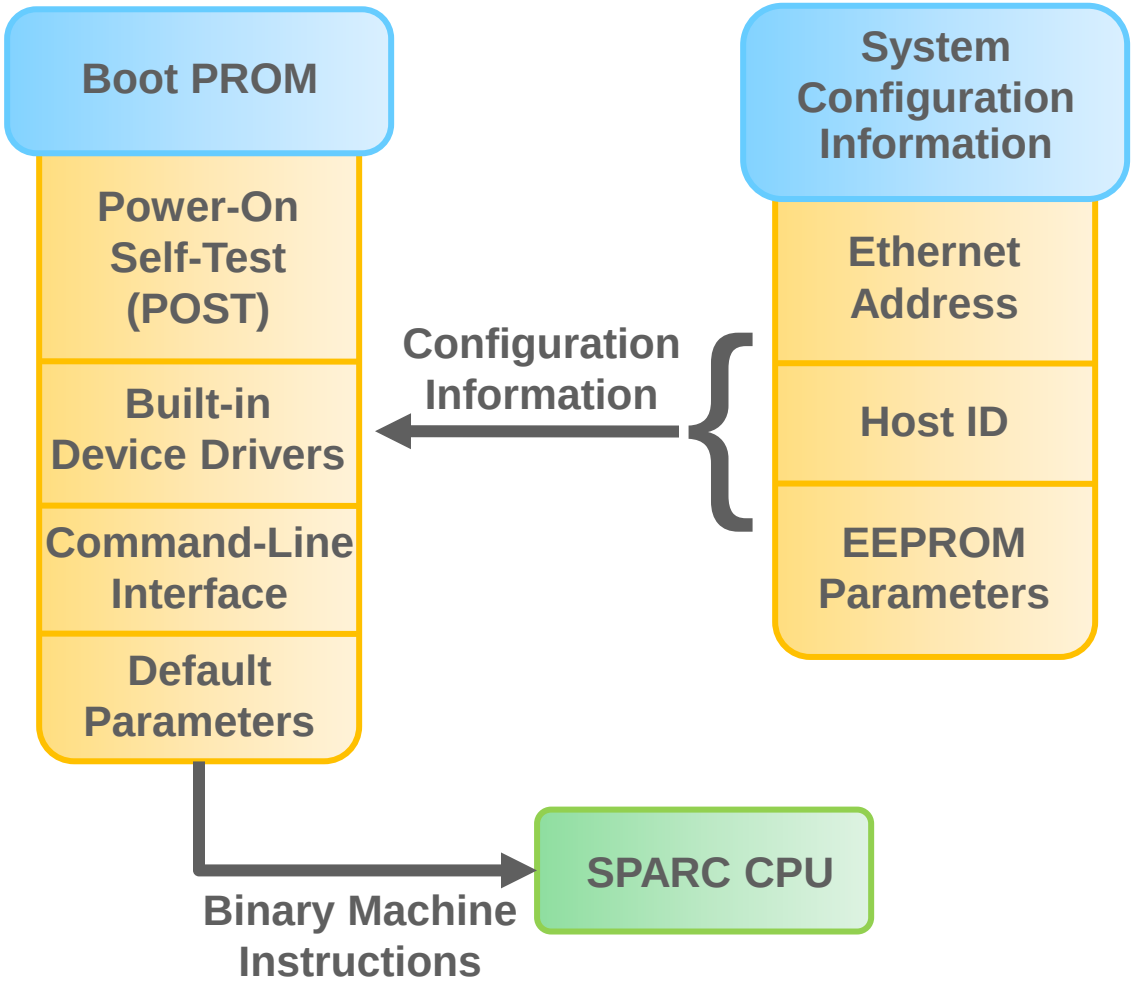
Boot PROM for SPARC Systems

The boot PROM firmware:

- Provides basic hardware testing and initialization before loading the operating system
- Enables booting from a wide range of devices
- Provides a user interface
- Has access to a standard set of device drivers

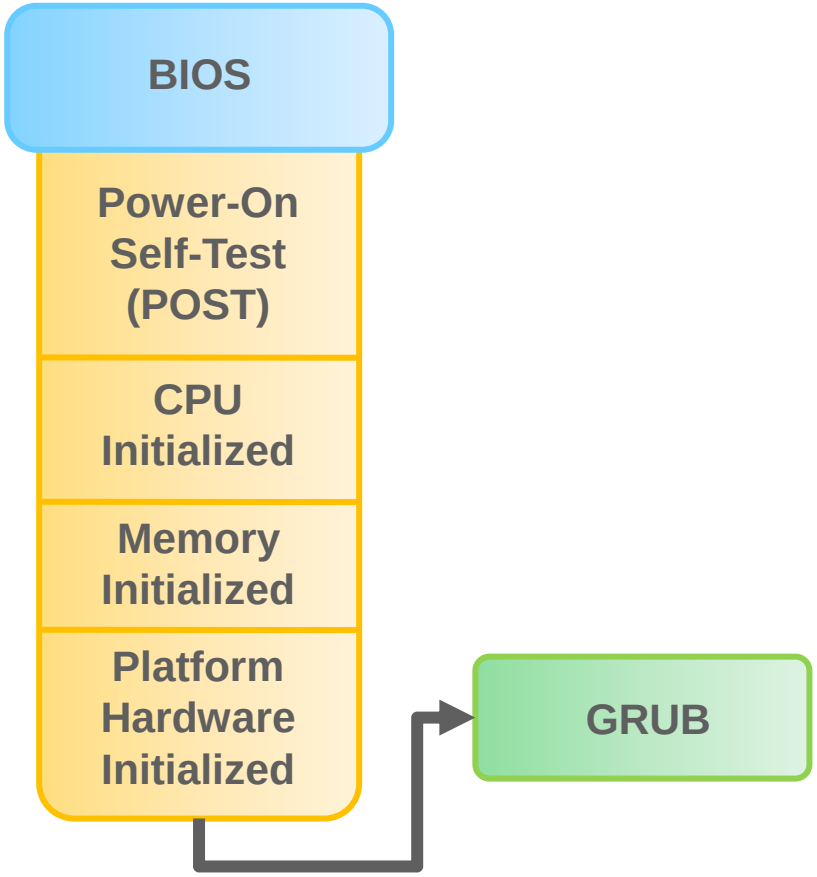
Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Bootstrapping Process for SPARC Systems *(Boot PROM Initialization)*



Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Bootstrapping Process for x86 Systems *(BIOS and GRUB Initialization)*



Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

GRUB 2

- GRUB 2 is managed through the `grub.cfg` file.
- The `grub.cfg` file should never be edited.
- GRUB 2 supports the Unified Extensible Firmware Interface (UEFI) and the GUID Partition Table (GPT) partitioning scheme.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Boot Process

Boot Loader Phase	The root file system archive is loaded.
Booter Phase	The boot archive is read and executed.
Ramdisk Phase	The kernel image is extracted and executed.
Kernel Phase	The OS is initialized and the root file system is mounted.
`init` **Phase**	The `init` daemon starts the `svc.startd` daemon.
`svc.startd` **Phase**	The `svc.startd` daemon starts the system processes.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Booting from Firmware-Inaccessible Storage Devices

Starting from Oracle Solaris 11.3, a new boot process is available for servers that can boot by using firmware-inaccessible storage devices, such as an iSCSI device accessed using IP over InfiniBand (IPoIB).

- Boot pools are created on systems that can boot from firmware-inaccessible storage devices. The boot archives in the boot pool can then be accessed.
- The boot pool includes boot loader data files and recovery data.
- On a system that boots from a hardware-inaccessible storage device, you can manage the boot pool with the `bootadm boot-pool` command.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

SMF and Booting

In Oracle Solaris 11, SMF provides an infrastructure that augments the traditional UNIX `startup` scripts, `init` states, and configuration files.

- The boot process has become faster with multiple SMF services started simultaneously.
- Due to SMF, the boot process creates fewer messages now.
- The SMF services do not display a message by default when they are started during the boot process.
- All information that was provided by the boot messages can now be found in a log file for each SMF service under `/var/svc/log`.
- You can use the `svcs` command to diagnose boot problems.
- To generate a message when each service is started during the boot process, you use the `-v` option with the `boot` command.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

How Oracle Solaris Boot Archives Are Managed

- Boot archive updates and verification are handled automatically by the `bootadm` command.
- During an installation or upgrade, an initial boot archive is created.
- During normal shutdown, the boot archive contents are compared with the root file system.
- If inconsistencies are found, the boot archive is rebuilt to make sure that the boot archive and the root file system are synchronized.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Fast Reboot

- Is supported on both SPARC and x86 platforms
- Bypasses firmware and boot loader processes to provide an extremely fast reboot
- Is controlled by the SMF
- Is implemented through a boot configuration service (`svc:/system/boot-config`) based on the setting of the `fastreboot_default` property:
  - Set to `true` on x86 systems by default
  - Set to `false` on SPARC systems by default

- Use reboot –p to back to normal

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

SMF Milestones

`init` State	SMF Milestone FMRI
S	milestone/single-user:default
2	milestone/multi-user:default
3	milestone/multi-user-server:default

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Quiz

Which SMF service helps in implementing the Fast Reboot feature?

a. `svc:/system/boot-update:default`
b. `svc:/system/boot-archive:default`
c. `svc:/system/boot-archive-update:default`
d. `svc:/system/boot-config:default`
e. `svc:/system/boot-config-update:default`

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Quiz

Which SMF service helps in implementing the Fast Reboot feature?

a. `svc:/system/boot-update:default`
b. `svc:/system/boot-archive:default`
c. `svc:/system/boot-archive-update:default`
d. `svc:/system/boot-config:default`
e. `svc:/system/boot-config-update:default`

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Quiz

In which phase of the boot process is the OS initialized and a minimal root file system mounted on the RAM disk that was constructed from the boot archive?

a. Kernel phase
b. Boot loader phase
c. `svc.startd` phase
d. Booter phase
e. Ramdisk phase
f. `init` phase

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Quiz

In which phase of the boot process is the OS initialized and a minimal root file system mounted on the RAM disk that was constructed from the boot archive?

a. Kernel phase
b. Boot loader phase
c. `svc.startd` phase
d. Booter phase
e. Ramdisk phase
f. `init` phase

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Agenda

- Analyzing the Boot Design and Boot Process
- **Booting a SPARC-Based System**
- Booting an x86-Based System
- Shutting Down a System

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Booting a SPARC-Based System

- Booting a SPARC system to the multiuser-server milestone (`init` state `3`)
- Booting a SPARC system to the single-user milestone (`init` state `S`)

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Booting a SPARC System to the Multiuser-Server Milestone (`init` State `3`)

1. Boot the system to the multiuser-server milestone by using the `boot` command at the `ok` prompt.

```
ok boot
Resetting ...
<output truncated>
```

2. When prompted, log in to the system.
3. Verify that the system has booted to the multiuser-server milestone.

```
# who -r
.       run-level 3 Nov 11 11:32          3        0 S
#
```

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Booting a SPARC System to the Single-User Milestone (`init` State `S`)

1. Boot the system to the single-user milestone by using the following command at the `ok` prompt:

```
ok boot -m milestone=single-user
```

2. When prompted, enter the root password.

```
SINGLE USER MODE
Root password for system maintenance (control-d to bypass): xxxxxx
```

3. Verify that the system is at the single-user milestone.

```
# who -r
.       run-level S Nov 11 10:15          S        0 S
#
```

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Initiating a Fast Reboot of a SPARC-Based System

To initiate a fast reboot of a SPARC system, run `reboot -f`.

```
# reboot -f
```

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Using the Basic Boot PROM Commands

Boot PROM Command	Description
`banner`	Displays a system's PROM revision number
`setenv`	Sets the specified NVRAM parameter to a value For example, `ok setenv auto-boot? false` sets the PROM `auto-boot?` value to `false`.
`reset-all`	Clears system registers and resets the entire system
`sifting probe`	Displays the probe commands that are available on your system
`probe-device`	Identifies the devices on the system
`devalias`	Identifies the device aliases and the associated paths of devices that *might* be connected to the system

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Using the Basic Boot PROM Commands

Boot PROM Command	Description
`printenv`	Displays all current and default parameters
`eeprom`	Helps in displaying and modifying the value of parameters in EEPROM

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Agenda

- Analyzing the Boot Design and Boot Process
- Booting a SPARC-Based System
- **Booting an x86-Based System**
- Shutting Down a System

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Booting an x86 System

- Booting an x86 system to the multiuser-server milestone
- Booting an x86 system to the single-user milestone

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Booting an x86 System to the Multiuser-Server Milestone

1. Reboot the system by using the `reboot` command.

```
# reboot -p
```

2. When the GRUB menu appears, press Enter to boot the default OS instance to the multiuser-server milestone.
3. When the login prompt appears, log in to the system as `root`.
4. Verify that the system has booted to the multiuser-server milestone (init state `3`).

```
# who -r
.           run-level 3 Nov 11 11:32          3          0 S
#
```

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Booting an x86 System to the Single-User Milestone (`init` State `S`)

1. Reboot the system by using the `reboot -p` command.
2. When the GRUB menu appears, enter `e` to edit the GRUB menu.
3. Use the arrow keys to choose the `kernel $` line.
4. Enter `e` again to edit the boot entry.
5. To boot the system in single-user mode, enter `-s` at the end of the boot entry line. Then press Return to go back to the previous screen.
6. To continue to boot the system in single-user mode, enter `b`.
7. When prompted, enter the `root` password.
8. Verify that the system is at the single-user milestone.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Initiating Fast Reboot on an x86-Based System

Because Fast Reboot is the default boot mode on an x86 system, you can use either the `reboot` command with the `-f` option or the `init 6` command to initiate the reboot.

```
# reboot -f
```

```
# init 6
```

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Securing the GRUB Menu

You can control access to the GRUB menu by setting a password lock. The options to the `bootadm set-menu-password` command that manage the password lock are:

Option	Description
`-s`	Sets the password needed to view, edit, or boot any entry in the GRUB menu
`-r`	Removes the password needed to access the GRUB menu
`-l`	Lists whether a password lock is in place and which users have access to each menu entry

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Using the `bootadm` Command

`bootadm` Subcommand	**Description**
`list-menu`	Lists the contents of the `grub.cfg` file
`generate-menu`	Generates the `grub.cfg` file
`set-menu`	Sets a particular GRUB menu entry as the default, and other menu options and boot loader options; helps in maintaining the GRUB menu
`add-entry`	Adds a boot entry to the GRUB menu
`change-entry`	Changes the attributes of a specified boot entry in the GRUB menu

ORACLE

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Using the `bootadm` Command

`bootadm` Subcommand	**Description**
`install-bootloader`	Installs the system boot loader **Note:** This subcommand applies to both x86 and SPARC platforms.
`remove-entry`	Removes a boot entry from the GRUB menu
`set-menu-password`	Sets a password to prevent the GRUB menu from being seen
`show-entry`	Shows a boot entry from the GRUB menu. This subcommand is equivalent to `list-menu`.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Example

1. Set password to the grub

bootadm set-menu-password –s

enter abcd1234

2. Remove password-protected GRUB menu

bootadm set-menu-password -P rpool -r

3. Add user into authenticated users

useradd user1

useradd user2

bootadm set-menu add-user=user1

abcd1234

bootadm set-menu add-user=user1

abcd1234

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Example

4. To give user authorization to boot and edit specific entries in GRUB menu

bootadm change-entry –i 1 add-auth=user2

5. List current menu

bootadm list-menu

6. Add new entry to boot menu

bootadm add-entry -i new-entry-number bootfs='pool-name/ROOT/be-name’

bootadm add-entry -i 2 Oracle Solaris 11 test

7. Change existing entry

bootadm change-entry -i 2 bootfs='rpool/ROOT/test’

8. Remove a boot entry

bootadm remove-entry -i 2

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Example

9. Change entry to boot with different kernel arguments

bootadm change-entry -i 2 kargs=“-v -s”

10. Disable e1000g network driver and load kernel debugger

bootadm change-entry -i 2 kargs=“-B disable-e1000g=true -k”

11. Remove added kernel arguments

bootadm change-entry -i 2 kargs=

12. List and verify

bootadm list-menu -i 2

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Agenda

- Analyzing the Boot Design and Boot Process
- Booting a SPARC-Based System
- Booting an x86-Based System
- **Shutting Down a System**

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Shutting Down a System

- Shutting down a server:
  - The `shutdown` command is used.
  - A clean shutdown is performed.
  - Superuser privileges are required.

- Shutting down a stand-alone system:
  - The `init` command is used.
  - A clean shutdown is performed.
  - Superuser privileges are required.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Run Levels

A system's run level (init state) defines what services and resources are available to users. A system can be in only one run level at a time.

Run Level	Init State	Type	Purpose
0	Power-down state	Power-down	To shut down the operating system so that it is safe to turn off power to the system
s or S	Single-user state	Single-user	To run as a single user with some file systems mounted and accessible
1	Administrative state	Single-user	To access all available file systems. User logins are disabled.
2	Multiuser state	Multiuser	For normal operations. Multiple users can access the system and all file systems. All daemons would be running except the NFS and SMB server daemons.

ORACLE

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Run Levels

Run Level	Init State	Type	Purpose
3	Multiuser level with NFS resources shared	Multiuser	For normal operations with NFS and SMB resources shared. This is the default run level.
4	Alternative multiuser state	Multiuser	Not configured by default, but available for customer use
5	Power-down state	Power-down	To shut down the operating system so that it is safe to turn off power to the system; if possible, automatically turns off power on systems that support this feature
6	Reboot state	Reboot	To shut down the system to run level 0, and then reboot to multiuser level with NFS and SMB resources shared (or whatever level is the default in the inittab file)

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Determining Who Is Logged In to a System

To determine who is logged in to a system, run the `who` command.

```
$ who
holly  console  Nov 11 07:30
kryten  pts/0   Nov 11 07:35 (starlite)
lister  pts/1   Nov 11 07:40 (bluemidget)
```

Best Practice: Always send an additional email notification to logged in users, indicating that the server is going to be down for a specified amount of time.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Shutting Down a Server

1. Determine who is logged in to the system.
2. Shut down the system by using the `shutdown` command with the `-i` *`init-level`*, `-g` *`grace-period`*, and `-y` options.
3. When prompted, enter the superuser password.
4. Verify that the system is at the init state that you specified.
5. When you have completed your administration tasks, press Ctrl + D to return to the default system init state.
6. Verify that the system is at the init state that you specified in the `shutdown` command.

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Shutting Down a Server

Specified init State	SPARC System Prompt	x86 System Prompt
`0` (exit the OS)	`ok >`	`Press any key to reboot`
`s` or `S` (single-user milestone)	`#`	`#`
`3` (multiuser-server milestone)	*`hostname`* `console login:`	*`hostname`* `console login:`
`5` (`shutdown`→`poweroff`)	`Power off`	`Power off`

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Shutting Down a Stand-Alone System

To bring a stand-alone system to init state `0`, run `init 0`.

```
# init 0
```

To bring a stand-alone system to init state `S`, run `init S`.

```
# init s
```

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Summary

In this lesson, you should have learned how to:

- Analyze the boot design and boot process
- Boot a SPARC-based system
- Boot an x86-based system
- Shut down a system

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.

Practice 3: Overview

- 3-1: Booting and Shutting Down a SPARC Host
- 3-2: Booting and Shutting Down an x86/64 Host

Copyright © 2017, Oracle and/or its affiliates. All rights reserved.