Oracle Linux System Administration

Activity Guide

D74508GC20
Edition 2.0
February 2014
D85362

ORACLE®

Author

Craig McBride

Technical Contributors and Reviewers

Avi Miller, Elena Zannoni, Lenz Grimmer, Sergio Leunissen, Waseem Daher, Wim Coekaerts, Al Flournoy, Harald Van Breederode, Joel Goodman, Manish Kapur,Soeren Binner,Jeremy Smyth, Yasar Akthar, Javier Saiz, Ozgur Yuksel, Antoinette O'Sullivan, Frank Allan, Gavin Bowe, Gino Kawalski, Jeff Suchomel, Rob Swank, Ron Hardin, Michele Dady, Matt Taylor

This book was published using: Oracle Tutor

Table of Contents

Practices for Lesson 1: Course Introduction

Chapter 1

Practices for Lesson 1: Course 1

Practices Overview

In these practices, you become familiar with the Oracle VM Server environment installed on your classroom PC. You log on to your classroom PC and connect to the virtual machines used for the hands-on practices.

Assumptions

- Your instructor has assigned a student PC to you.
- The student PC is running OVM 2.2.1.
- The GNOME desktop is installed on **dom0**.
- Three guests (virtual machines) are created: **host01**, **host02**, and **host03**.
- Guest VMs **host01** and **host02** are running Oracle Linux 6.5.
- You install Oracle Linux 6.5 on **host03** in Practice 3.

Practice 1-1: Exploring the dom0 Environment

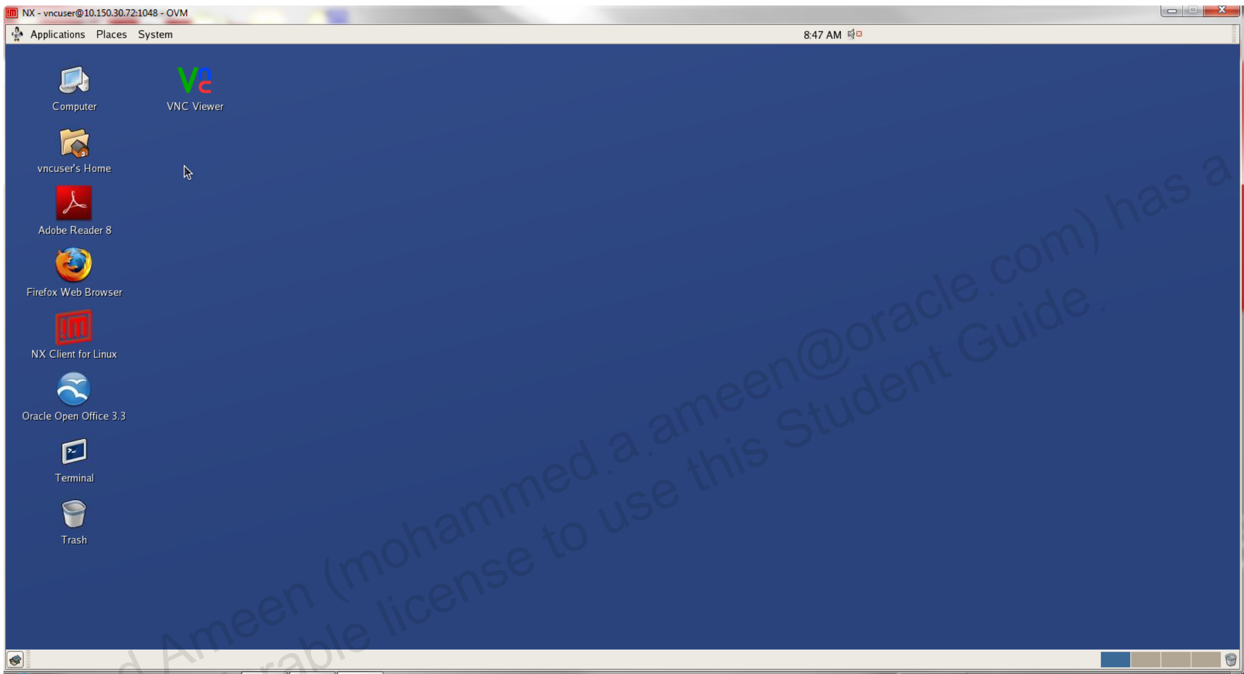
Overview

In this practice, you explore **dom0** configuration and directory structure.

Tasks

1. Open a terminal window.

- Begin this task from the **dom0** GNOME virtual desktop window as shown below.

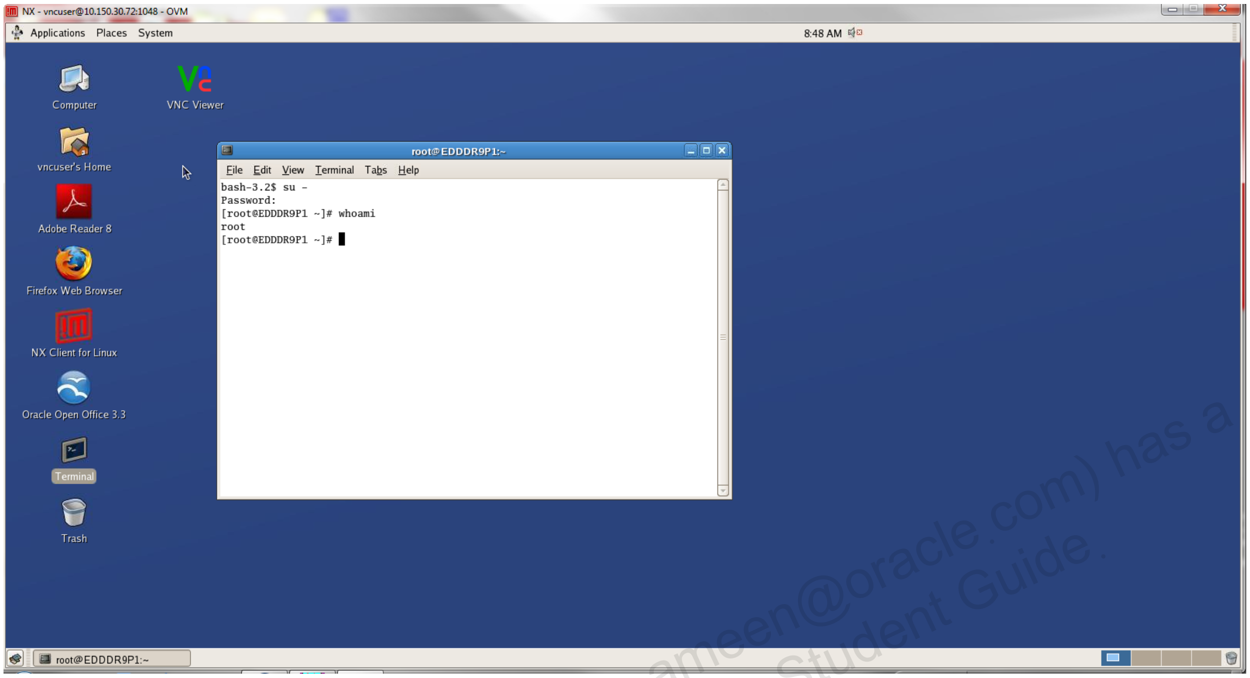


Double-click the **Terminal** icon on the GNOME desktop.

- A terminal window opens.

2. Become the `root` user.

- Enter the commands from an open terminal window as shown below.



Become `root` by using the `su -` command. The `root` password is `oracle`. Use the `whoami` command to print the user identity to confirm that you are `root`:

```
$ su -
Password: oracle
# whoami
root
```

3. Determine the operating system running on **dom0**.

Use the `uname -a` command to display the operating system version. The output shown is a sample and may not represent the host name and kernel version running on your system.

```
# uname -a
Linux EDDDR9P1 2.6.18-128.2.1.4.25.el5xen #1 SMP Tue Mar 23
12:43:27 EDT 2010 i686 i686 i383 GNU/Linux
```

- In this example, the operating system is `Linux`.
- The Linux kernel is `2.6.18-128.2.1.4.25.el5xen`.
- The host name is `EDDDR9P1` (your host name is different).

4. Determine the network configuration of **dom0**.

   Use the `ifconfig -a` command to display the network configuration. The output shown is a sample and may not represent the network configuration of your system. Only partial output is shown.

```
# ifconfig -a
...
eth0      Link encap:Ethernet
          inet addr:10.150.30.72
...
lo        Link encap:Local Loopback
          inet addr:127.0.0.1
...
vif...    Link encap:Ethernet
...
virbr0    Link encap:Ethernet
          inet addr:192.0.2.1
...
virbr1    Link encap:Ethernet
          inet addr:192.168.1.1
...
xenbr0    Link encap:Ethernet
          inet addr:10.150.30.72
...
```

- In this example, the network interface for **dom0** is `eth0` and is assigned an IP address of `10.150.30.72`. Connection to the outside world is provided from this interface.
- The `lo` interface is a software loopback interface that identifies `localhost`. It is always assigned an IP address of `127.0.0.1`.
- The `virbr0` interface is a `xen` bridge interface used by the VM guests. It is assigned an IP address of `192.0.2.1`.
- The `virbr1` interface is a second `xen` bridge interface used by the VM guests. It is assigned an IP address of `192.168.1.1`.
- As you will see later, each of the guest virtual machines is on the `192.0.2` subnet and the `192.168.1` subnet. Communication to the classroom network is through `xenbr0`.
- You also notice `vif<#>.0` and `vif<#>.1` entries. These are the virtual interfaces that are tied to the VM/domU IDs. You can get the VM/domU IDs from the `xm list` command, which you run later in this practice.

5. Explore the `/OVS` directory structure on **dom0**.
   - The output shown is a sample and may not represent your system. Only partial output is shown.

   a. Explore the top level of the `/OVS` directory:

```
# ls -l /OVS
lrwxrwxrwx /OVS -> /var/ovs/mount/...
# cd /OVS
# ls -l
drwxrwxrwx iso_pool
drwxrwxrwx lost+found
drwxrwxrwx publish_pool
drwxrwxrwx running_pool
drwxrwxrwx seed_pool
drwxrwxrwx sharedDisk
```

   - Note that `/OVS` is a symbolic link to the `/var/ovs/mount/...` directory.
   - There are six directories in the `/OVS` directory.

   b. Explore the `/OVS/running_pool` directory:

```
# cd /OVS/running_pool
# ls -l
drwxr-xr-x host01
drwxr-xr-x host02
drwxr-xr-x host03
```

   - The files needed to create the VMs are in separate directories in the `/OVS/running_pool` directory.
   - This example shows that three VMs currently exist: **host01**, **host02**, and **host03**.

   c. Explore the **host03** VM directory.

```
# cd /OVS/running_pool/host03
# ls -l
-rw-r--r-- system.img
-rw-r--r-- u01.img
-rw-r--r-- u02.img
-rw-r--r-- vm.cfg
```

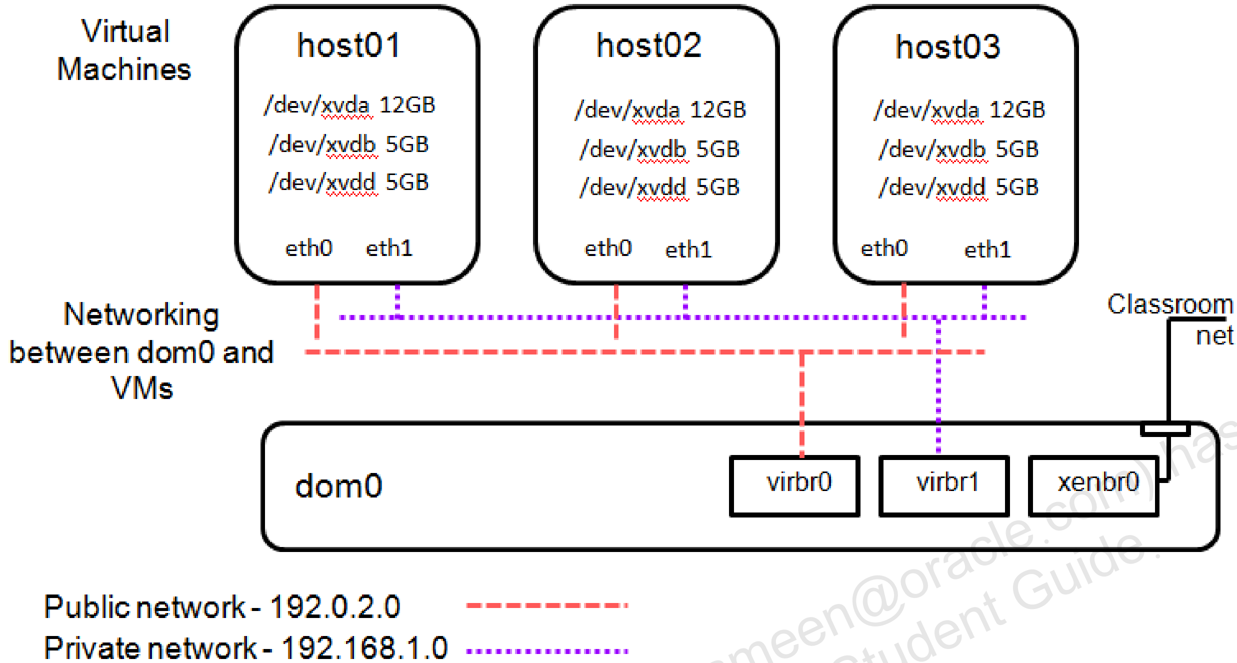
   - The `system.img` file is the operating system virtual disk.
   - The `u01.img` and `u02.img` files are additional virtual disks used by the storage administration practices.
   - The `vm.cfg` file is the configuration file for the virtual machine. This file is read when the virtual machine is created.

d. View the `vm.cfg` file.

```
# cat vm.cfg
name = 'host03'
builder = 'hvm'
memory = 2048
boot = 'cd'
disk = [ 'file:/OVS/running_pool/host03/system.img,hda,w',
         'file:/OVS/running_pool/host03/u01.img,hdb,w',
         'file:/OVS/running_pool/host03/u02.img,hdd,w',
         'file:/OVS/seed_pool/OracleLinux-R6-U5-Server-x86_64-
dvd.iso,hdc:cdrom,r']
vif = [ 'mac=00:16:3e:00:01:03, bridge=virbr0',
        'mac=00:16:3e:00:02:03, bridge=virbr1']
device_model = '/usr/lib/xen/bin/qemu-dm'
kernel = '/usr/lib/xen/boot/hvmloader'
vnc = 1
vncunused=1
vcpus = 1
timer_mode = 0
apic = 1
acpi = 1
pae = 1
serial = 'pty'
on_reboot = 'restart'
on_crash = 'restart'
usb = 1
usbdevice = 'tablet'
```

- Notice there are three virtual disks represented by the three `.img` files.
- Notice the Oracle Linux 6.5 `dvd.iso` is mounted on a virtual CDROM device.
- Notice there are two virtual network interfaces. The interface on the `virbr0` bridge is `eth0`, and the interface on the `virbr1` bridge is `eth1`.

The following displays the configuration. All three VMs (**host01**, **host02**, and **host03**) have two network interfaces and three disks.



Practice 1-2: Starting, Stopping, and Listing Guests

Overview

In this practice, you use `xm` commands to list, create, and shut down virtual machines.

Assumptions

- You are logged on to **dom0**.
- You have a terminal window open.
- You are the `root` user.

Tasks

1. List all currently active guests, as well as **dom0** itself.

   Use the `xm list` command. The output shown is a sample and will not represent your system exactly.

```
# xm list
Name                   ID    Mem  VCPUs      State    Time(s)
Domain-0                0   1024      2     r-----      281.1
host01                  3   2048      1     -b----      157.6
host02                  1   2048      1     -b----      159.0
host03                  2   2048      1     -b----       13.2
```

   - You should have three guests (**host01**, **host02**, and **host03**) listed in the preceding output.

2. Shut down the specified VMs.

   Use the `xm shutdown -w` *<VM name>* command to shut down the VMs. The `-w` option tells the system to wait until all services in the domain shut down cleanly.

```
# xm shutdown -w host01
Domain host01 terminated
All domains terminated
# xm shutdown -w host02
Domain host02 terminated
All domains terminated
# xm shutdown -w host03
Domain host03 terminated
All domains terminated
# xm list
Name                   ID    Mem  VCPUs      State    Time(s)
Domain-0                0   1024      2     r-----      289.6
```

3. Start the VMs.

   Use the `xm create <config_file>` command to start the **host01** and **host02** VMs. The *<config_file>* is named `vm.cfg` and is located in the `/OVS/running_pool/<VM_name>` directory.

```
# cd /OVS/running_pool/host01
# xm create vm.cfg
Using config file "./vm.cfg".
Started domain host01 (id=4)
# xm list
Name                     ID     Mem  VCPUs       State       Time(s)
Domain-0                  0    1024      2      r-----        304.5
host01                    4    2048      1      -b----         18.7
# cd /OVS/running_pool/host02
# xm create vm.cfg
Using config file "./vm.cfg".
Started domain host02 (id=5)
# xm list
Name                     ID     Mem  VCPUs       State       Time(s)
Domain-0                  0    1024      2      r-----        318.4
host01                    4    2048      1      -b----         24.7
host02                    5    2048      1      -b----         13.7
```

- Do not start the **host03** VM.

Practice 1-3: Connecting to a Guest VM

Overview

In this practice, you connect to a virtual machine guest.

Assumptions

- You are logged on to **dom0**.
- You have a terminal window open.
- You are the `root` user.
- Both **host01** and **host02** VMs are running.

Tasks

1. Connect to **host01** guest by using **vncviewer**.
   a. Determine the vnc port number for **host01** by running the `xm list -l host01 | grep location` command.

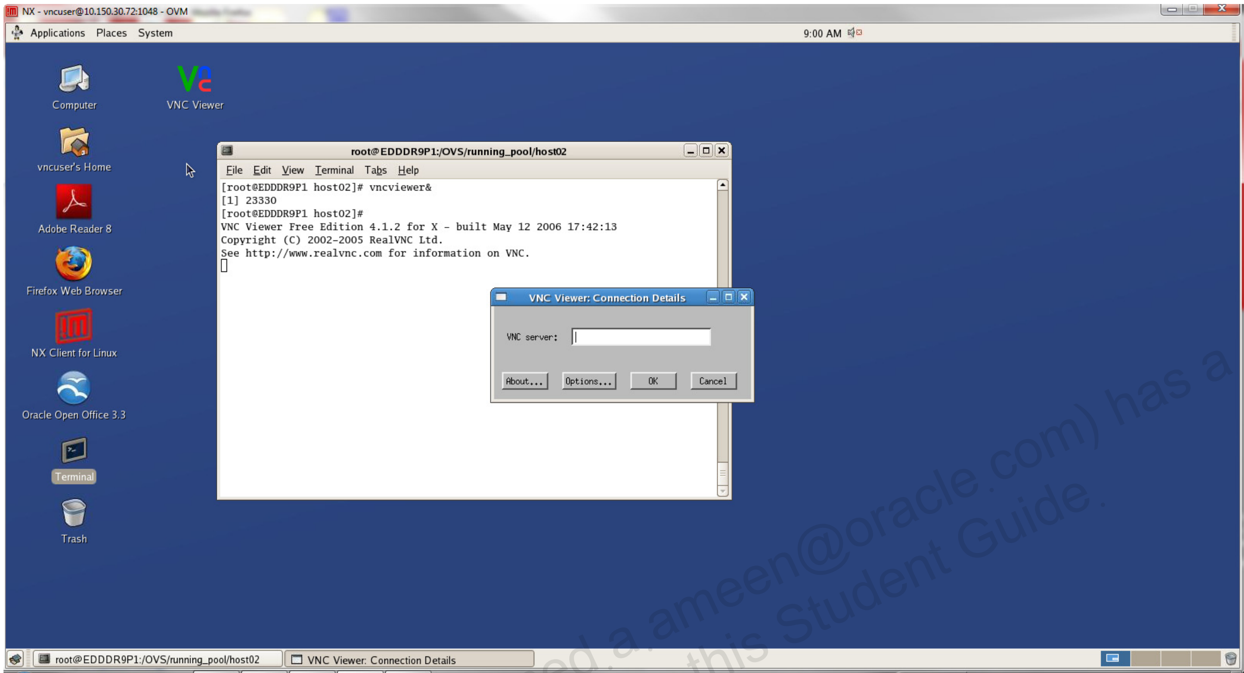
```
# xm list -l host01 | grep location
                (location 0.0.0.0:5900)
                (location 2)
```

   - The sample shown indicates that the port number is **5900**. This may not be true in your case.

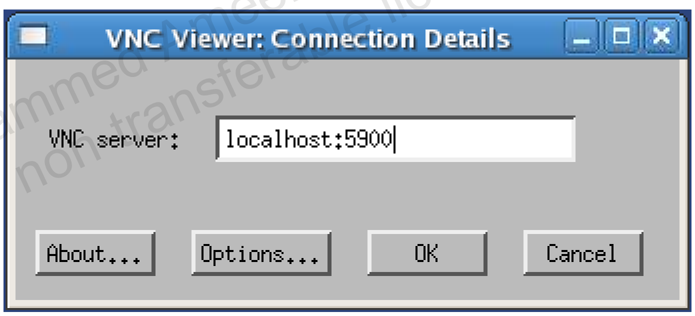
b. Run the `vncviewer&` command:

```
# vncviewer&
```

- The **VNC Viewer: Connection Details** dialog box is displayed:



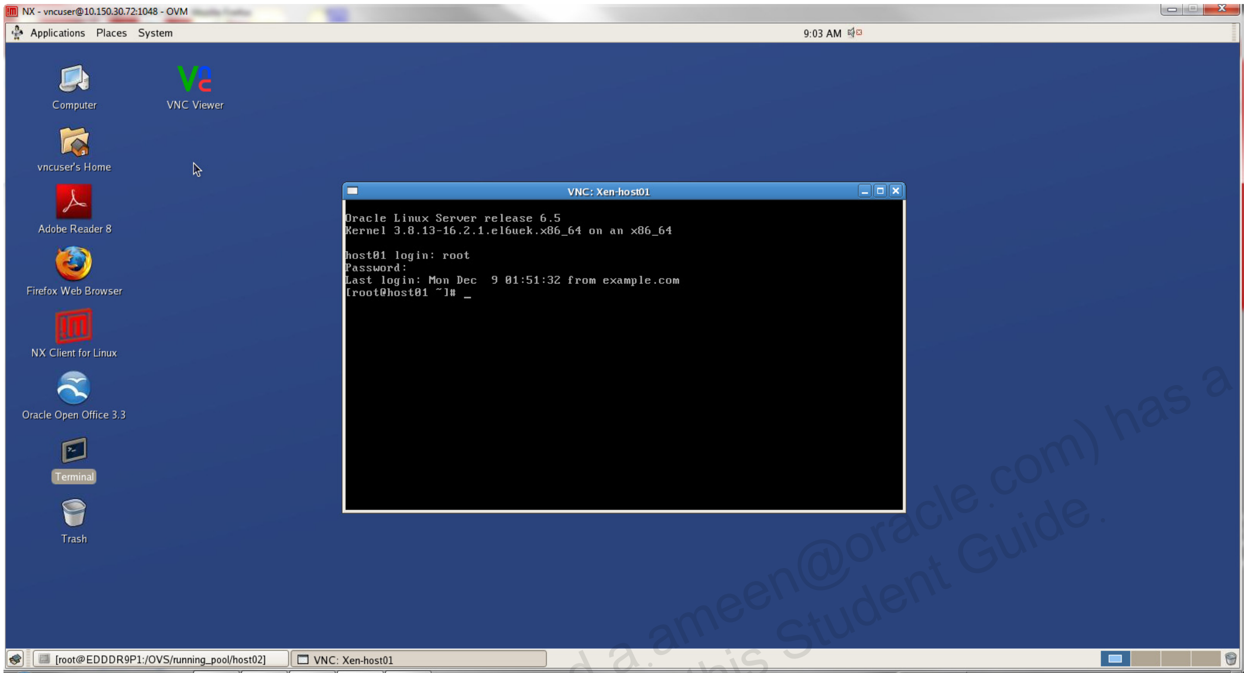
c. Enter the `localhost:`*`<port_number>`* command, substituting the port number displayed from the previous `xm list` command. For example, if the port number is `5900`, enter `localhost:5900` and click **OK**.



- You could also enter `localhost:0` (omit the "`590`" from the port number).

d. The window, which is shown as follows, is displayed. Log in as `root` with password `oracle` (all lowercase).

- If the screen is blank, press Enter to display the login prompt.



e. Use the `hostname` command to confirm that you are logged on to the **host01** VM.

```
# hostname
host01.example.com
```

f. Log out by entering either the `logout` command or the `exit` command.

```
# logout
```

g. Close the VNC window by clicking the **X** at the top-right corner of the window.

2. Connect to **host01** guest by using `ssh`.

a. Alternatively, use the `ssh` command to connect to the VM guest.

- You are presented with the RSA key fingerprint question the first time you connect. Answer **yes** to the question and then enter the `root` password `oracle`.
- RSA and the `ssh` utility are covered in Lesson 15, “OpenSSH.”

```
# ssh host01
The authenticity of host ‘host01 (192.0.2.101)’ can’t be
established. RSA key fingerprint is ...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added ‘host01,192.0,2,101’ (RSA) to the
list of known hosts.
root@host01’s password: oracle
```

- Most of the practices instruct you to log in using `vncviewer` rather than `ssh`.

b. To confirm that you have connected to **host01**, enter the `hostname` command:

```
# hostname
host01.example.com
```

c. Terminate the `ssh` connection by entering the `exit` command. To verify that you are back to **dom0**, enter the `hostname` command:

```
# exit
Connection to host01 closed.
# hostname
EDDDR9P1
```

- In this example, the host name is **EDDR9P1**. Your host name is different.

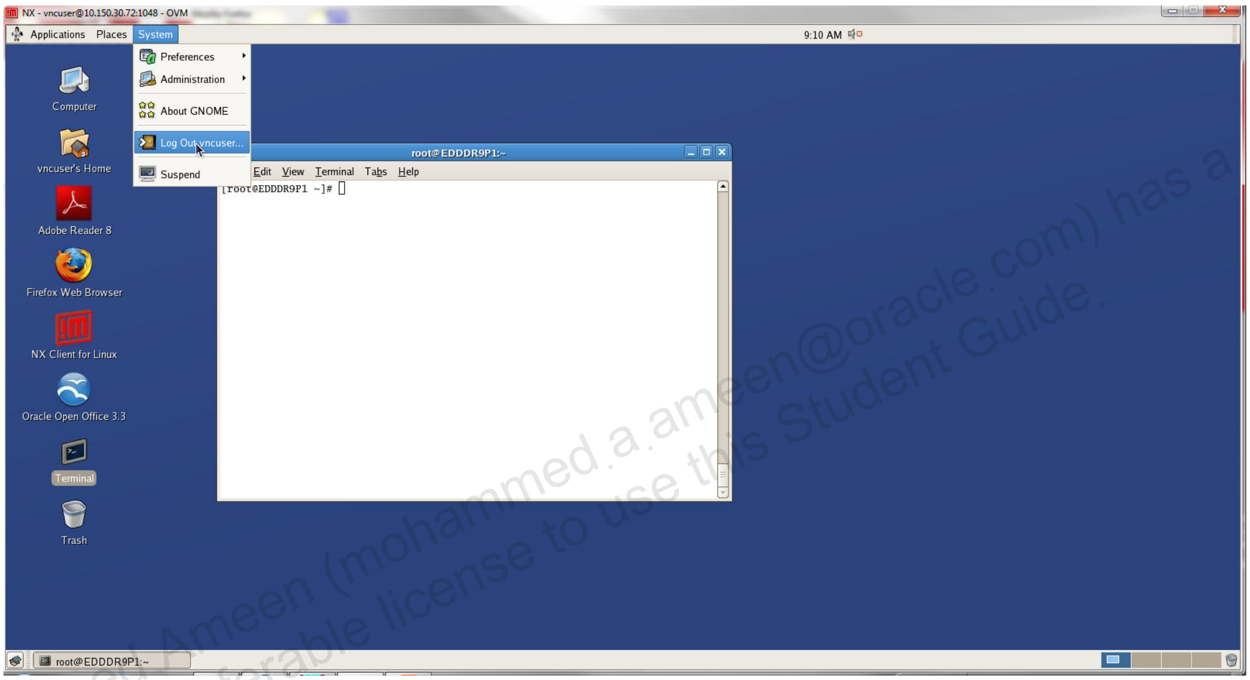
Practice 1-4: Logging Off Your Student PC

Overview

In this practice, you learn how to log off your system.

Tasks

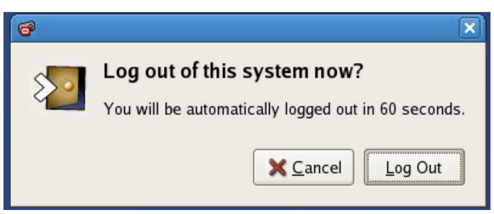
1. Log off your student PC.
   a. Open the **System** menu on the GNOME desktop.



   b. Select **Log Out vncuser** from the System menu.

   Do not perform this next step. This is information only.

   c. Click the **Log Out** button to log out.



- Alternatively, you could log off by clicking the red X in the upper-right corner of the window. You are then prompted to click Terminate to exit the session and close all running programs.

Practices for Lesson 2: Introduction to Oracle Linux

Chapter 2

Practices for Lesson 2: Introduction to Oracle Linux

Practices Overview

In these practices, you test your knowledge of Linux and of Oracle's contributions and commitment to Linux. You also view the different kernel packages that are installed with Oracle Linux.

Practice 2-1: Quiz – Introduction to Oracle Linux

Overview

In this quiz, you answer questions about the history of Linux in general, as well as Oracle's commitment and contributions to Linux.

Quiz Questions

Choose the best answer from those provided for each multiple choice or True/False question.

1. Which of the following statements are true? (Choose all that apply).
   a. Linux is a UNIX-like operating system named for its developer, Linus Torvalds.
   b. Richard Stallman released version 0.01 of the Linux kernel in September 1991.
   c. The goal of the GNU Project to create a free, UNIX-like operating system was realized in 1992 with the addition of the Hurd kernel.
   d. Distributors of GPL software must provide the binaries at no cost, but they can charge for the source code.
2. Thousands of programmers all over the world participate in the development of the Linux operating system.
   a. True
   b. False
3. Linus Torvalds releases a new version of the mainline kernel approximately:
   a. Every month
   b. Every three months
   c. Every six months
   d. Every year
4. The revision control system for the Linux kernel source code is called:
   a. SCCS
   b. RCS
   c. BitKeeper
   d. Git
5. A Linux distribution is a collection of software built on top of the Linux kernel and offered as a complete package.
   a. True
   b. False

6. Which of the following statements about Oracle’s commitment to Linux are true? (Choose all that apply).
   a. Oracle runs its business on Linux.
   b. Oracle has a global Linux support organization.
   c. Oracle has a dedicated Linux kernel development team.
   d. Oracle’s goal is to deliver the best performing, most modern, and most reliable Linux operating system for the enterprise.
7. Oracle Linux is fully compatible with which of the following Linux distributions?
   a. Debian
   b. Ubuntu
   c. SUSE
   d. RHEL
8. The Unbreakable Enterprise Kernel is optimized for Oracle applications and is the only kernel offered with Oracle Linux.
   a. True
   b. False

Solution 2-1: Quiz – Introduction to Oracle Linux

Quiz Solutions

1. Which of the following statements are true? (Choose all that apply).

   a. Linux is a UNIX-like operating system named for its developer, Linus Torvalds.

2. Thousands of programmers all over the world participate in the development of the Linux operating system.

   a. True

3. Linus Torvalds releases a new version of the mainline kernel approximately:

   b. Every three months

4. The revision control system for the Linux kernel source code is called:

   d. Git

5. A Linux distribution is a collection of software built on top of the Linux kernel and offered as a complete package.

   a. True

6. Which of the following statements about Oracle's commitment to Linux are true? (Choose all that apply).

   a. Oracle runs its business on Linux.

   b. Oracle has a global Linux support organization.

   c. Oracle has a dedicated Linux kernel development team.

   d. Oracle's goal is to deliver the best performing, most modern, and most reliable Linux operating system for the enterprise.

7. Oracle Linux is fully compatible with which of the following Linux distributions?

   d. RHEL

8. The Unbreakable Enterprise Kernel is optimized for Oracle applications and is the only kernel offered with Oracle Linux.

   b. False. It is true that the Unbreakable Enterprise Kernel is optimized for Oracle applications. But Oracle Linux also includes a Red Hat–compatible kernel for customers requiring strict RHEL compatibility.

Practice 2-2: Viewing Kernel Information

Overview

In this practice, you connect to the **host01** VM, view the installed kernel packages, and determine which kernel is running. You also view the text file used by Oracle support teams to easily identify the active kernel. Note that the displayed sample output might not represent your own system.

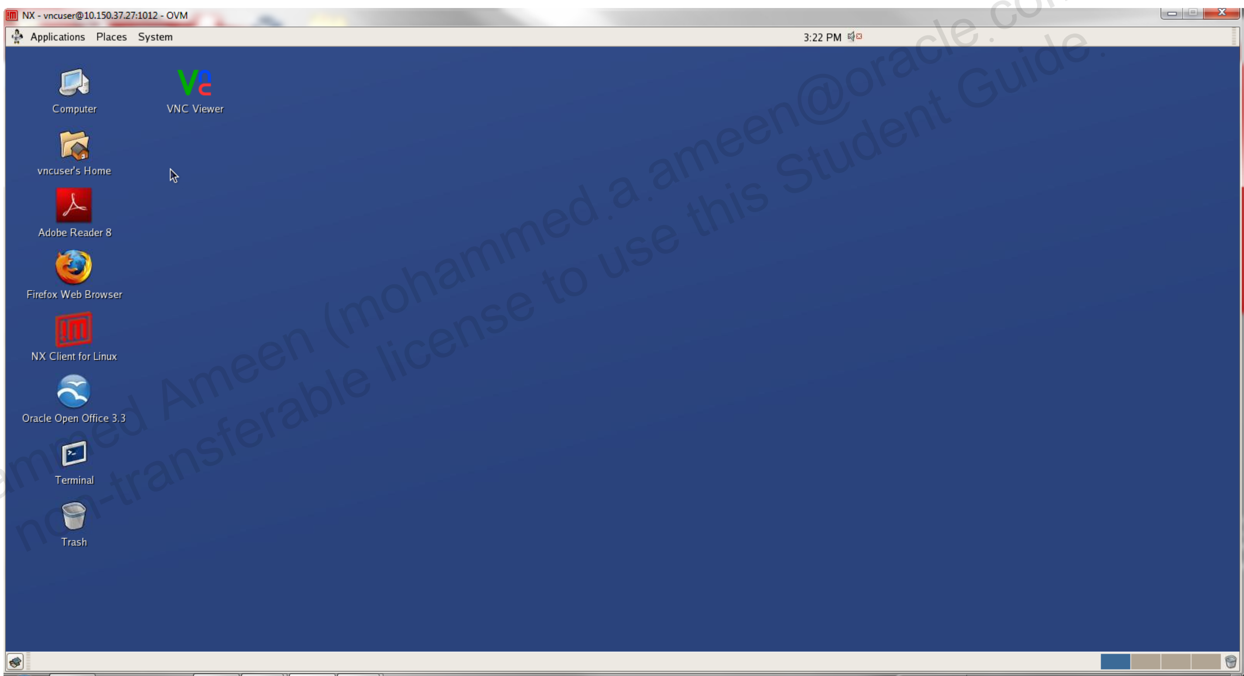
Assumptions

- You are logged on to **dom0**.
- VM **host01** is running.

Tasks

1. Open a terminal window.

- Begin this task from the GNOME virtual desktop window:

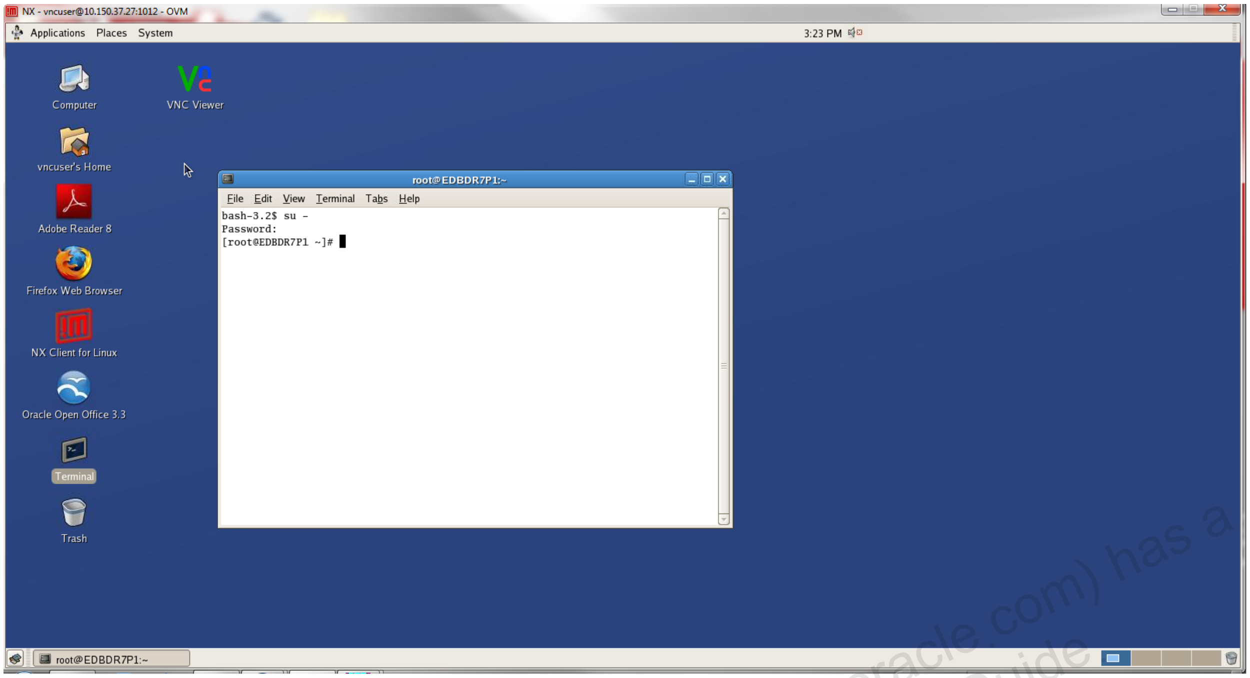


Double-click the **Terminal** icon on the GNOME desktop.

- A terminal window opens.

2. Become the `root` user.

- Enter the commands from an open terminal window:



Become `root` by using the `su -` command. The `root` password is `oracle`. Use the `whoami` command to print the user identity to confirm that you are `root`:

```
$ su -
Password: oracle
# whoami
root
```

3. Connect to **host01** guest as `root` user (password is `oracle`) by using `ssh`.

   Use the `ssh` command to connect to the VM guest **host01** as `root`. The password is `oracle`. Confirm that you have connected to **host01** by entering the `hostname` command:

```
# ssh host01
root@host01's password: oracle
# hostname
host01.example.com
```

4. List the kernel release that is currently running on your system.

   Use the `uname -r` command to print the kernel release:

```
# uname -r
3.8.13-16.2.1.el6uek.x86_64
```

- Note that the Unbreakable Enterprise Kernel is running by default.

5. List the kernel packages that are installed on your system.

   Use the `rpm -qa` command to query the kernel packages installed on your system:

```
# rpm -qa | grep "^kernel"
kernel-2.6.32-431.el6.x86_64
kernel-uek-3.8.13-16.2.1.el6uek.x86_64
...
```

   - Note that the following two `kernel` packages are installed (associated `firmware` packages, `headers` and `doc` packages are also installed):
     - Unbreakable Enterprise Kernel package (`kernel-uek-3.8.13-16.2.1.el6uek.x86_64`)
     - Red Hat–compatible kernel package (`kernel-2.6.32-431.el6.x86_64`)

6. View the `/etc/*release` files on your system.

   Change to the `/etc/` directory and view the "release" files:

```
# cd /etc
# ls -l *release
...
-rw-r--r--. oracle-release
-rw-r--r--. redhat-release
lrwxrwxrwx. system-release -> oracle-release
# cat oracle-release
Oracle Linux Server release 6.5
# cat redhat-release
Red Hat Enterprise Linux Server release 6.5 (Santiago)
```

   - Note that `/etc/system-release` is a symbolic link to `/etc/oracle-release`.

7. Log out from **host01**.

```
# logout
Connection to host01 closed.
```

Practices for Lesson 3: Installing Oracle Linux

Chapter 3

Practices for Lesson 3: Installing Oracle Linux

Practices Overview

In these practices, you:

- Install Oracle Linux on the **host03** virtual machine
- Run FirstBoot on **host03**
- Log in to **host03** and perform shutdown operations

Practice 3-1: Installing Oracle Linux

Overview

In this practice, you install Oracle Linux on the **host03** virtual machine (VM).

Assumptions

- You are logged on as the `root` user on **dom0**.
- You have a terminal window open.

Tasks

1. Create the **host03** VM.

   Change to the `/OVS/running_pool/host03` directory and use the `xm create` command as follows:

```
# cd /OVS/running_pool/host03
# xm create vm.cfg
Using config file "./vm.cfg".
Started domain host03 (id=6)
```

2. Connect to the **host03** guest by using **vncviewer**.

   Determine the vnc port number for **host03** by running the `xm list –l host03 | grep location` command.

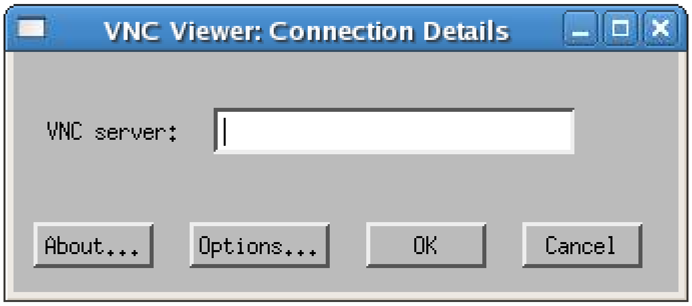
```
# xm list –l host03 | grep location
                (location 0.0.0.0:5903)
                (location 3)
```

   - The sample shown indicates that the port number is **5903**. Your port number might be different.

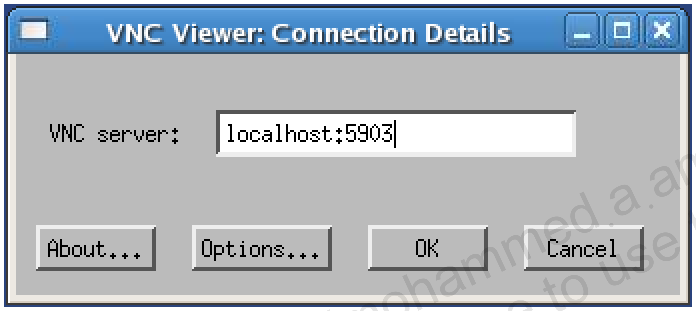
Run the `vncviewer&` command.

```
# vncviewer&
```

- The VNC Viewer: Connection Details dialog box is displayed.

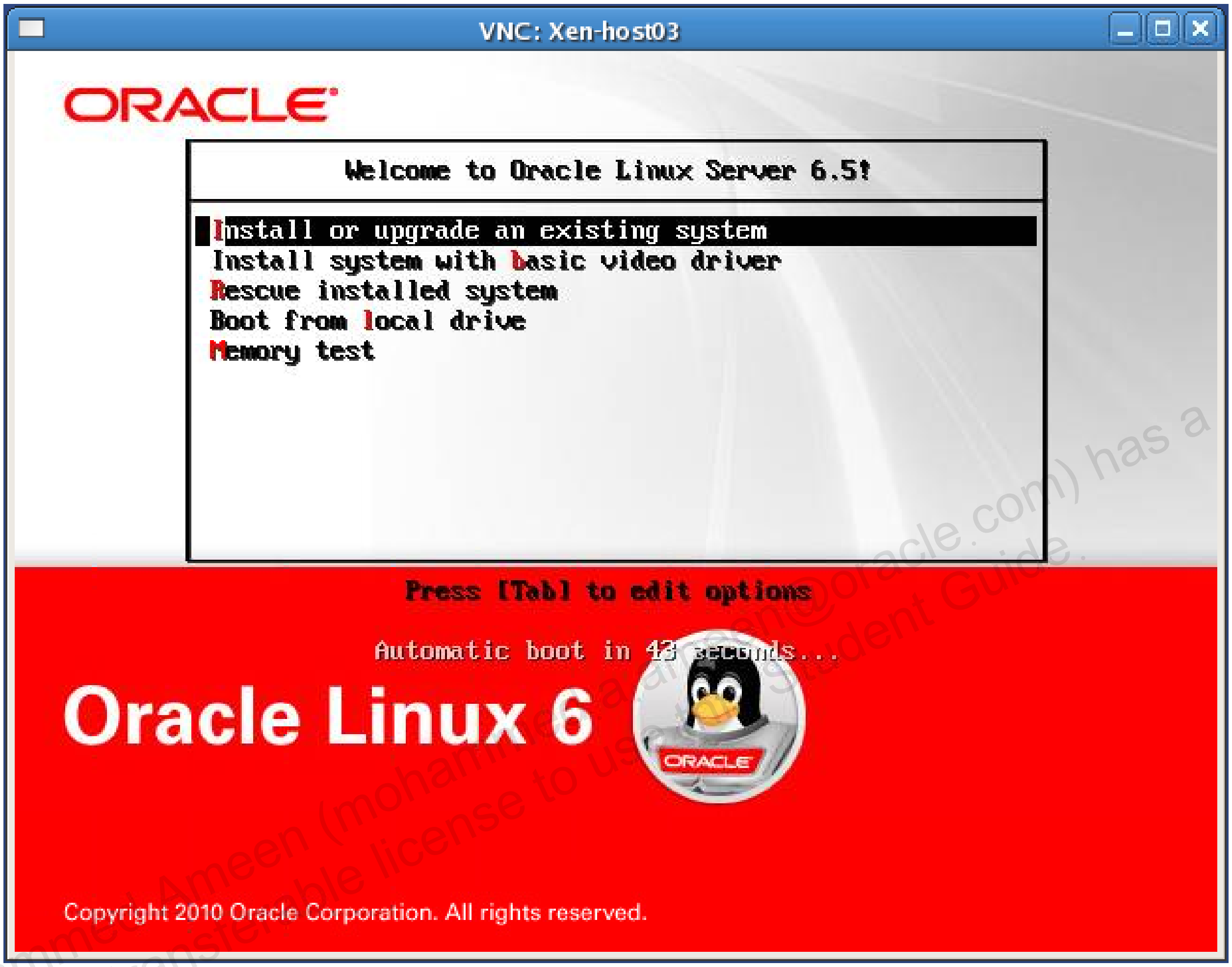


Enter `localhost:<port_number>`, substituting the port number displayed from the previous `xm list –l host03 | grep location` command. For example, if the port number is `5903`, enter `localhost:5903` and click **OK**.



- The Oracle Linux boot menu appears.

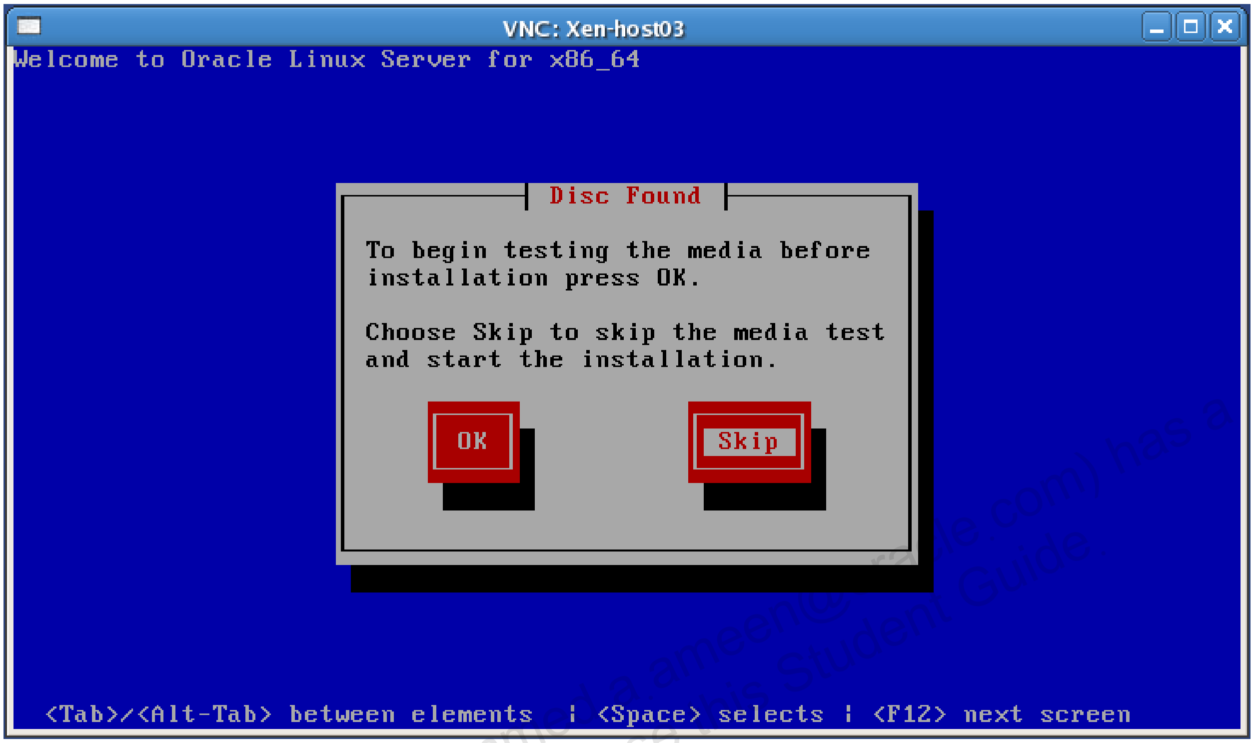
3. Press Enter to select **Install or upgrade an existing system** from the Oracle Linux boot menu.



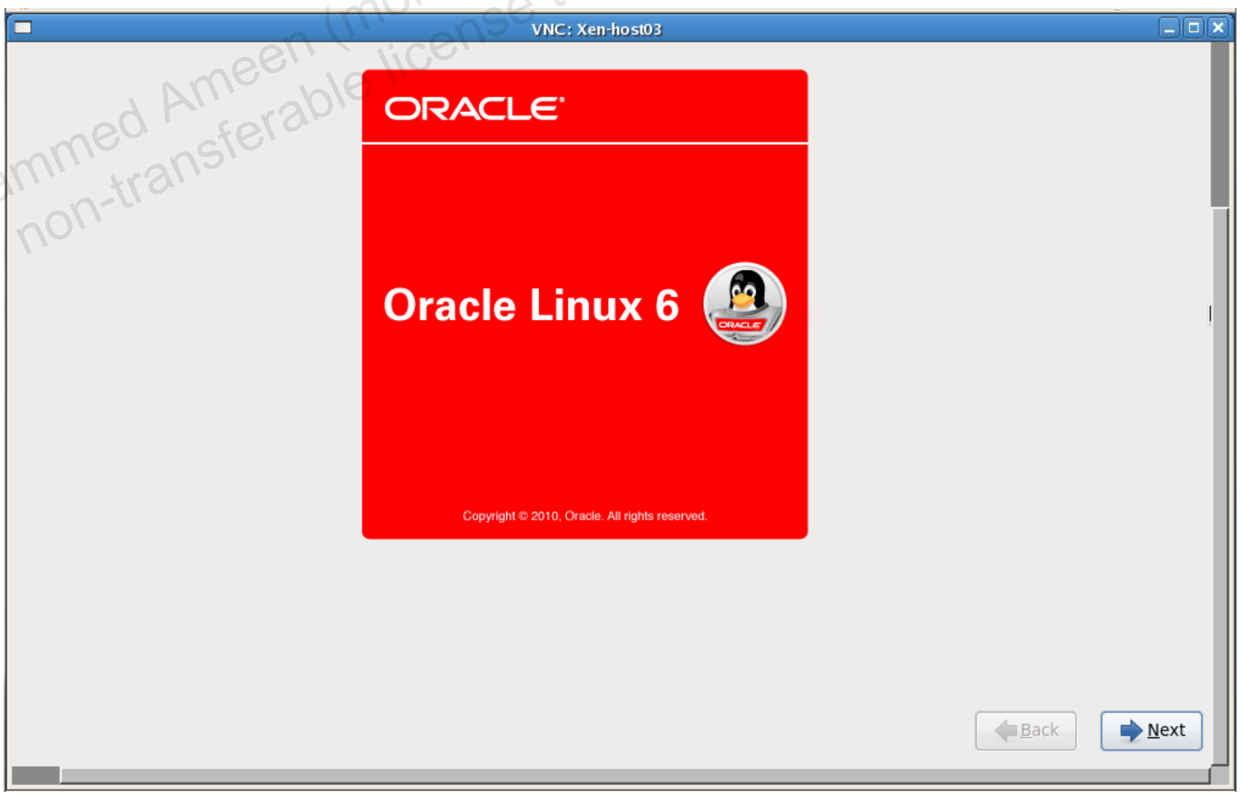
- The default option of **Install or upgrade an existing system** is selected automatically after 60 seconds. Therefore, this boot menu may or may not appear, depending on how quickly you connected to **host03** using **vncviewer**.

4. The next window to appear is the Media Test window.

   Use the Tab key to select the **Skip** button, and then press Enter.

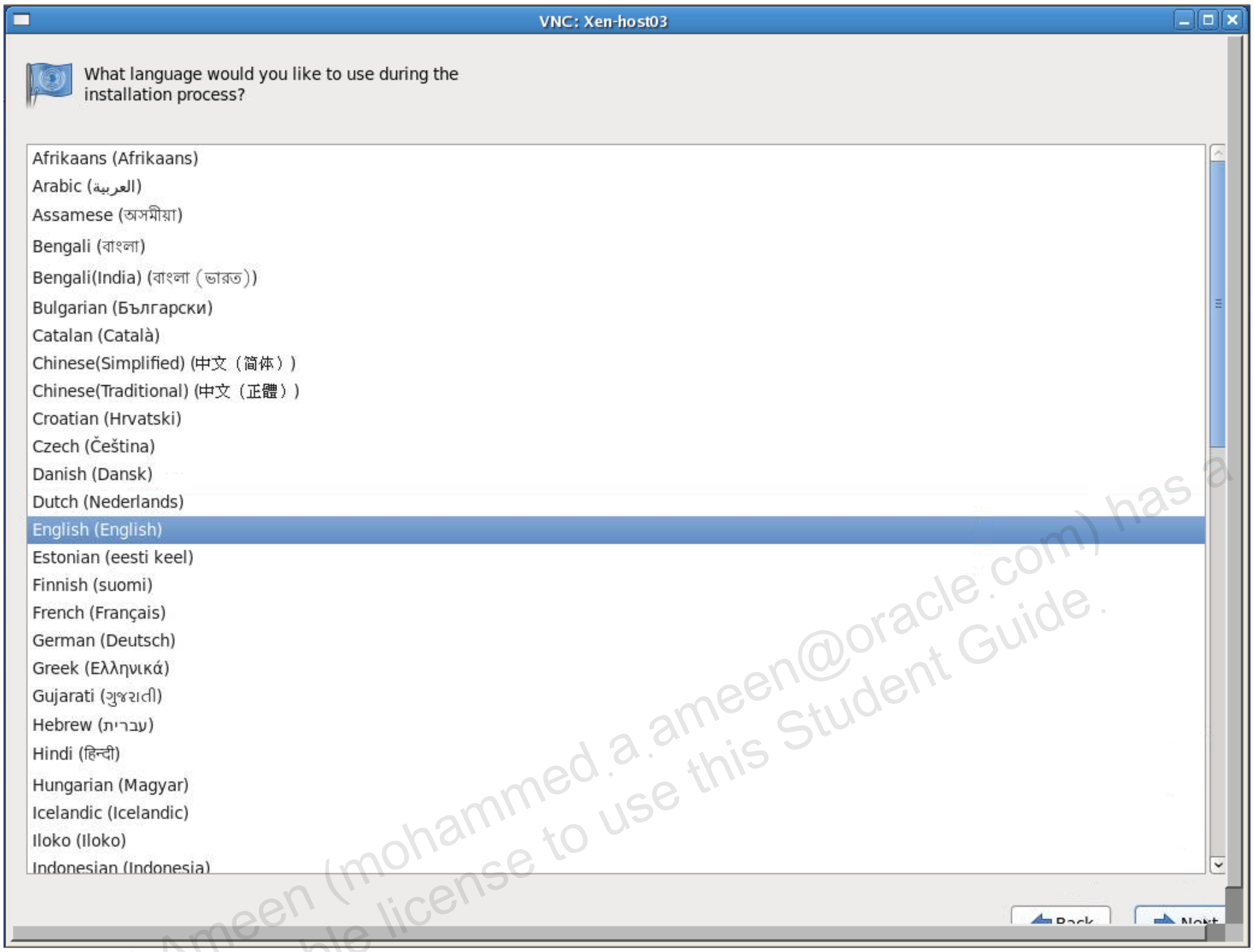


5. The Logo window appears.



   Scroll down if necessary and click the **Next** button.

6. The Language Selection window appears.

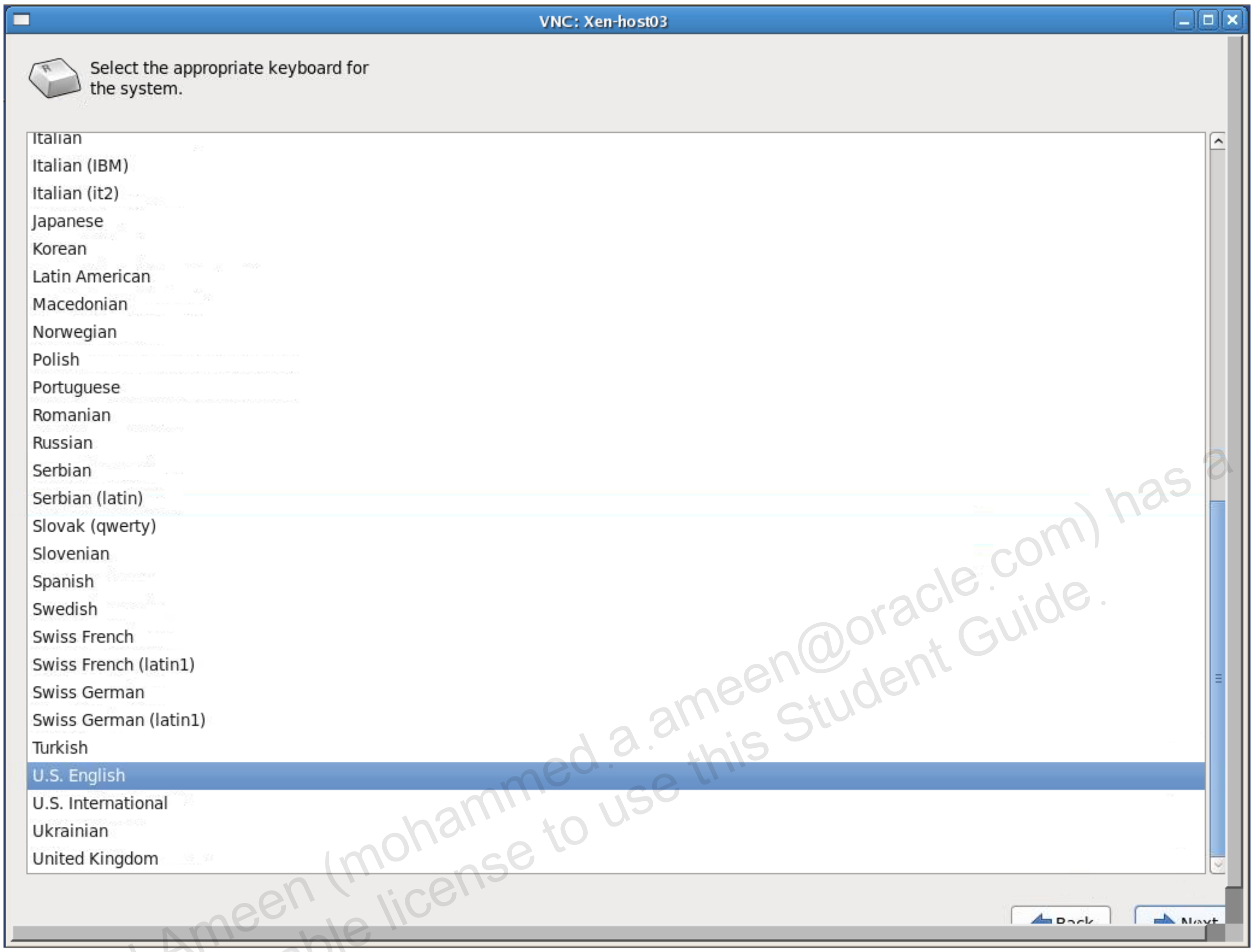


a. Select the appropriate language.

If taking this class in the United States, select **English (English)**.

b. Scroll down if necessary and click **Next**.

7. The Keyboard Selection window appears.

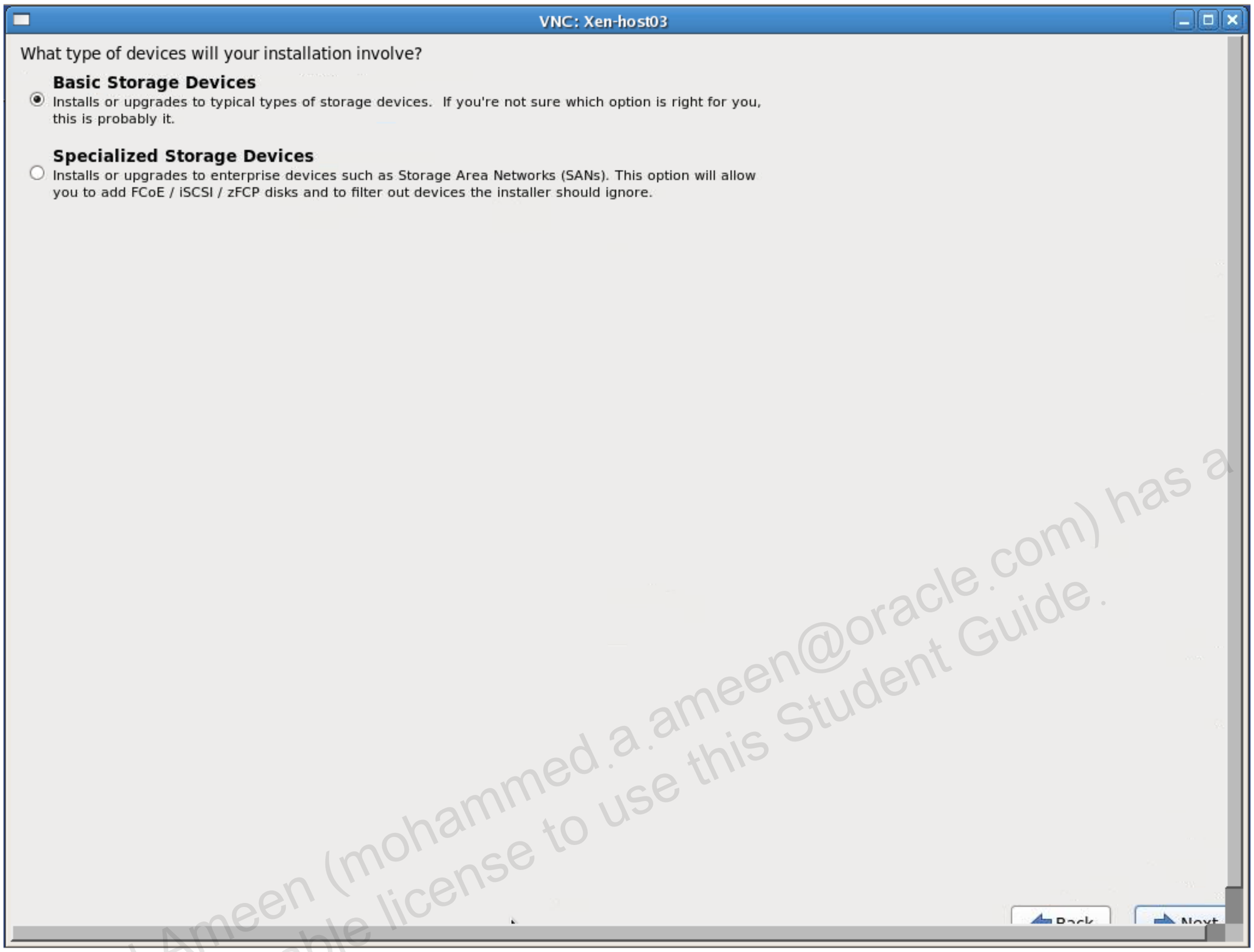


a. Select the appropriate keyboard for your system.

   If taking this class in the United States, select **U.S. English**.

b. Scroll down if necessary and click **Next**.

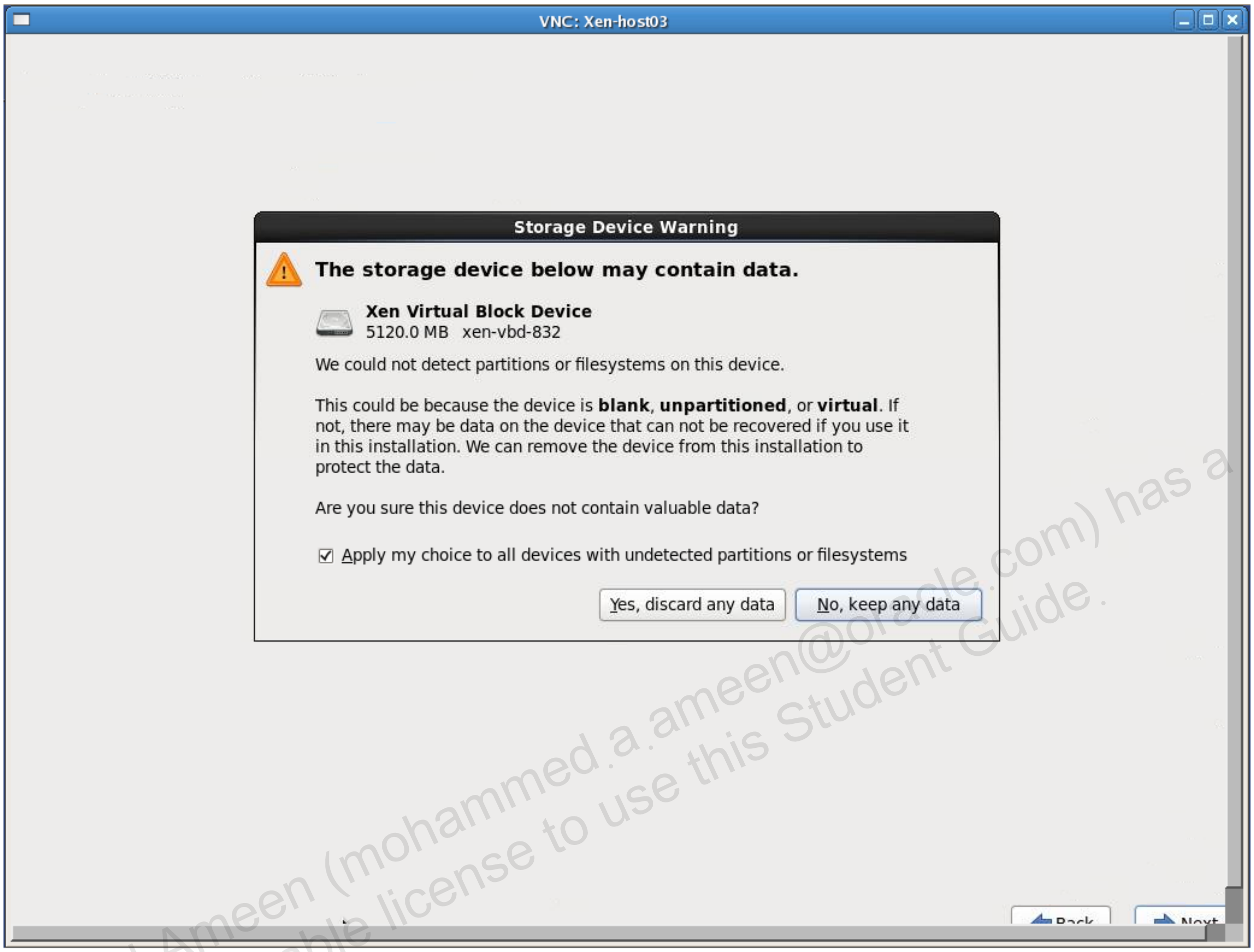
8. The Storage Devices Selection window appears.



a. Ensure that **Basic Storage Devices** is selected.

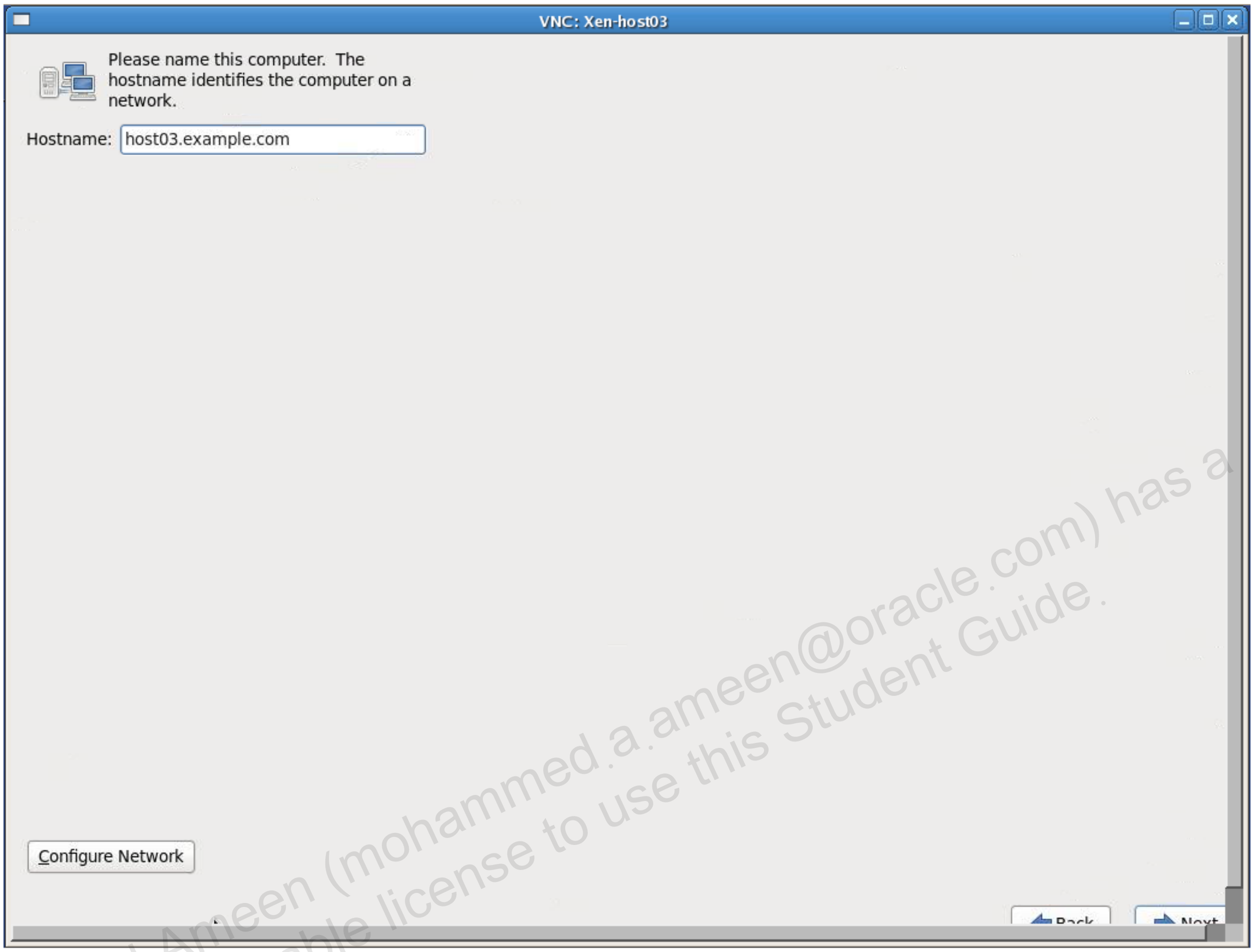
b. Scroll down if necessary and click **Next**.

9. The Storage Devices Warning dialog box appears.



Click the **Yes, discard any data** button.

10. The Set Hostname window appears.

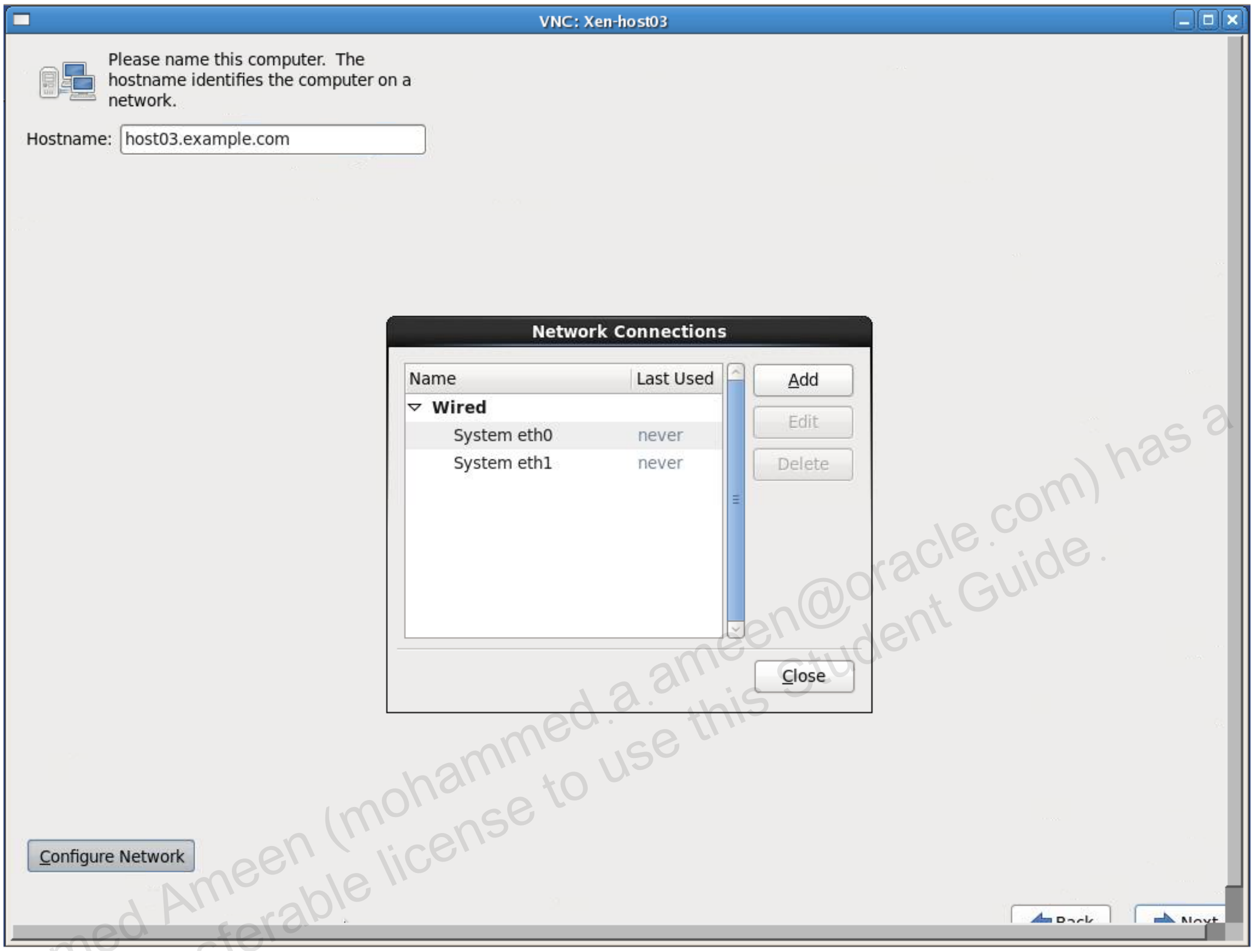


a. Enter `host03.example.com` as the Hostname.

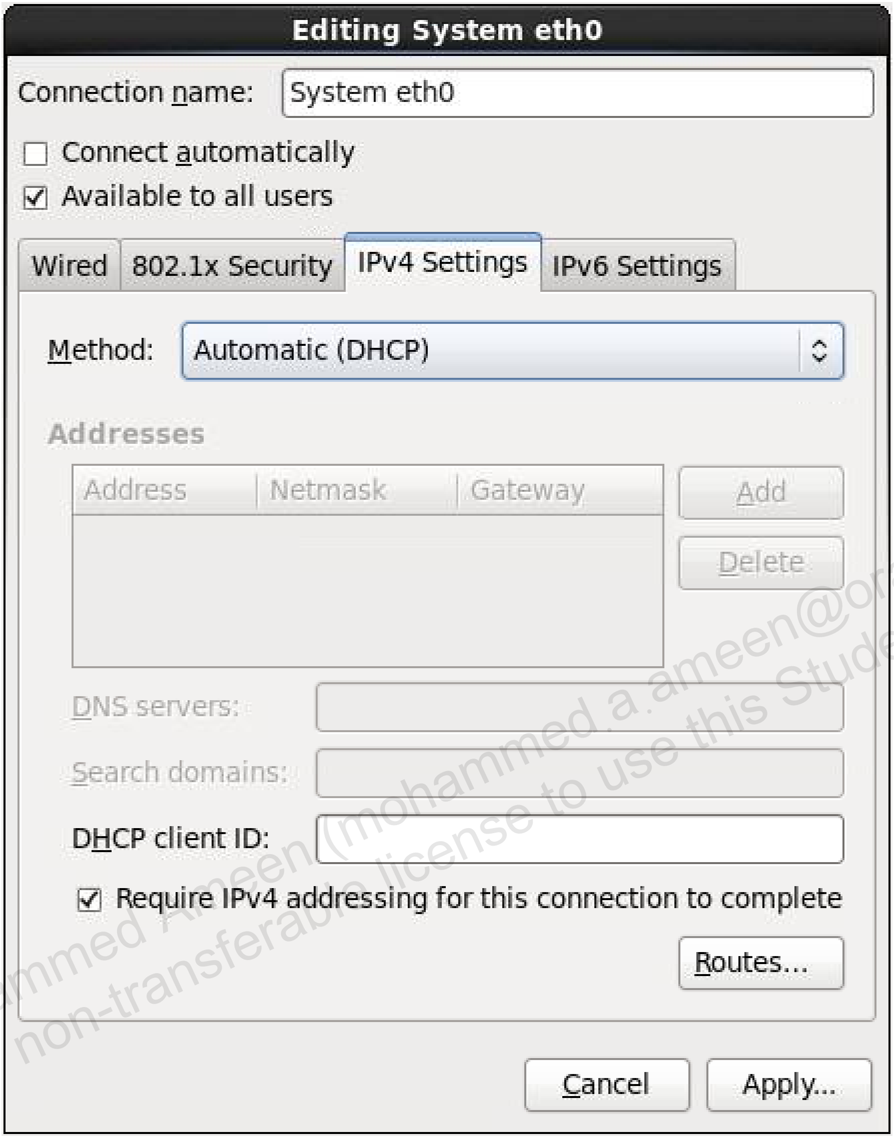
b. Scroll down if necessary and click the **Configure Network** button.

11. Configure Network.

- The Network Connections window appears.



a. Select **System eth0** from the list and click **Edit**. Then click the **IPv4 Settings** tab to display the following screen.



b. Make the following changes:

1) Select **Connect automatically**.

2) Change **Method**=Manual.

3) Click **Add** to add the following:

a) **Address**=192.0.2.103

b) **Netmask**=24

c) **Gateway**=192.0.2.1

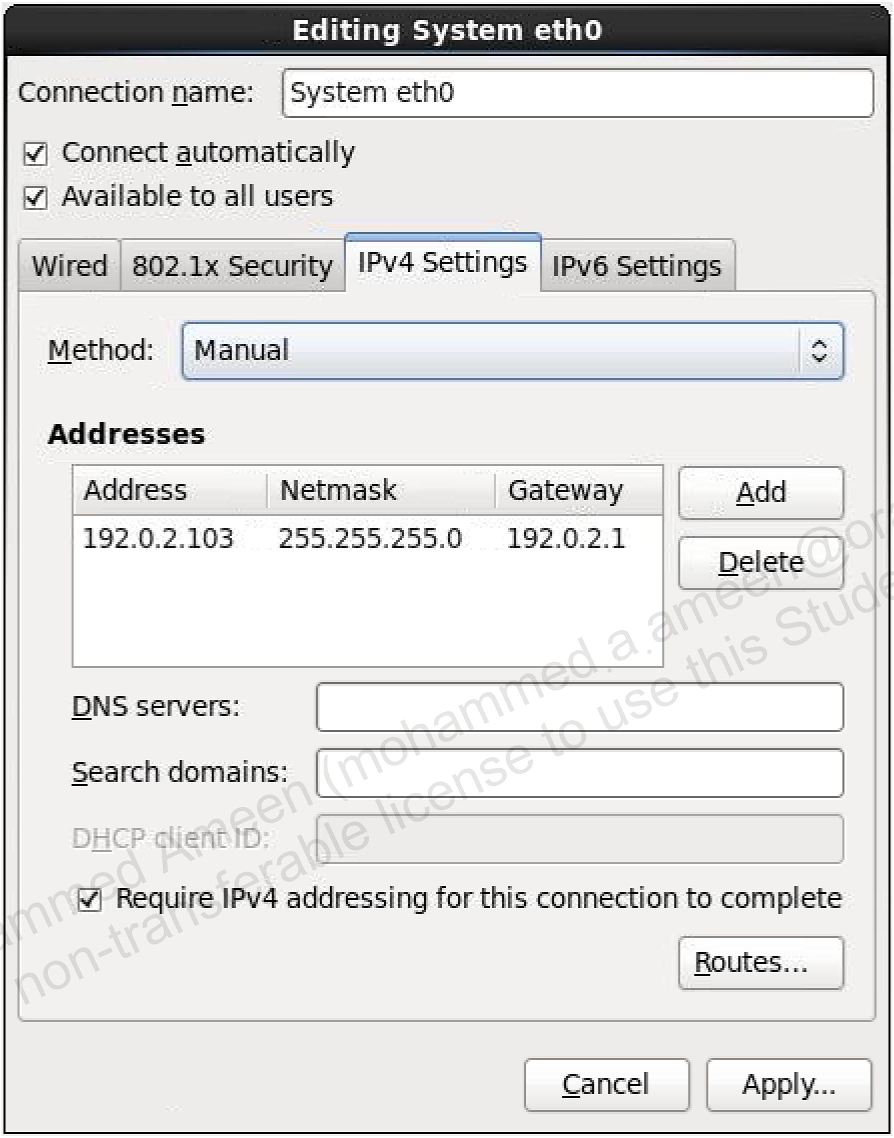
4) Accept the defaults for all other settings.

c. Click **Apply**.

d. The Network Connections window appears, select **System eth0** and click **Edit**.

1) Click the **IPv4 Settings** tab.

- The screen should appear as follows.

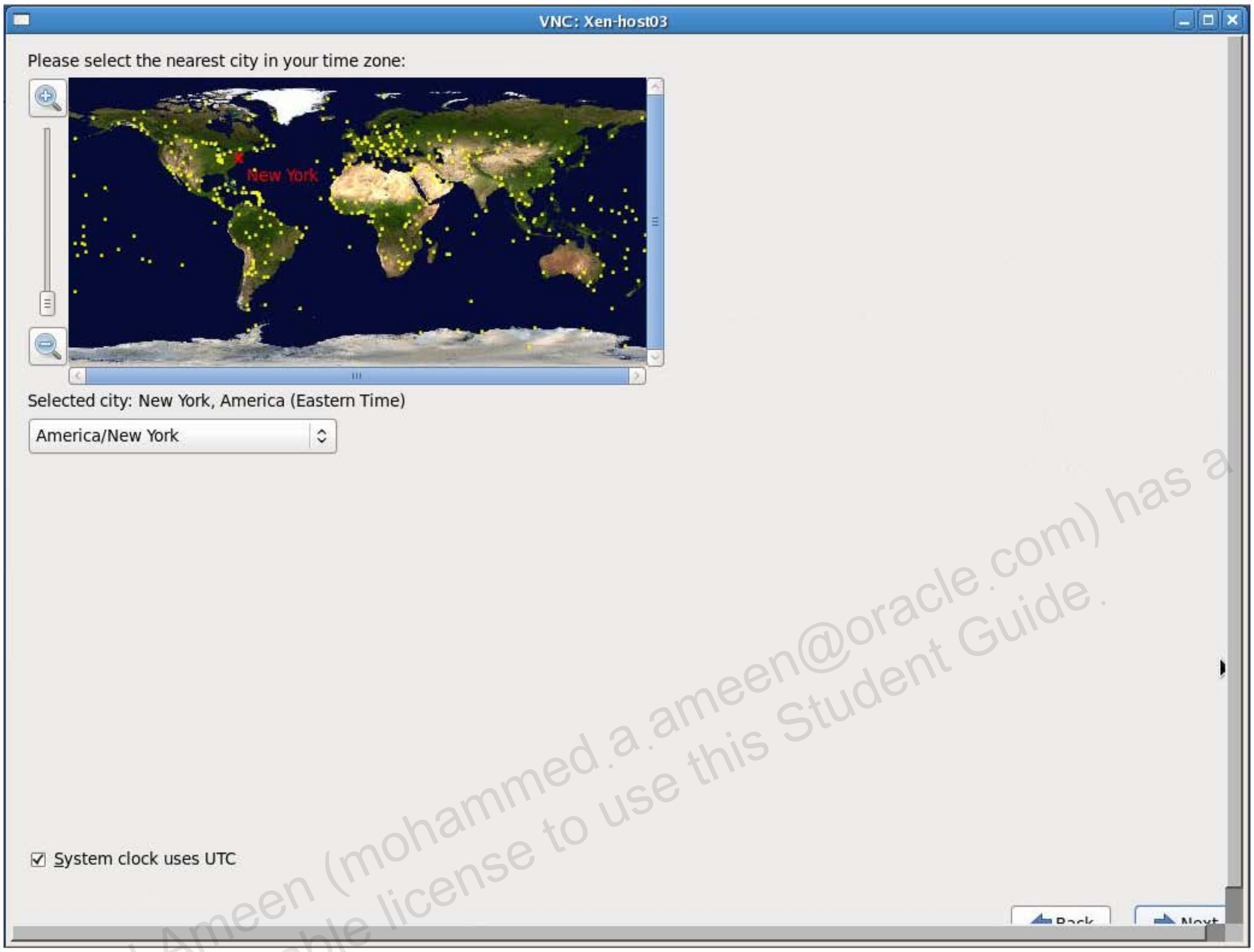


2) Click **Cancel** to close the Editing System eth0 window.

3) Click **Close** to close the Network Connections window.

4) Scroll down if necessary and click **Next**.

12. The Time Zone Selection window appears.



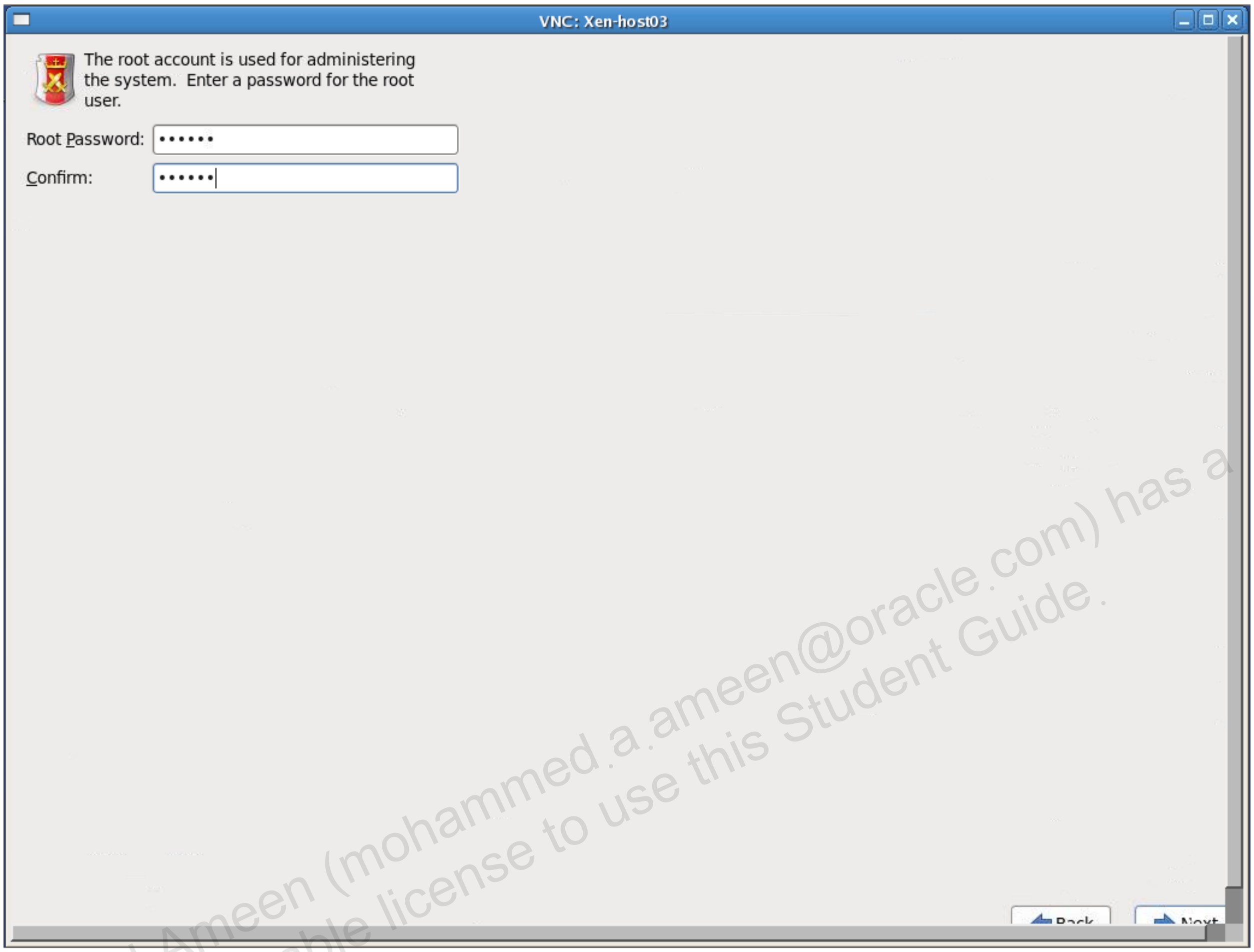
a. Select the appropriate time zone for your location.

   The time zone can be chosen by either selecting from the drop-down list or clicking the map.

b. Select the **System clock uses UTC** check box (this is the default).

c. Scroll down if necessary and click **Next**.

13. The Set Root Password window appears.



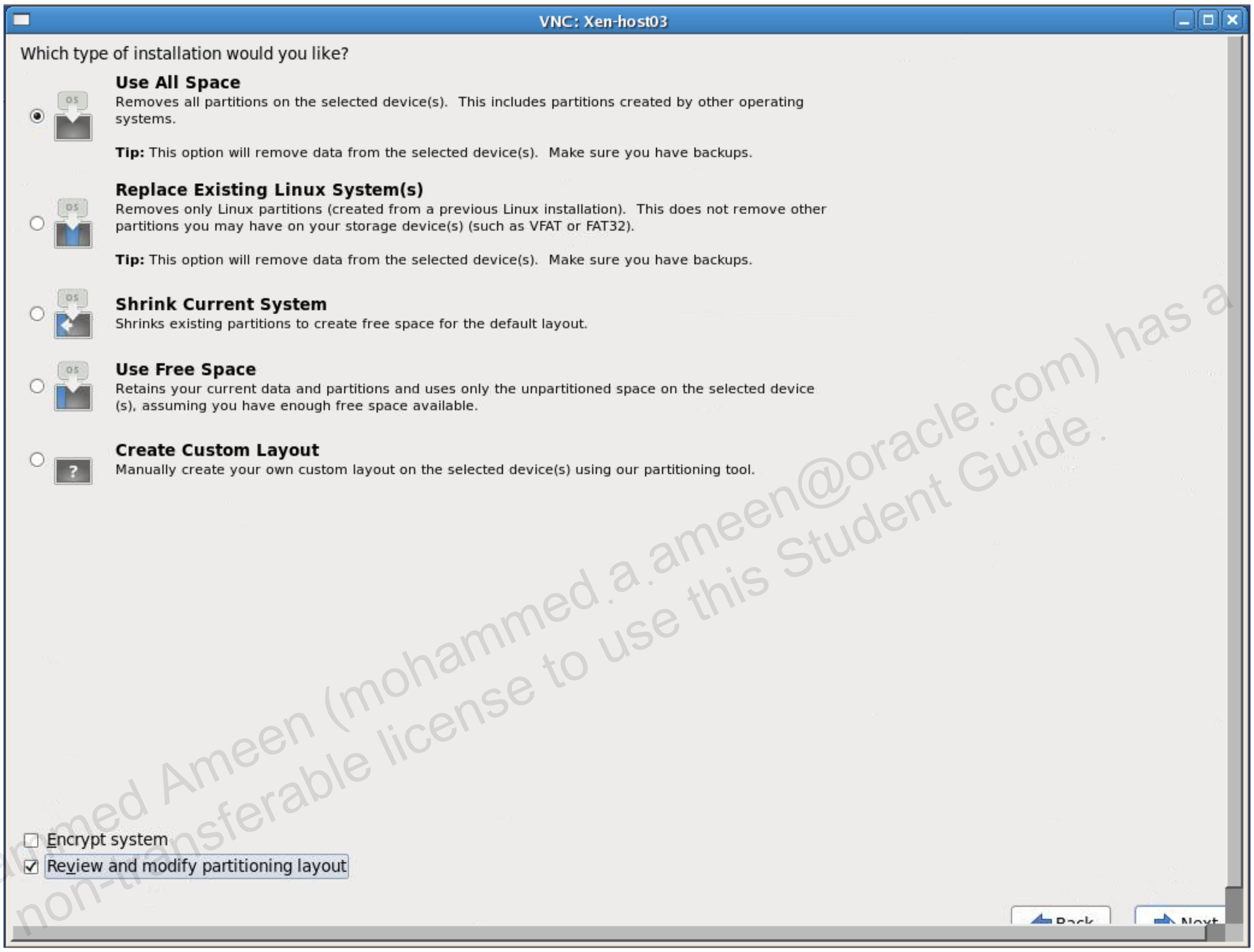
a. Enter the password `oracle` for the `root` user.

b. Press Tab or click the **Confirm** field and re-enter the same password `oracle`.

c. Scroll down if necessary and click **Next**.

d. A Weak Password box appears. Click **Use Anyway**.

14. The Disk Partitioning Setup window appears.
   - You select an option that creates a default partition layout.
   - You then delete the default partitions and create a custom layout.
   - An easier way is to select **Create Custom Layout** from the following window. However, it is beneficial to see the default partition layout.



a. Select **Use All Space**.

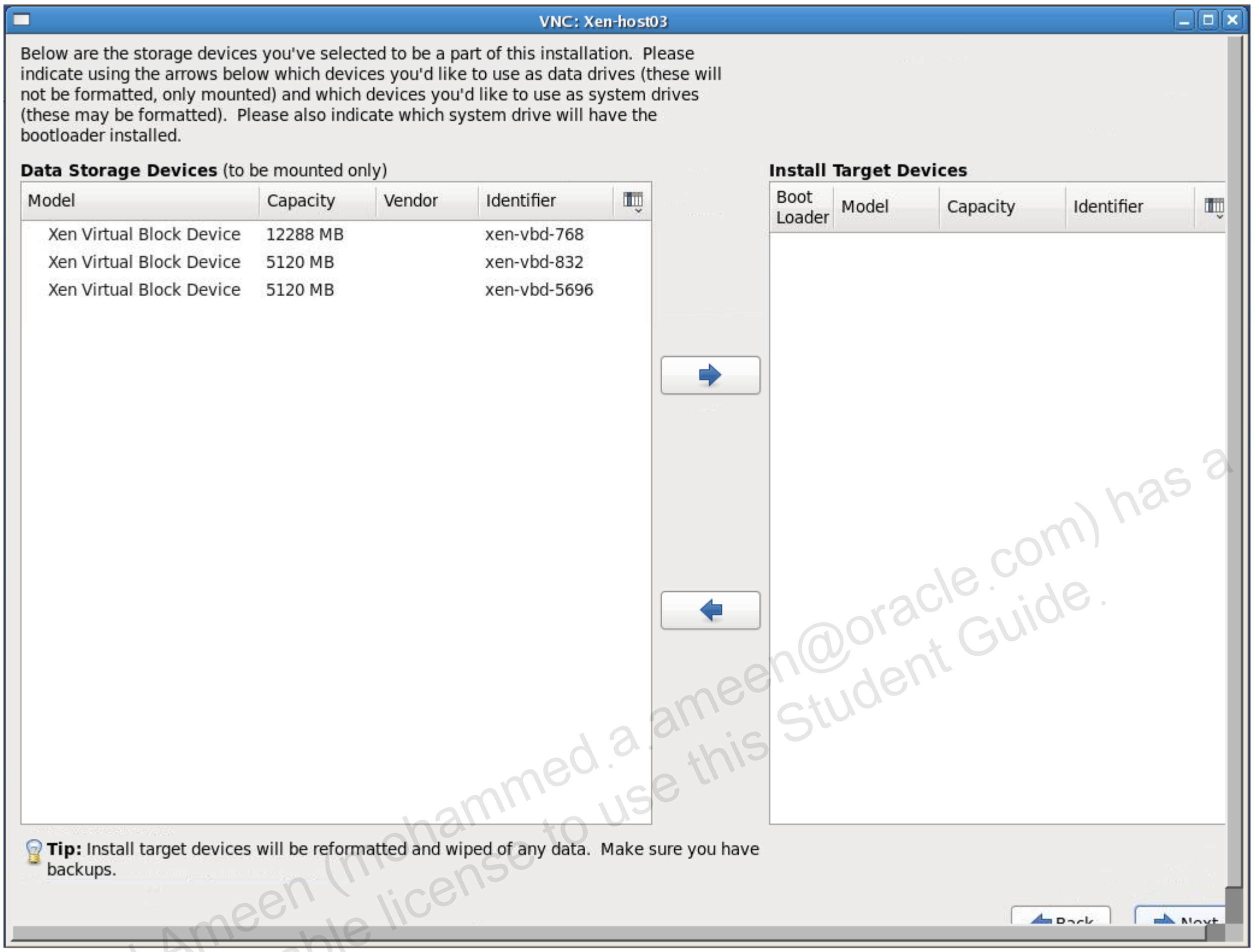
b. Scroll down if necessary.

c. Do not select the **Encrypt system** check box.

d. Select the **Review and modify partitioning** layout check box.

e. Click **Next**.

15. The Storage Devices window appears.

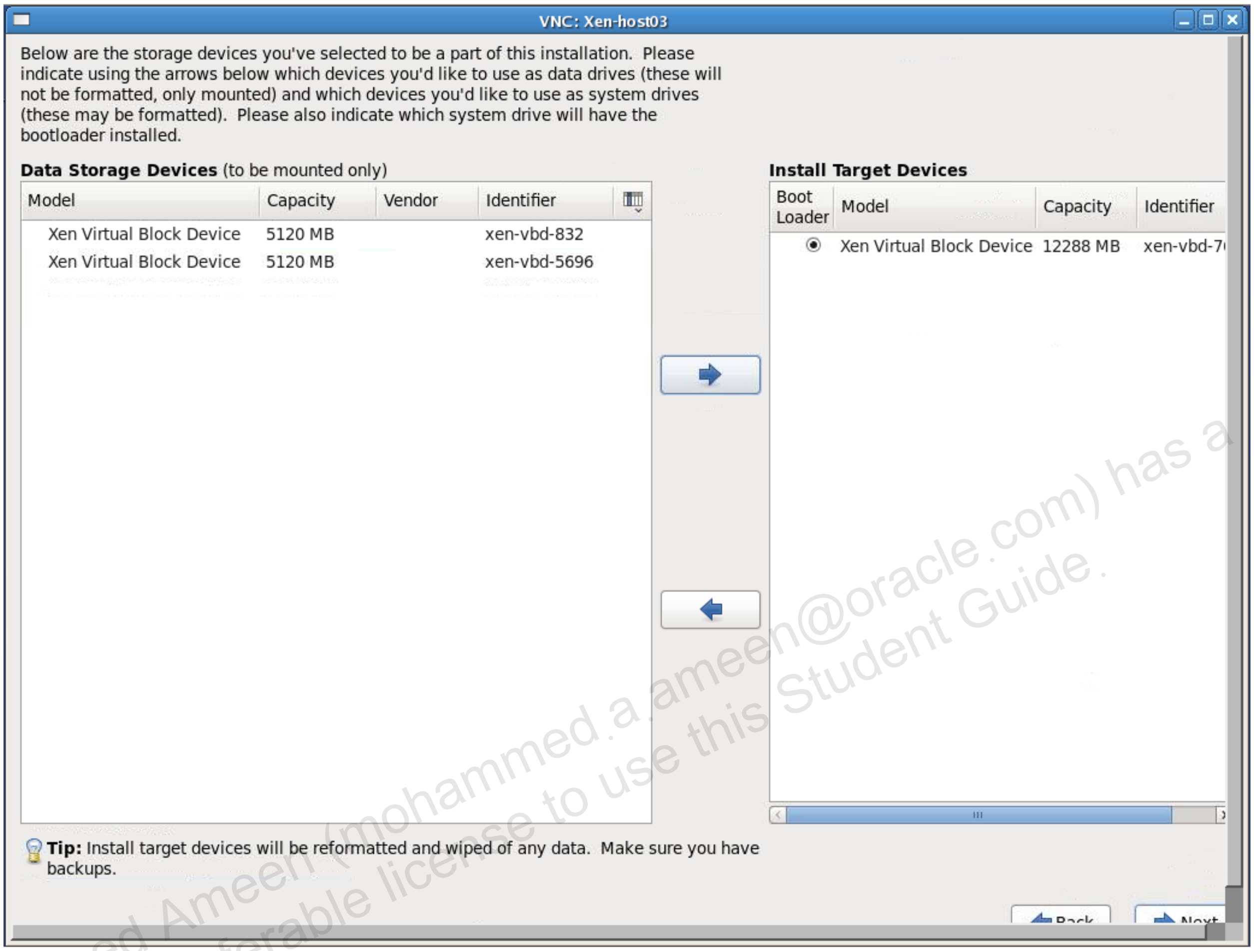


a. Three virtual disk images appear. These were pre-created for you:
   - 12 GB disk image (**system.img**) for the operating system
   - 5 GB disk image (**u01.img**) for the storage administration practices
   - 5 GB disk image (**u02.img**) for the storage administration practices

b. Select the 12 GB disk (**xen-vbd-768** from the list in the Storage Devices window) and click the **right arrow** to move the disk to the right column.

c. Ensure that **Boot Loader** is selected for this 12 GB disk.

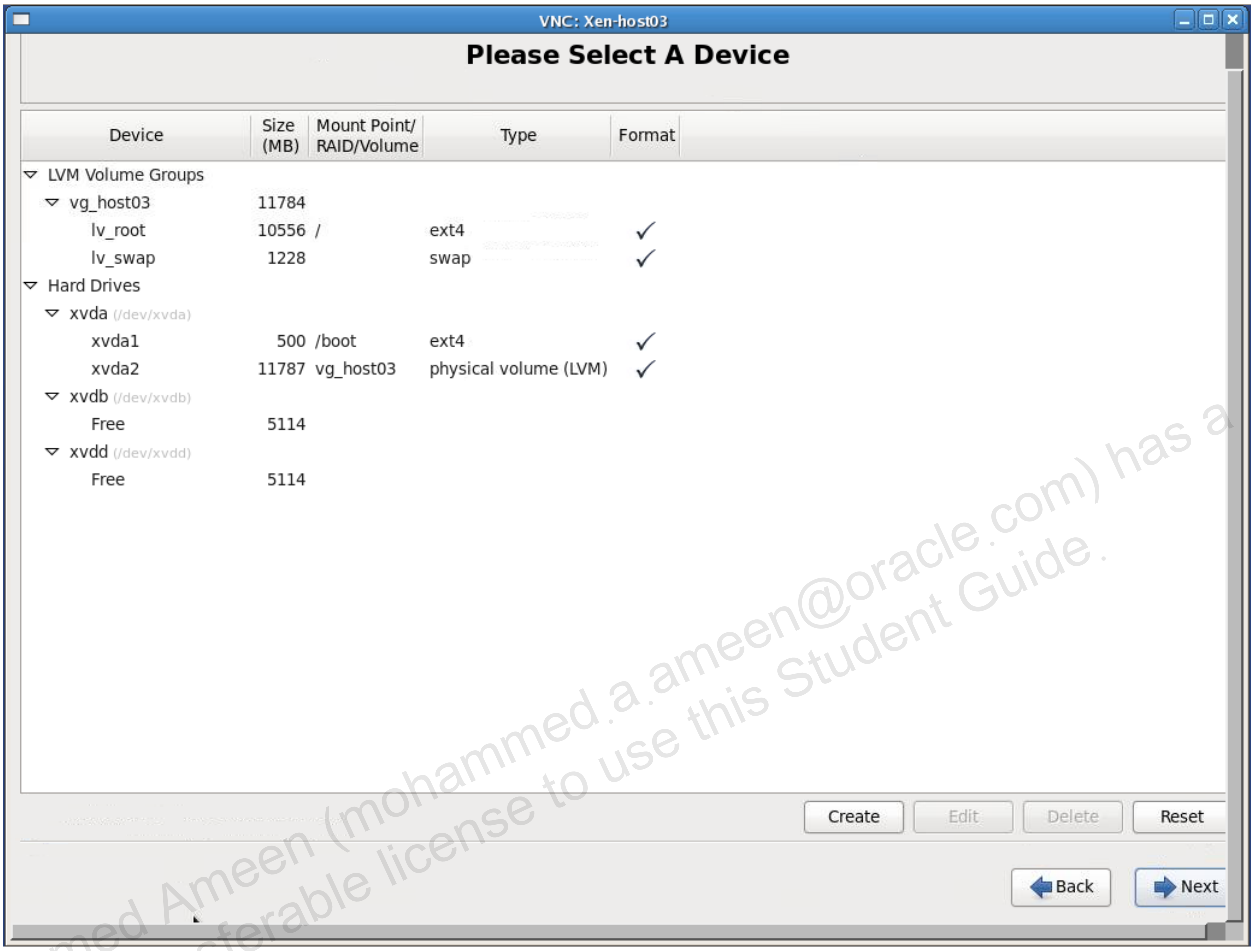
- After you select the designated disk, the Storage Devices window should look like the following screenshot:



d. Scroll down if necessary and click **Next**.

16. Delete all the default partitions.

- The Default Partition Layout window appears:

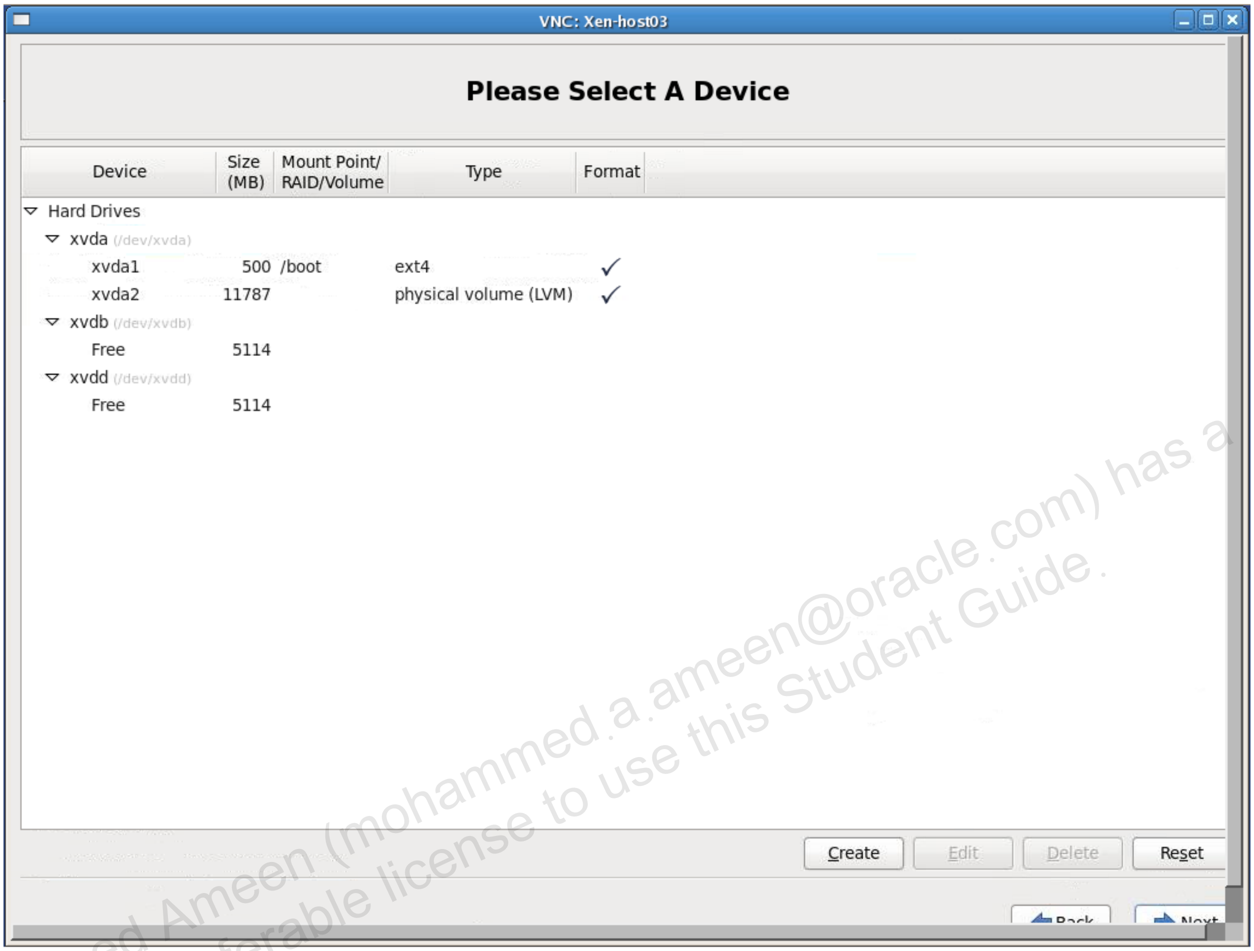


a. Notice that the default layout uses Logical Volume Manager (LVM).
   - One volume group (**vg_host03**) is created.
   - Two logical volumes (**lv_root** and **lv_swap**) are created.

b. Select the first entry, **vg_host03**, scroll down if needed, and click **Delete**.

c. A **Confirm Delete** box appears. Click **Delete** to confirm.

- After you delete a volume group, all associated logical volumes are deleted. The window should look like the following:

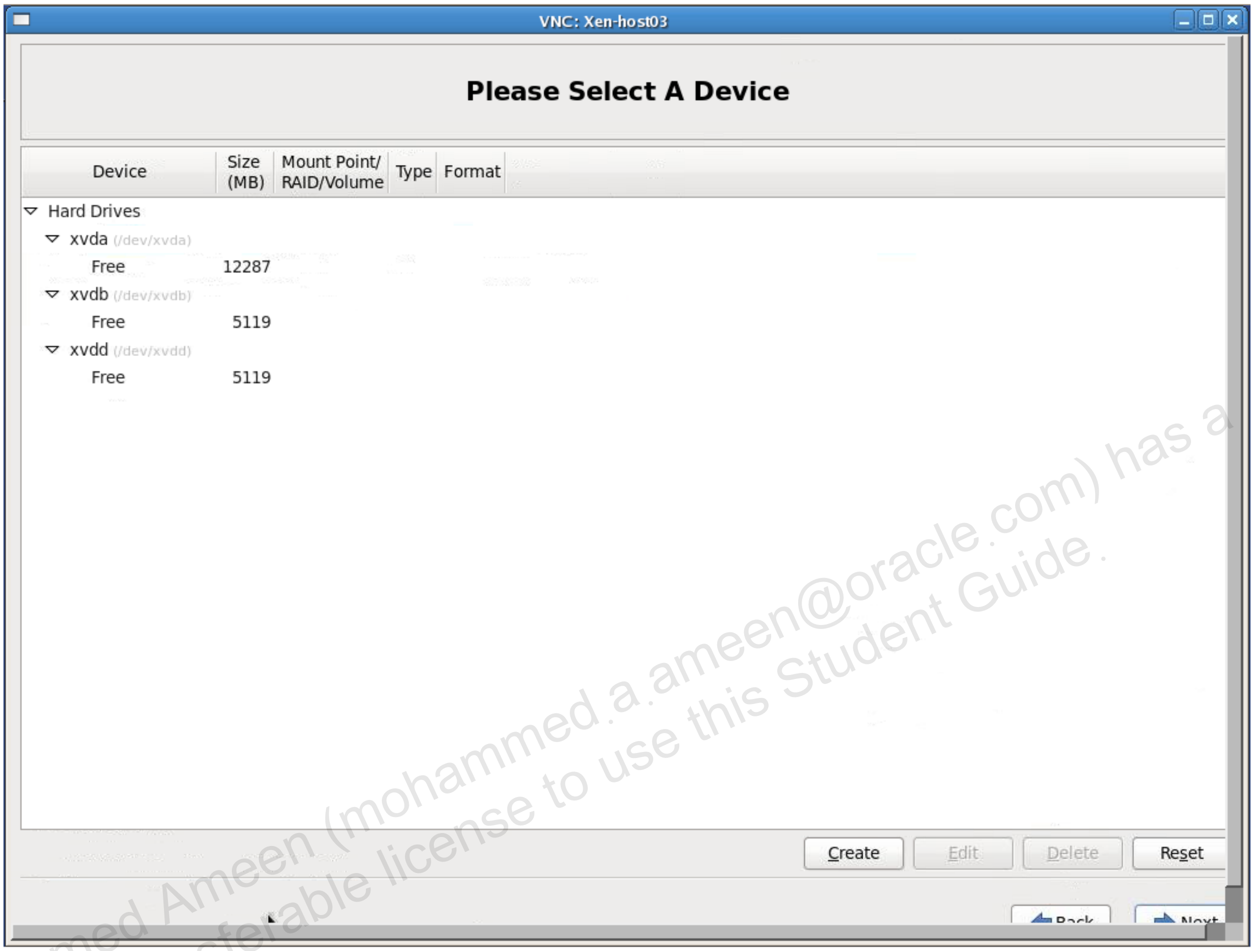


d. Select the first virtual hard drive, **xvda**, and click **Delete**.

e. A **Confirm Delete** box is again displayed. Click **Delete** to confirm.

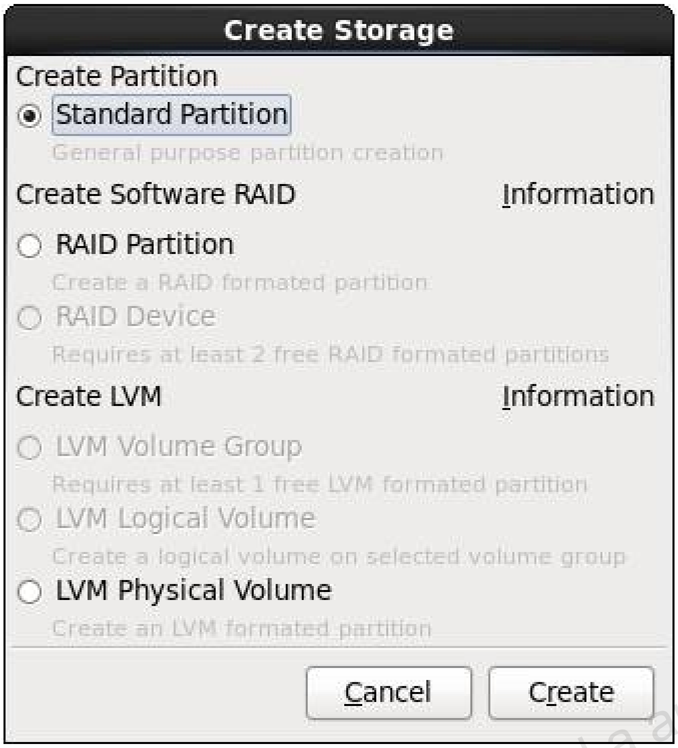
17. Create Custom Layout.

- After deleting all the default partitions, the window should look like the following:



a. Scroll down if necessary and click **Create**.

- The Create Storage dialog box appears:



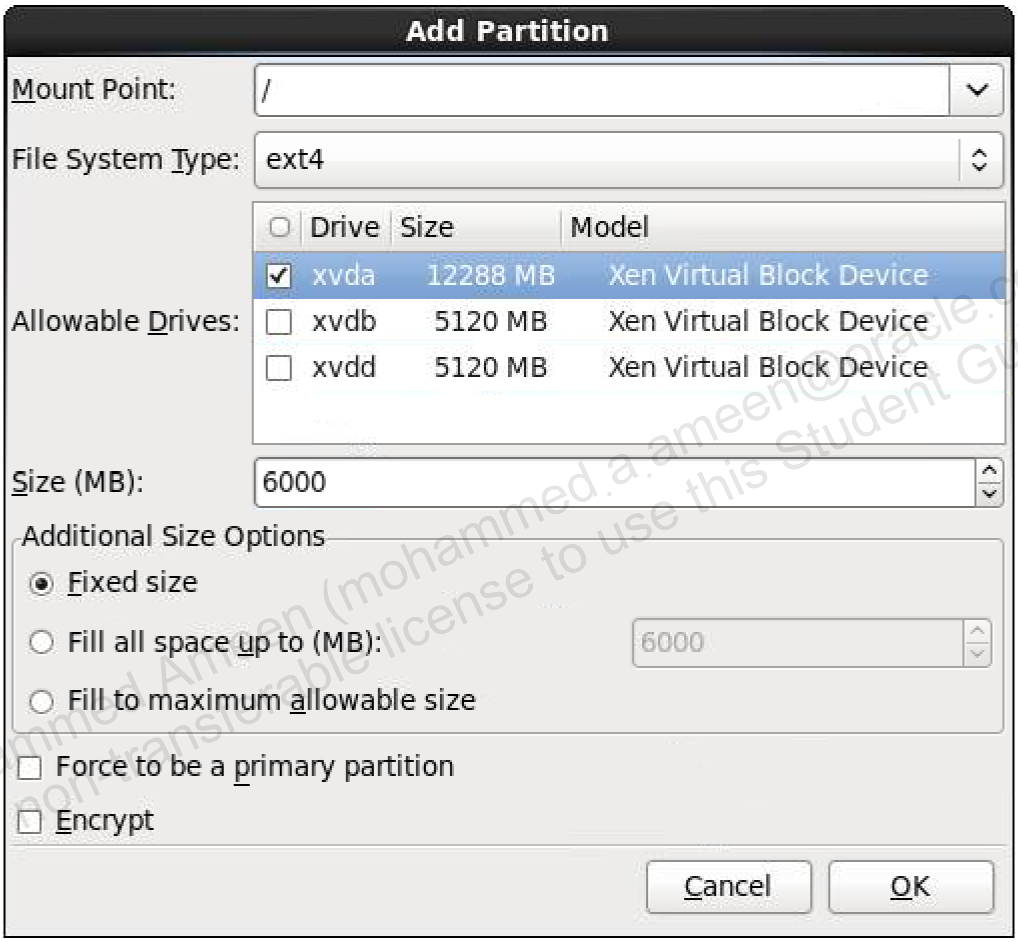
b. Select **Standard Partition** and click **Create**.

- The Add Partition box appears.

c. Add a standard partition with the following characteristics:

- Mount Point: **/**
- File System Type: **ext4**
- Allowable Drives: **xvda**
- Size (MB): **6000**
- Additional Size Options: **Fixed size**

The completed Add Partition dialog box should look like the following:

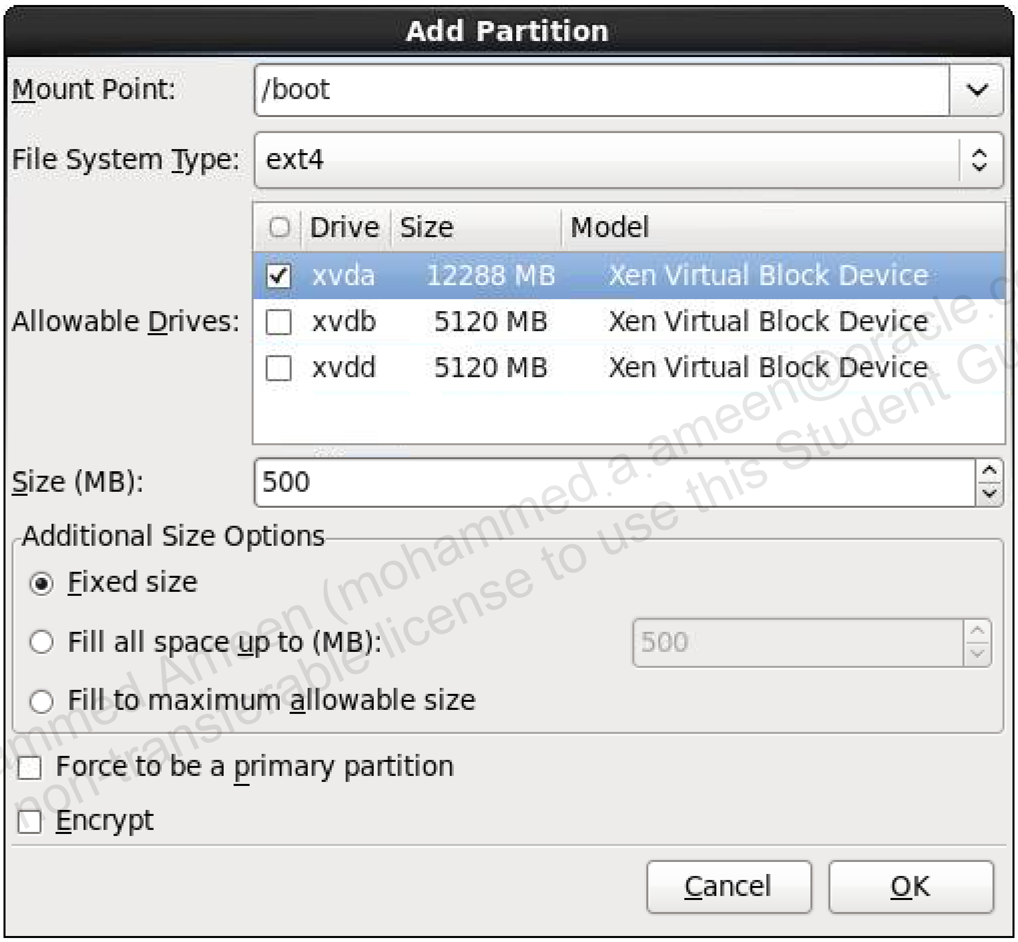


d. Click **OK**.

e. Click **Create** to add a second Standard Partition with the following characteristics:

- Mount Point: **`/boot`**
- File System Type: **ext4**
- Allowable Drives: **xvda**
- Size (MB): **500**
- Additional Size Options: **Fixed size**

The completed Add Partition dialog box should look like the following:



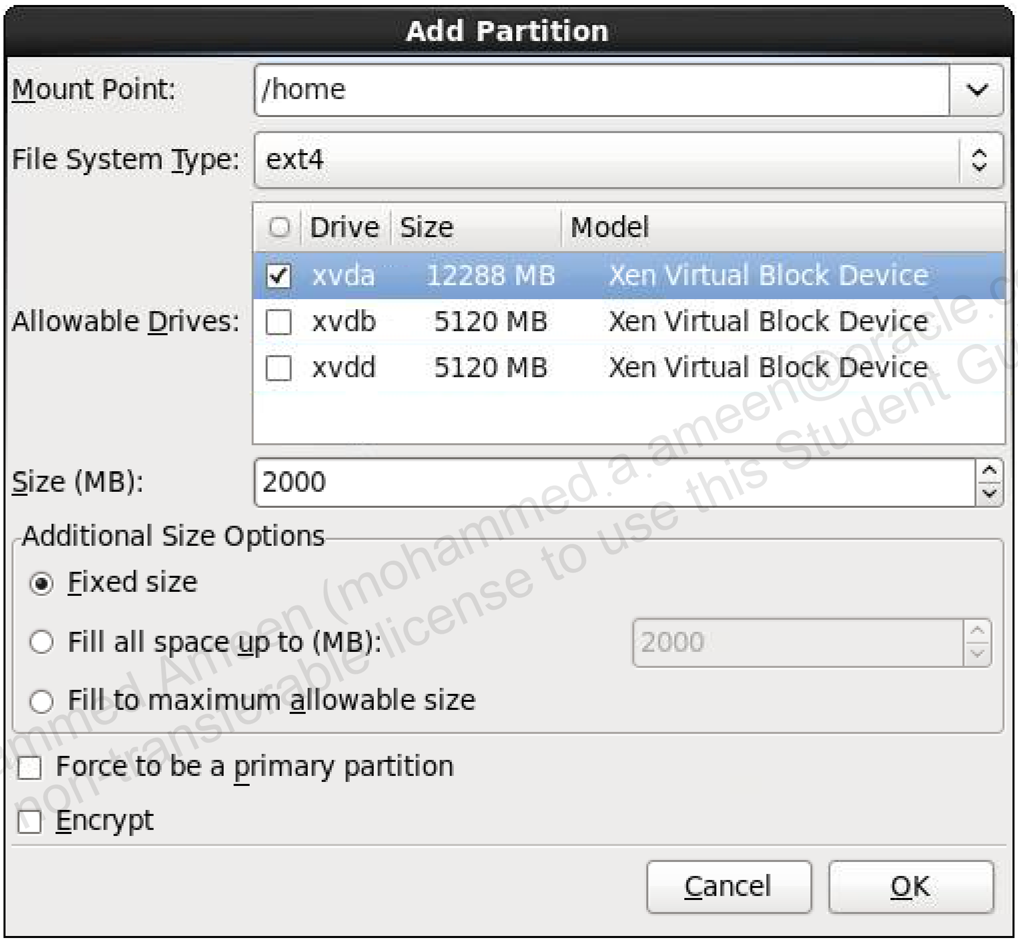
Note that the recommended size of `/boot` went from 100 MB to 500 MB between Oracle Linux 5 and Oracle Linux 6. The `initramfs` files are much larger in Oracle Linux 6 and multiple `initramfs` files can easily fill a small `/boot` partition.

f. Click **OK**.

g. Click **Create** to add a third Standard Partition with the following characteristics:

- Mount Point: **`/home`**
- File System Type: **ext4**
- Allowable Drives: **xvda**
- Size (MB): **2000**
- Additional Size Options: **Fixed size**

The completed Add Partition dialog box should look like the following:

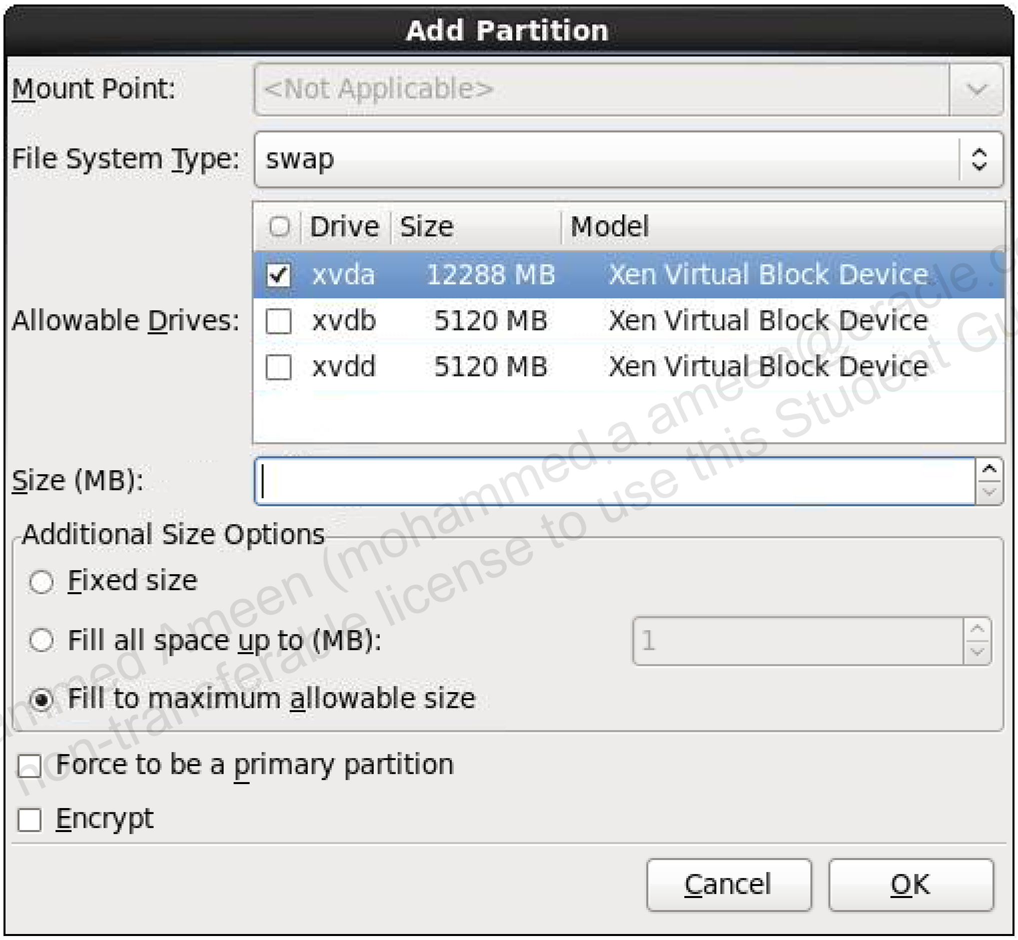


h. Click **OK**.

i. Click **Create** to add a fourth Standard (swap) Partition with the following characteristics:

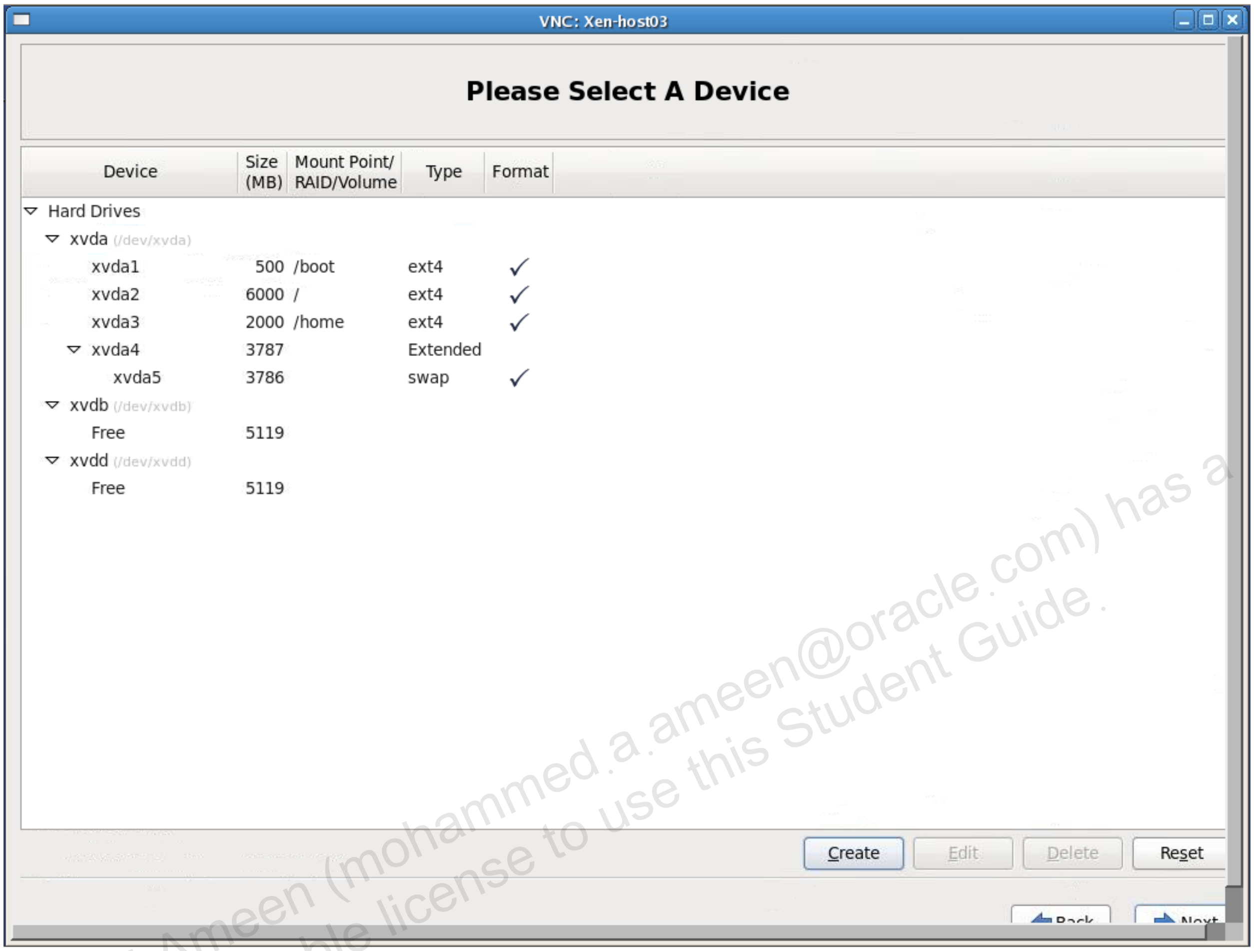
- Mount Point: **no selection**
- File System Type: **swap**
- Allowable Drives: **xvda**
- Size (MB): **no selection**
- Additional Size Options: **Fill to maximum allowable size**

The completed Add Partition dialog box should look like the following:



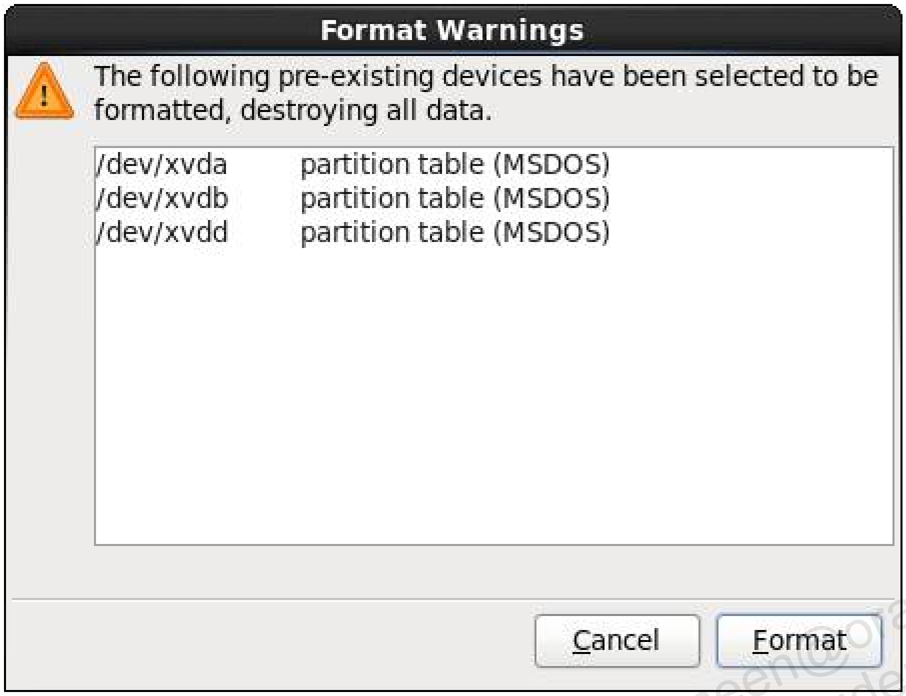
j. Click **OK**.

- Confirm that the final custom layout looks like the following:



k. Scroll down if necessary and click **Next**.

- A Format Warnings box appears.

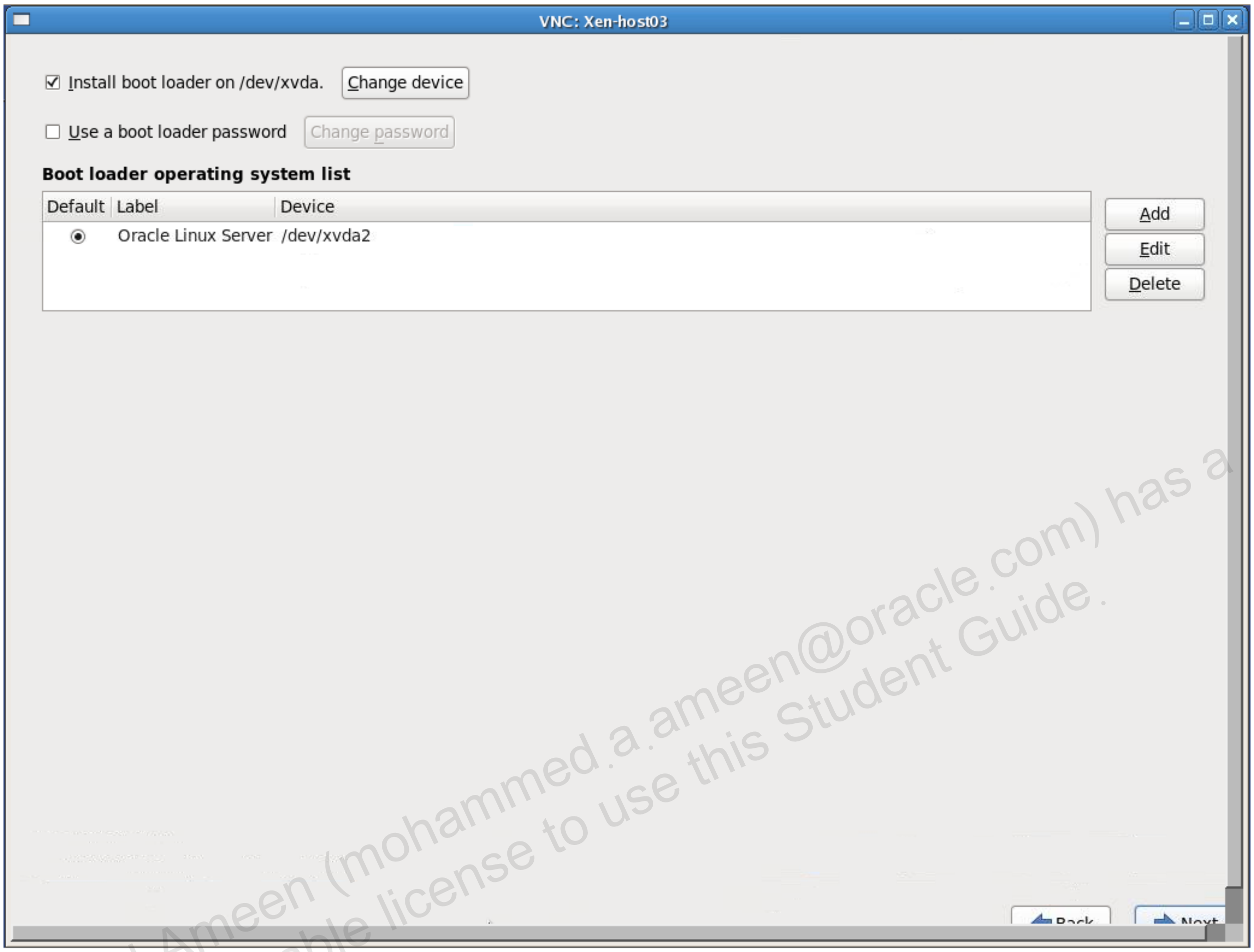


l. Click **Format**.

- A **Writing storage configuration to disk** dialog box appears.

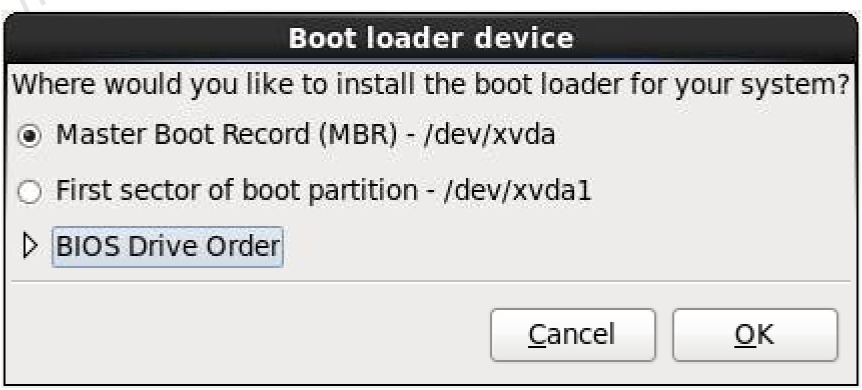
m. Click **Write changes to disk**.

18. The Boot Loader Selection window appears.



a. Ensure that the **Install boot loader on /dev/xvda** check box is selected (this is the default).

b. Click **Change device** to display the Boot loader device dialog box:

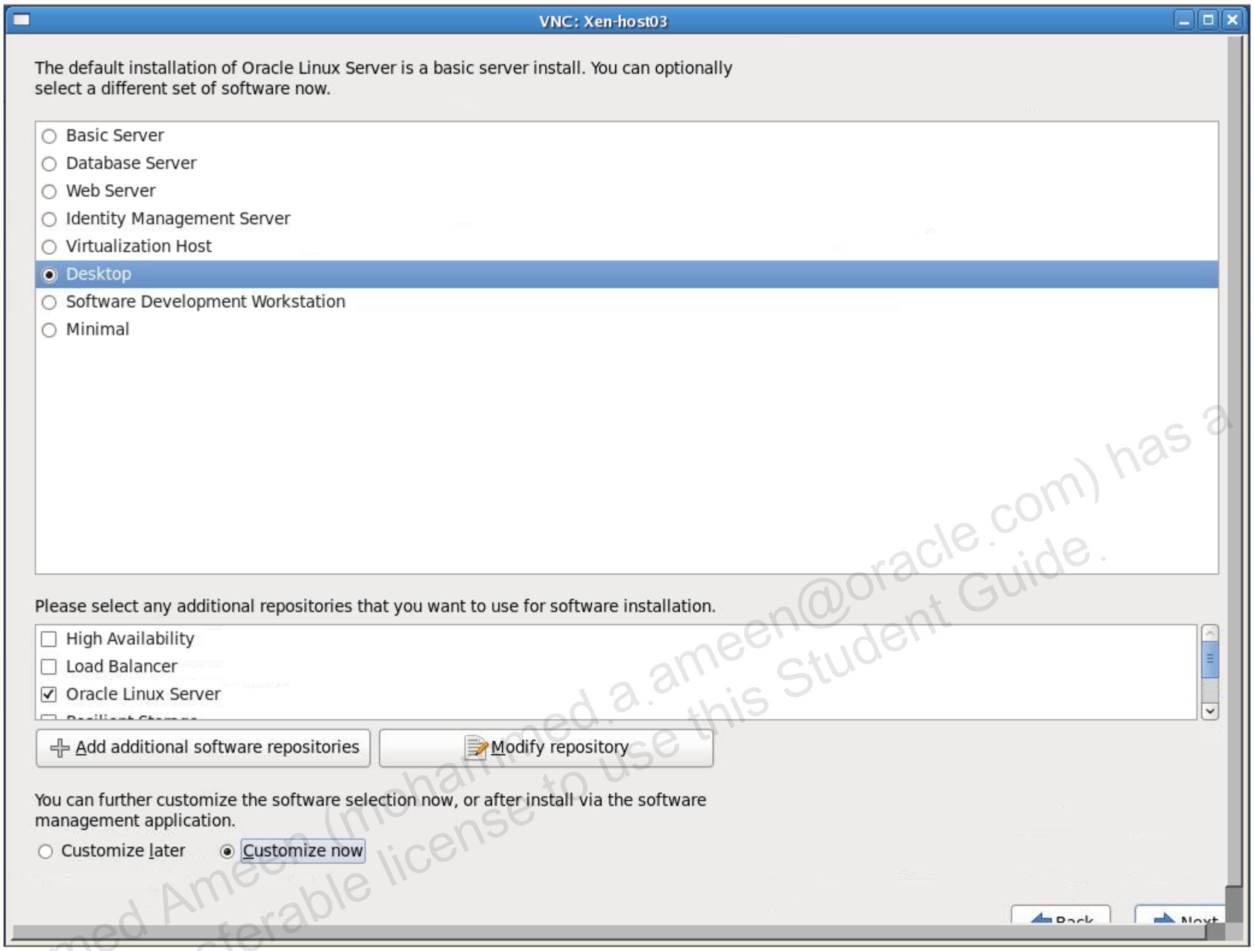


c. Do not make any changes. Click **Cancel** to install the boot loader on the MBR.

d. Scroll down if necessary and click **Next**.

19. Select software to install.

- The Software Package Selection window appears:



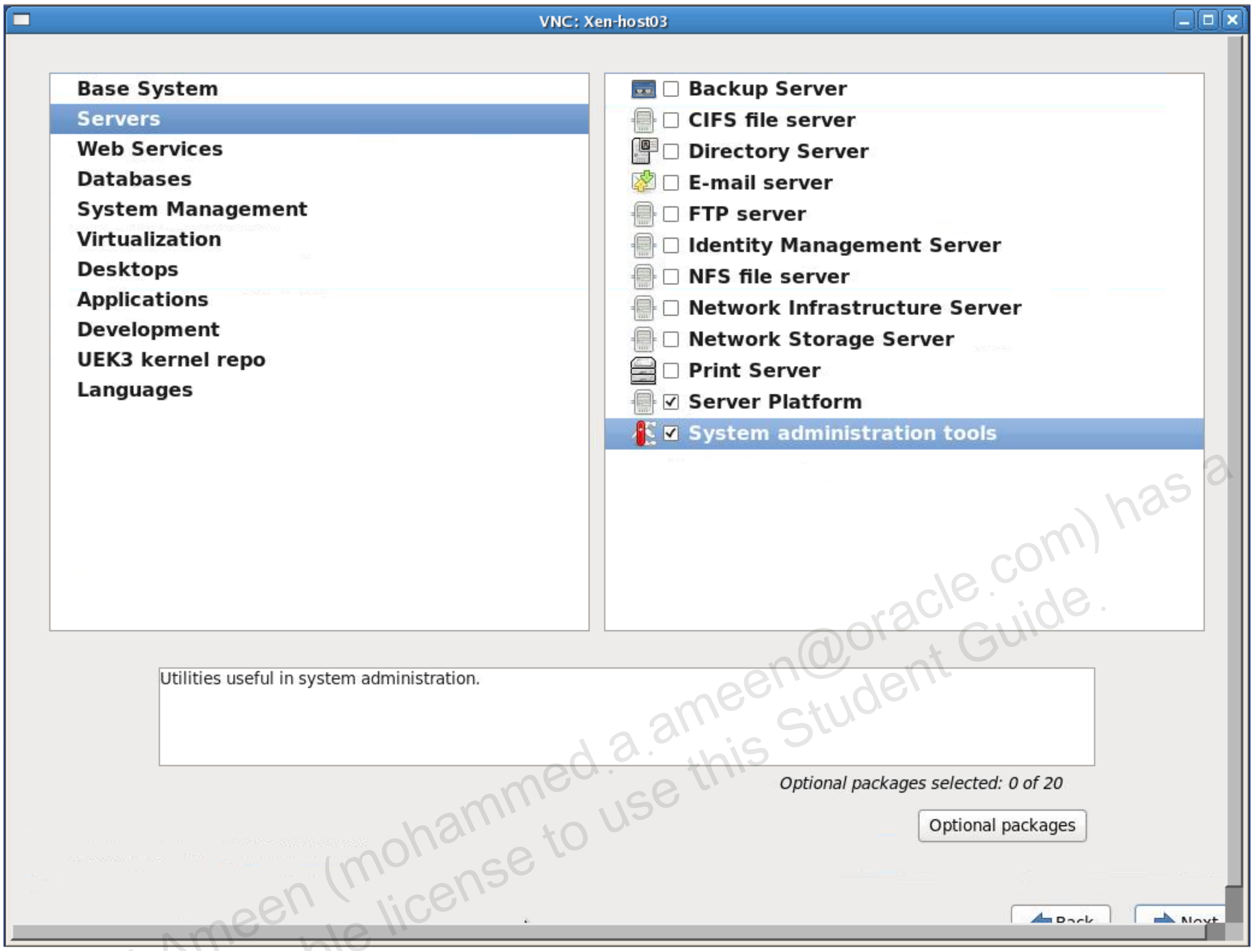
a. Select **Desktop** from the list of software sets.

b. Scroll down if necessary and click **Customize Now**.

c. Accept all the other default selections.

d. Click **Next**.

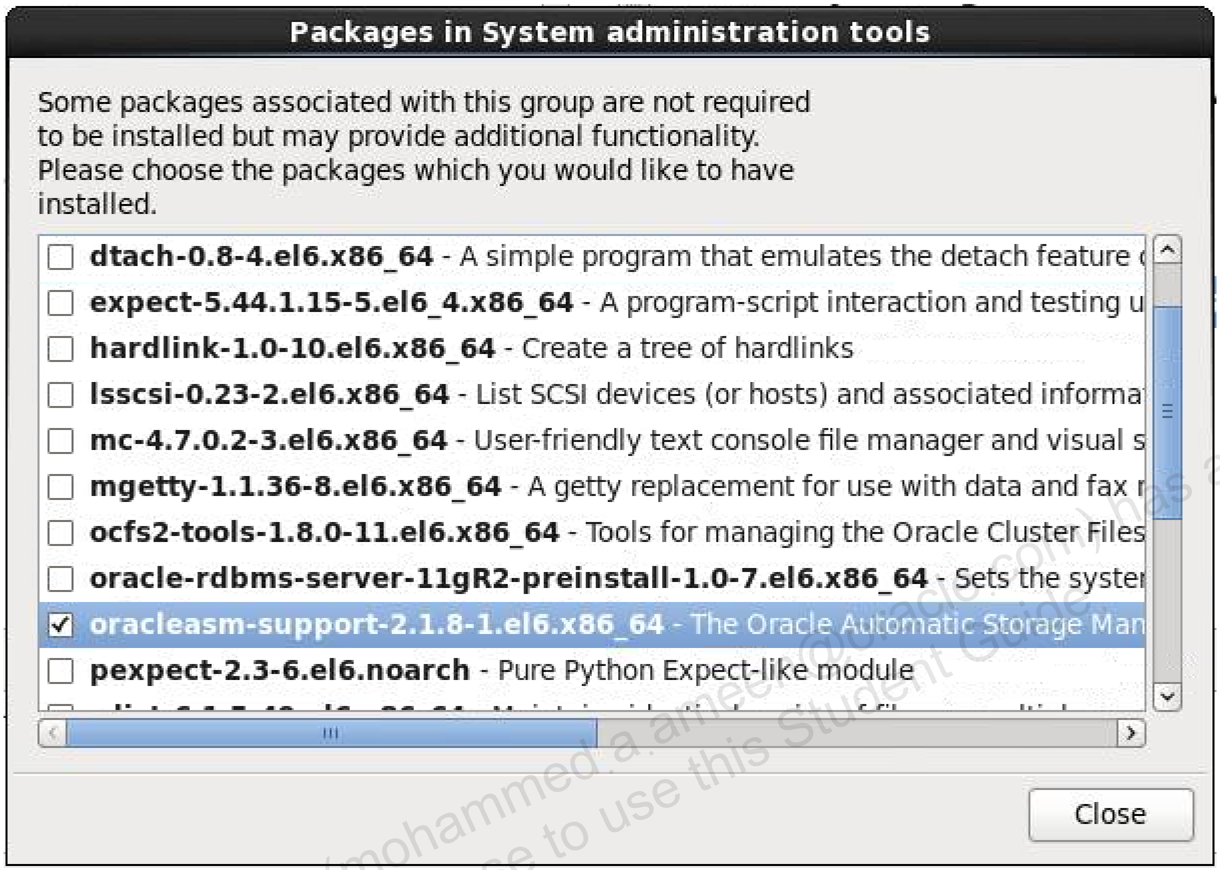
- The Customize Package Selection window appears.



e. Select **Servers** from the left pane, and then select **System administration tools** from the right pane (as shown in the screenshot).

f. Scroll down if necessary and click **Optional packages**.

- The Packages in System administration tools screen appears as follows:



g. Scroll down if necessary and select the **oracleasm-support** package (as shown):

h. Click **Close**.

i. Click **Next**.
   - The Software Install screen is displayed, showing the progress as the selected software packages are installed.

- After the installation is complete, the following window appears:

j. Click **Reboot**.
  - After you reboot your system, your vnc session closes.

Practice 3-2: Using FirstBoot

Overview

In this practice, you complete FirstBoot.

Assumptions

- You have completed the installation of Oracle Linux.
- You have rebooted your system.

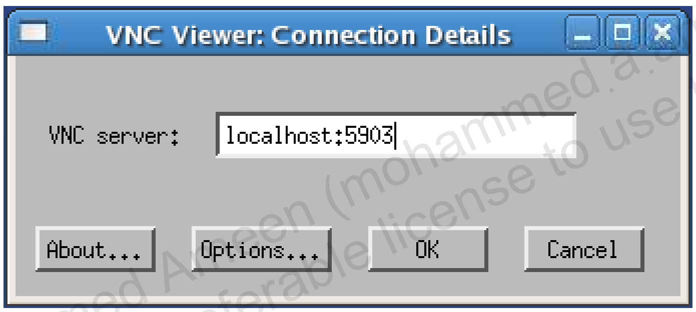
Tasks

1. Connect to the **host03** guest by using **vncviewer**.
   a. Run the `vncviewer&` command.

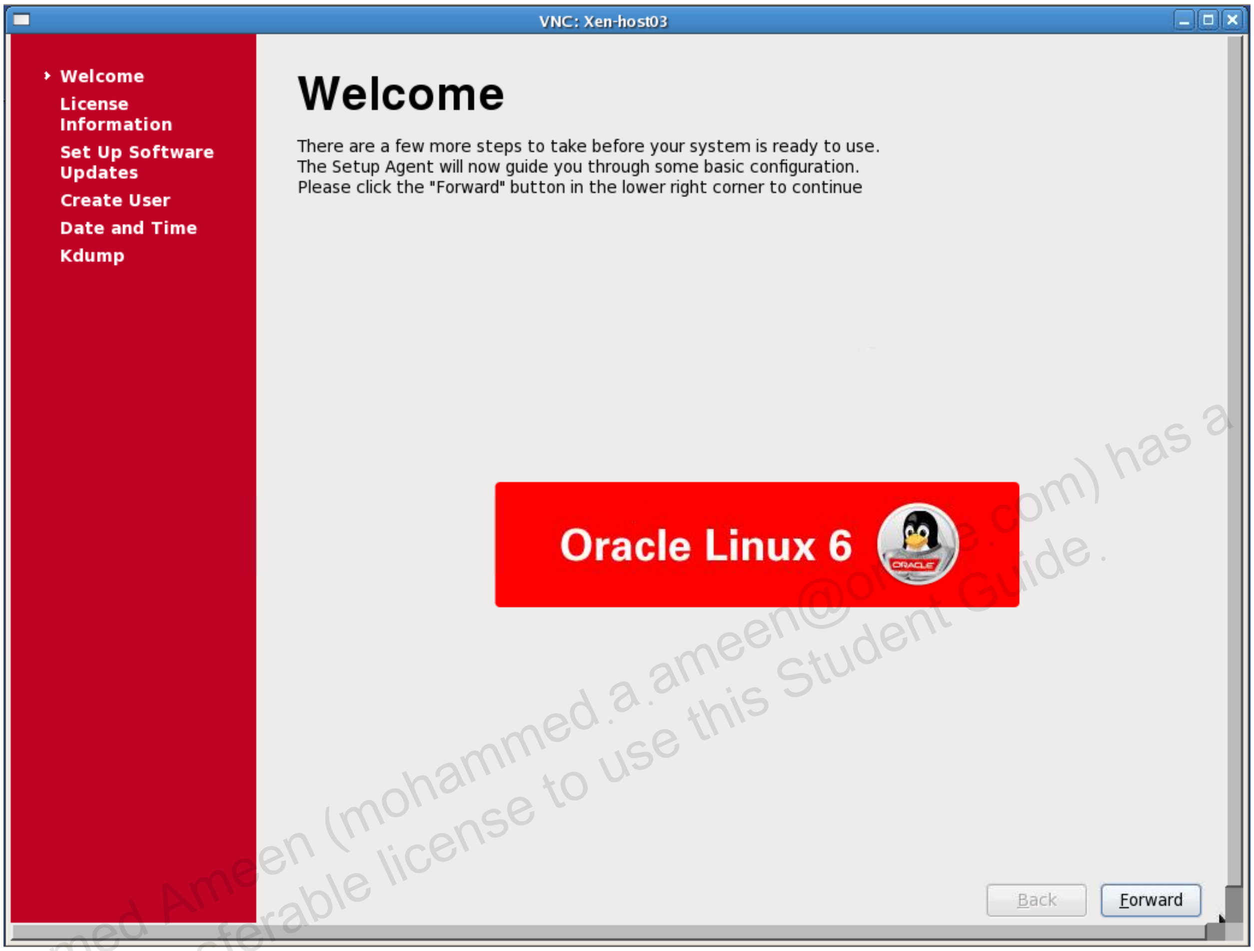
   ```
   # vncviewer&
   ```

   - The VNC Viewer: Connection Details dialog box is displayed.

   b. Enter `localhost:<port_number>`, substituting the correct port number for the **host03** guest. For example, if the port number is `5903`, enter `localhost:5903` and click **OK** (as shown in the following screenshot):

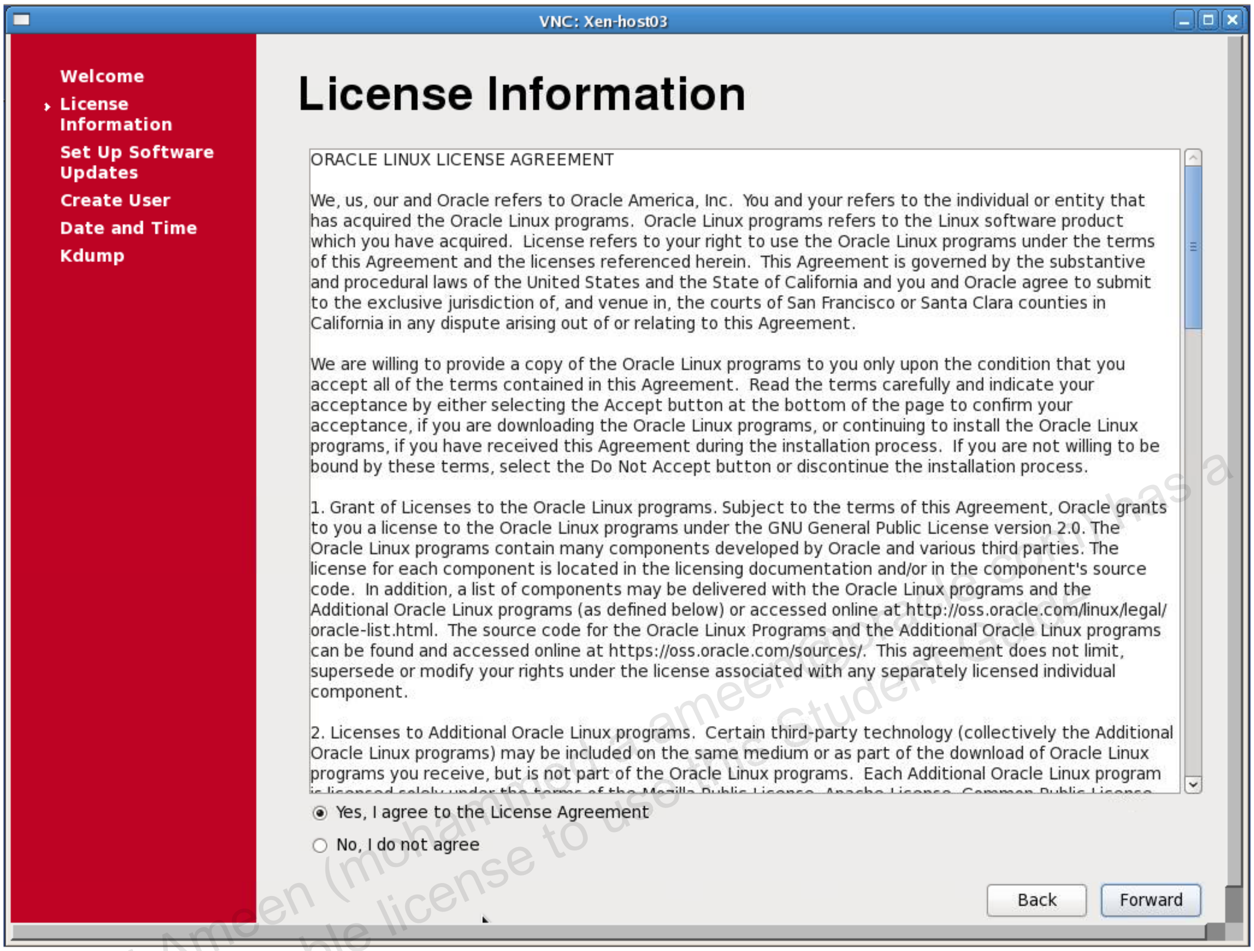


2. Complete FirstBoot.
    - The **Welcome** window appears.



    a. Scroll down if necessary and click **Forward**.

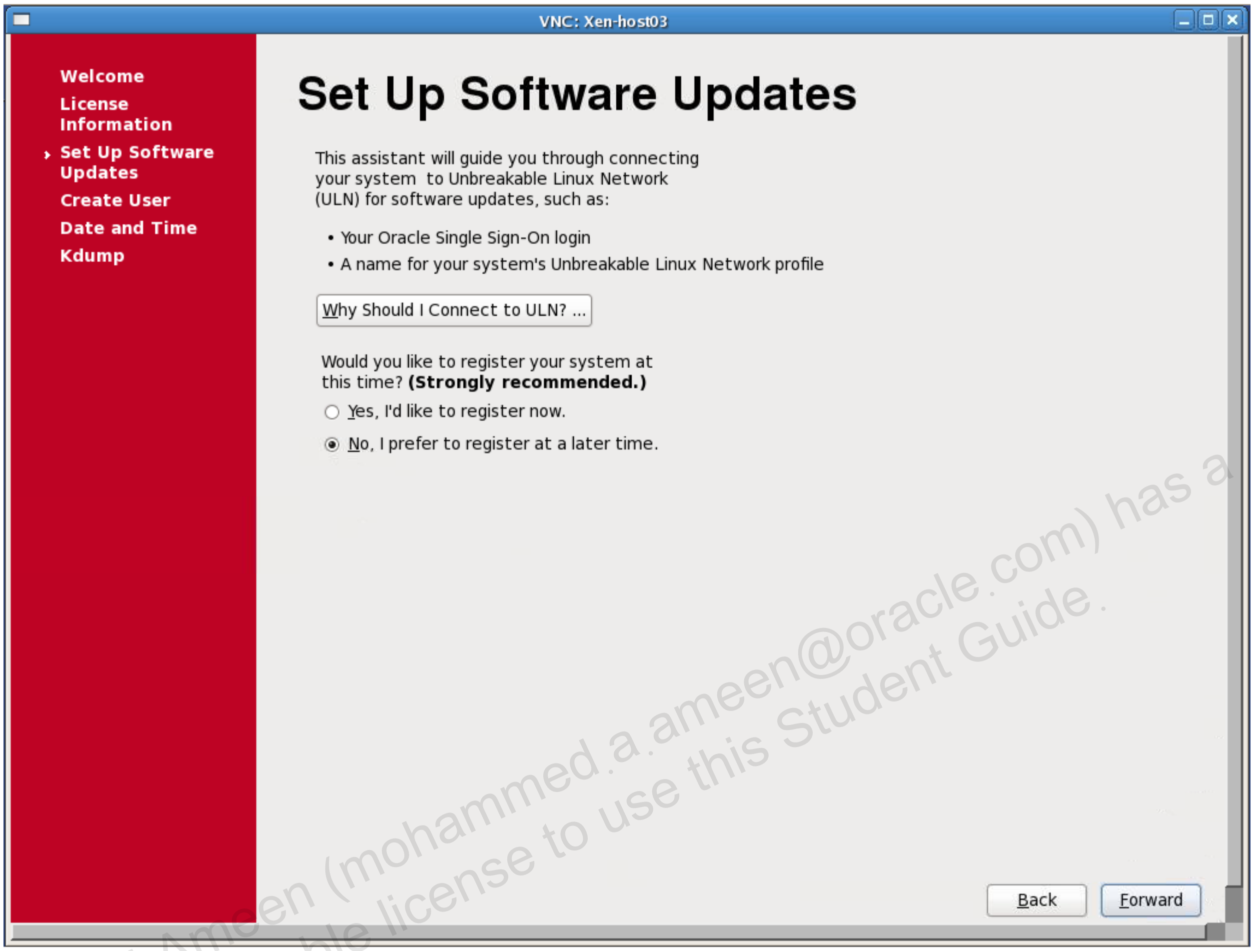
- The **License Information** window appears.



b. Scroll down if necessary and click **Yes, I agree to the License Agreement**.

c. Click **Forward**.

- The **Set Up Software Updates** window appears.



- The Unbreakable Linux Network (ULN) is discussed in Lesson 6, "Package Management."

d. Select **No, I prefer to register at a later time**.

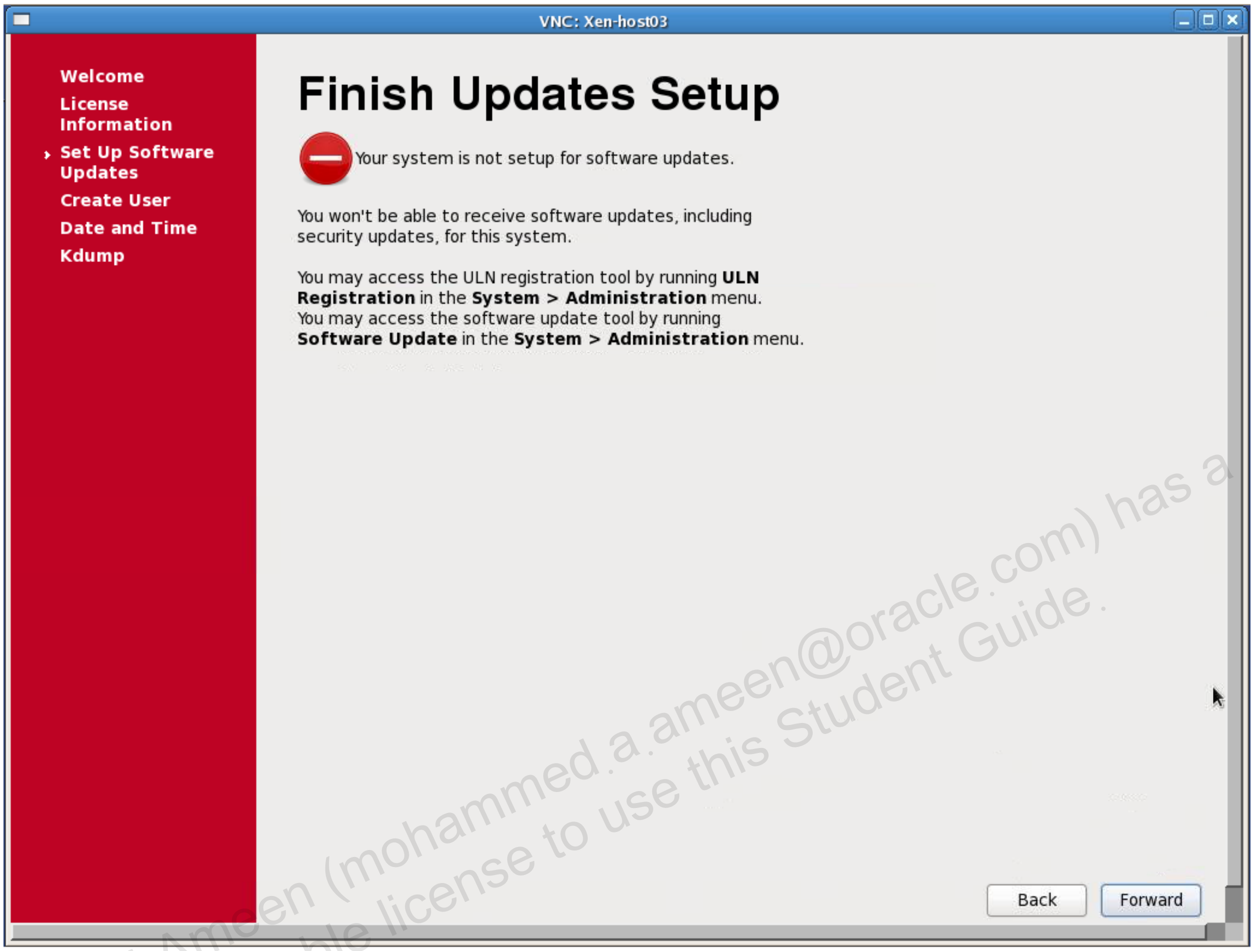
e. Scroll down if necessary and click **Forward**.

- A FirstBoot window displays reasons you should connect to ULN.

f. Click **No thanks, I'll connect later**.

g. Scroll down if necessary and click **Forward**.

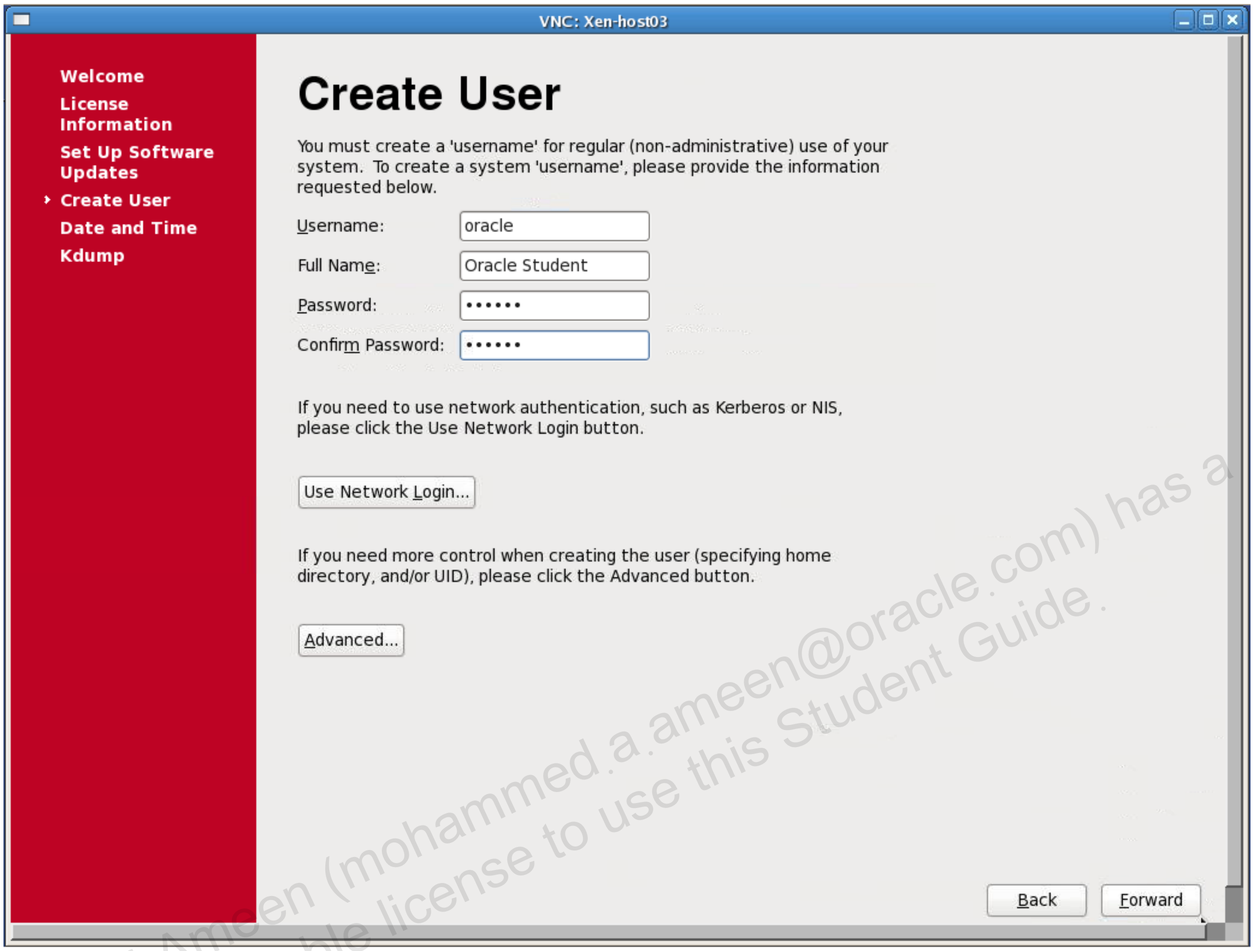
- The **Finish Updates Setup** window appears.



- You are notified that your system is not set up for software updates.

h. Scroll down if necessary and click **Forward**.

- The Create User window appears.



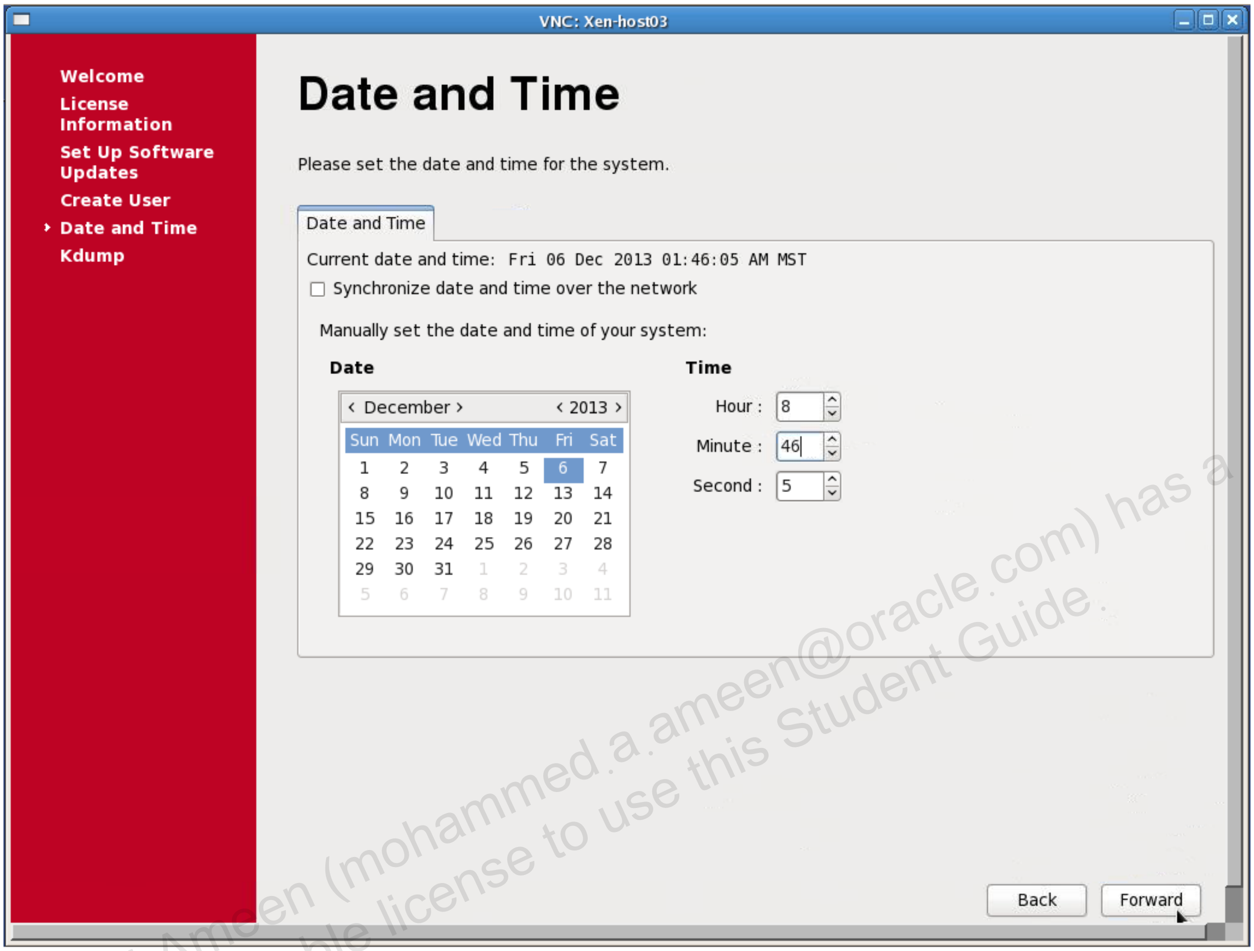
i. FirstBoot requires you to create an initial user to log in to your system

j. Create a user with the following attributes:
   - Username: `oracle`
   - Full Name: `Oracle Student`
   - Password: `oracle`
   - Confirm Password: `oracle`

k. Scroll down if necessary and click **Forward**.

l. Click **Yes** to use the chosen password.

- The **Date and Time** window appears.



m. Set the current date and time.

n. Scroll down if necessary and click **Forward**.

- Kdump is the last component to configure with FirstBoot.
- However, do not enable kdump at this time.

o. Scroll down if necessary and click **Finish**.

The GNOME desktop login window appears after completing FirstBoot.

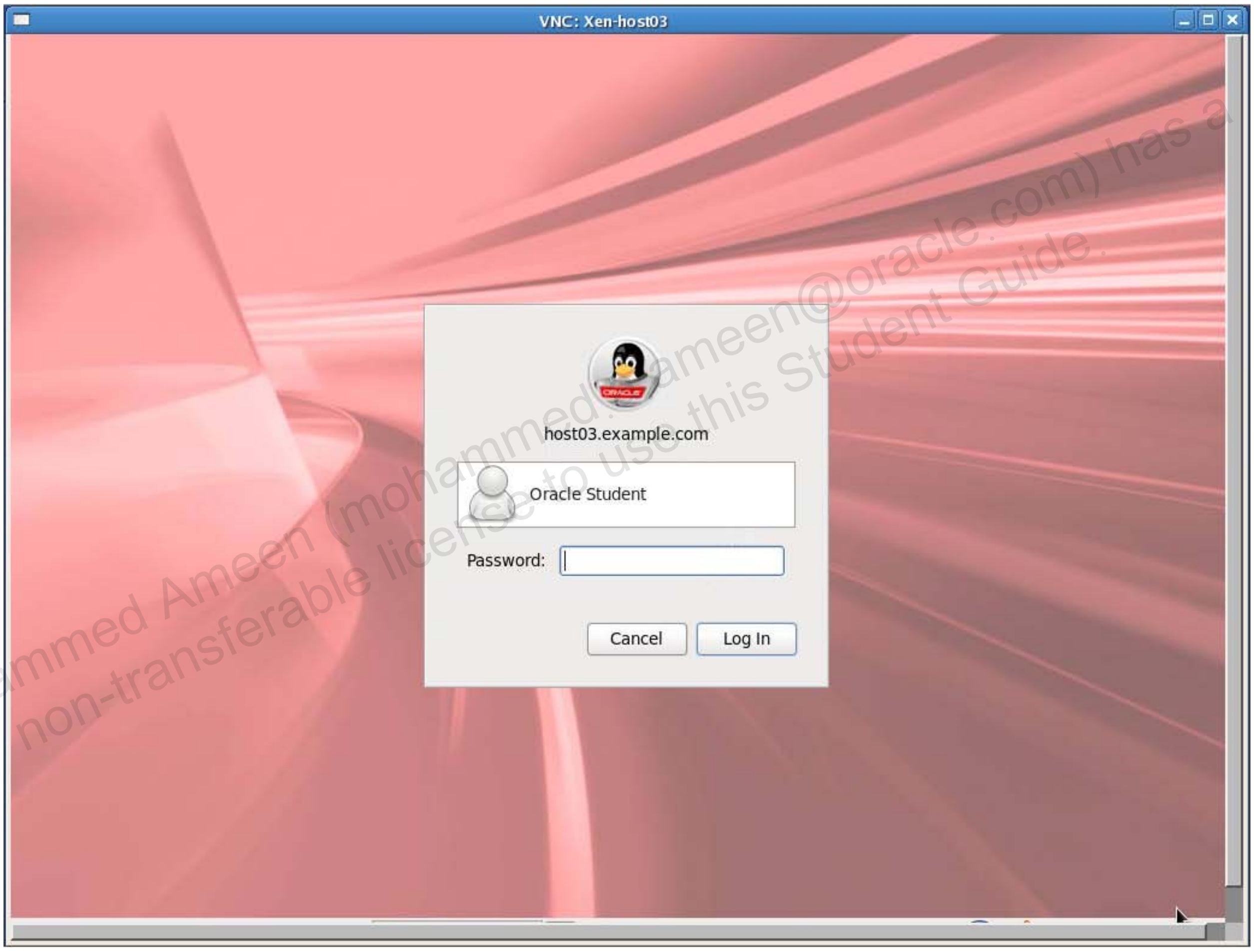
Practice 3-3: Logging In to Oracle Linux and Shutting Down

Overview

In this practice, you log in as the `oracle` user. Then you shut down the **host03** VM.

Tasks

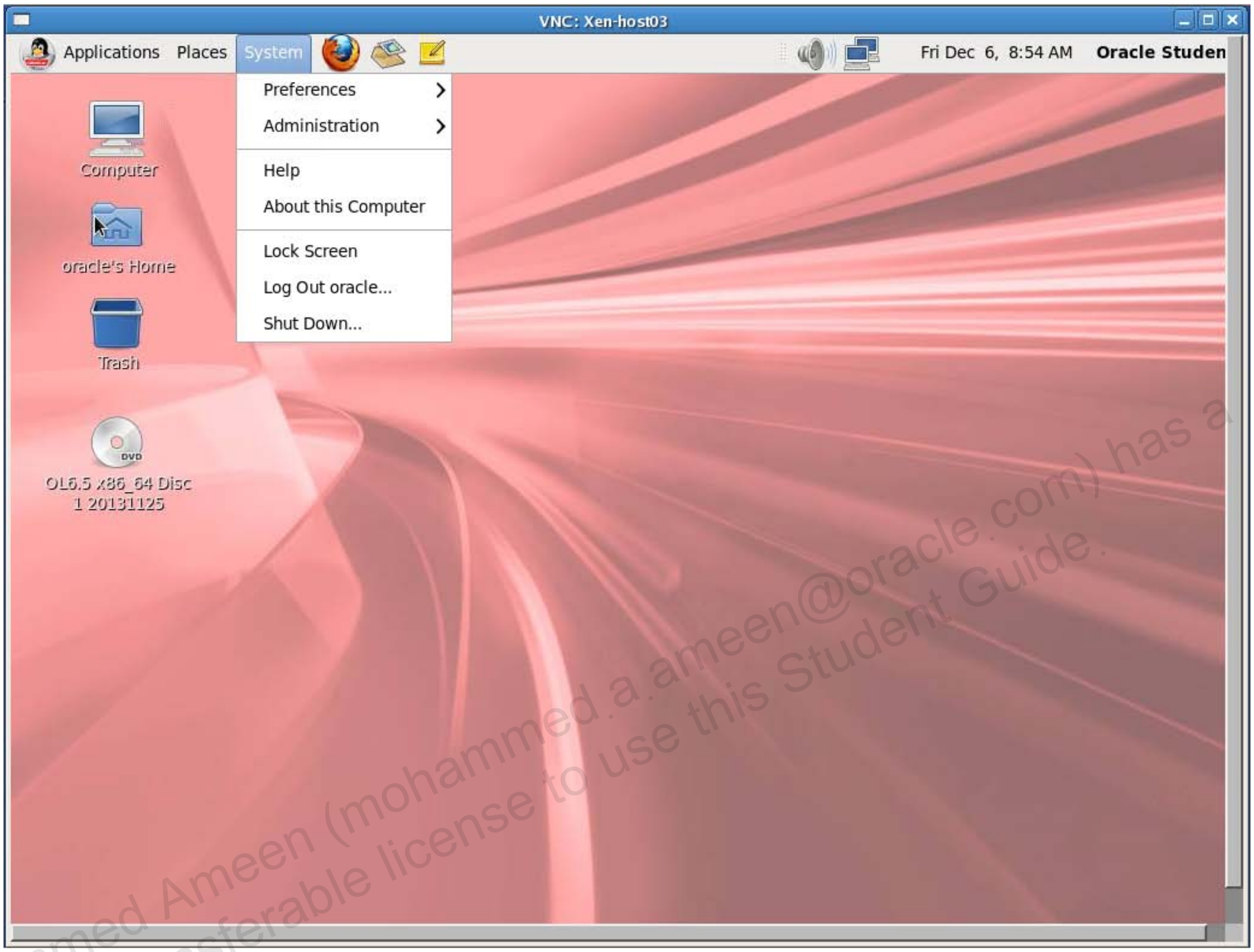
1. Log in to Oracle Linux from the GNOME desktop.
   a. Select `Oracle Student`.
      - The following window appears:



   b. Enter the password `oracle` for Oracle Student.
   c. Click **Log In**.
      - The GNOME desktop appears.

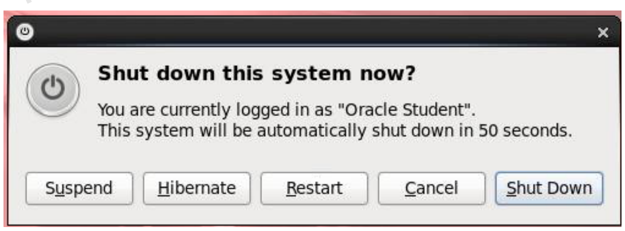
2. Shut down Oracle Linux.

a. Open the **System** menu:



b. Select **Shut Down** from the System menu.

- The following dialog box appears:



c. Click **Shut Down**.

- The VM **host03** is shut down and the vnc session closes.

Practice 3-4: Re-Creating the host03 VM Guest

Overview

In this practice, you re-create the **host03** VM.

Task

Re-create the **host03** VM:

a. From **dom0**, run the `xm list` command as follows to verify **host03** is not running.

```
# xm list
Name                    ID    Mem  VCPUs      State      Time(s)
Domain-0                 0   1024      2     r-----        758.9
host01                   4   2048      1     -b----         37.4
host02                   5   2048      1     -b----         37.3
```

- Only **dom0** (Domain-0) and two guests (**host01** and **host02**) are running.

b. Use the `cd` command to change to the `/OVS/running_pool/host03` directory:

```
# cd /OVS/running_pool/host03
```

c. Use the `xm create` command as follows:

```
# xm create vm.cfg
Using config file "./vm.cfg".
Started domain host03 (id=8)
```

d. Confirm that **host03** VM is running by using the `xm list` command as follows:

```
# xm list
Name                    ID    Mem  VCPUs      State      Time(s)
Domain-0                 0   1024      2     r-----        774.6
host01                   4   2048      1     -b----         37.5
host02                   5   2048      1     -b----         37.4
host03                   8   2048      1     -b----          3.3
```

- This output shows **host03** is now running.

Practices for Lesson 4: Linux Boot Process

Chapter 4

Practices for Lesson 4: Linux Boot Process

Practices Overview

In these practices, you perform the following tasks:

- Explore the GRUB boot loader.
- Edit the GRUB configuration file to boot different kernels.
- Use the GRUB menu to modify kernel boot parameters.
- Change the default run level.
- Explore the `init` script directories.
- Configure a service to start and stop at a given run level.
- Explore and configuring Upstart jobs.

In Lesson 14 titled “File Sharing,” you configure `xinetd` to start `vsftpd` on demand.

Practice 4-1: Exploring the GRUB Boot Loader

Overview

In this practice, you relate boot loader options selected during the installation of Oracle Linux to actual disk partitions and directories. You explore the `/boot` directory, the GRUB configuration file, and kernel boot parameters. The displayed sample output may not exactly match what you see on your system. In some cases, only partial output is shown.

Assumptions

- You are logged on to **dom0**.
- You have a terminal window open.
- You are the `root` user.
- You completed Practice 3 (OS install and FirstBoot configuration).
- VM **host03** is running.

If you were unable to complete the OS install and FirstBoot configuration on **host03** in Practice 3, substitute **host02** for **host03** in this practice and all future practices in this course.

Tasks

1. Connect to **host03** guest using **vncviewer**.

   a. Run the `xm list -l host03 | grep location` command to determine the vnc port number for **host03**.

```
# xm list -l host03 | grep location
                    (location 0.0.0.0:5903)
                    (location 3)
```

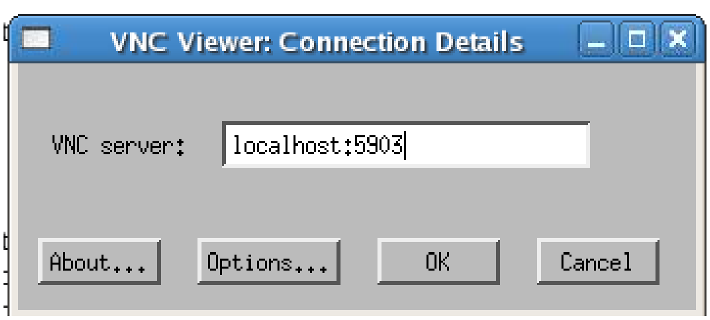
   - The sample shown indicates that the port number is **5903**. This may not be true in your case.

   b. Run the `vncviewer&` command.

```
# vncviewer&
```

   - The VNC Viewer: Connection Details dialog box is displayed.

   c. Enter `localhost:<port_number>`, substituting the port number displayed from the previous `xm list` command. For example, if the port number is `5903`, enter `localhost:5903`. Then click **OK**.

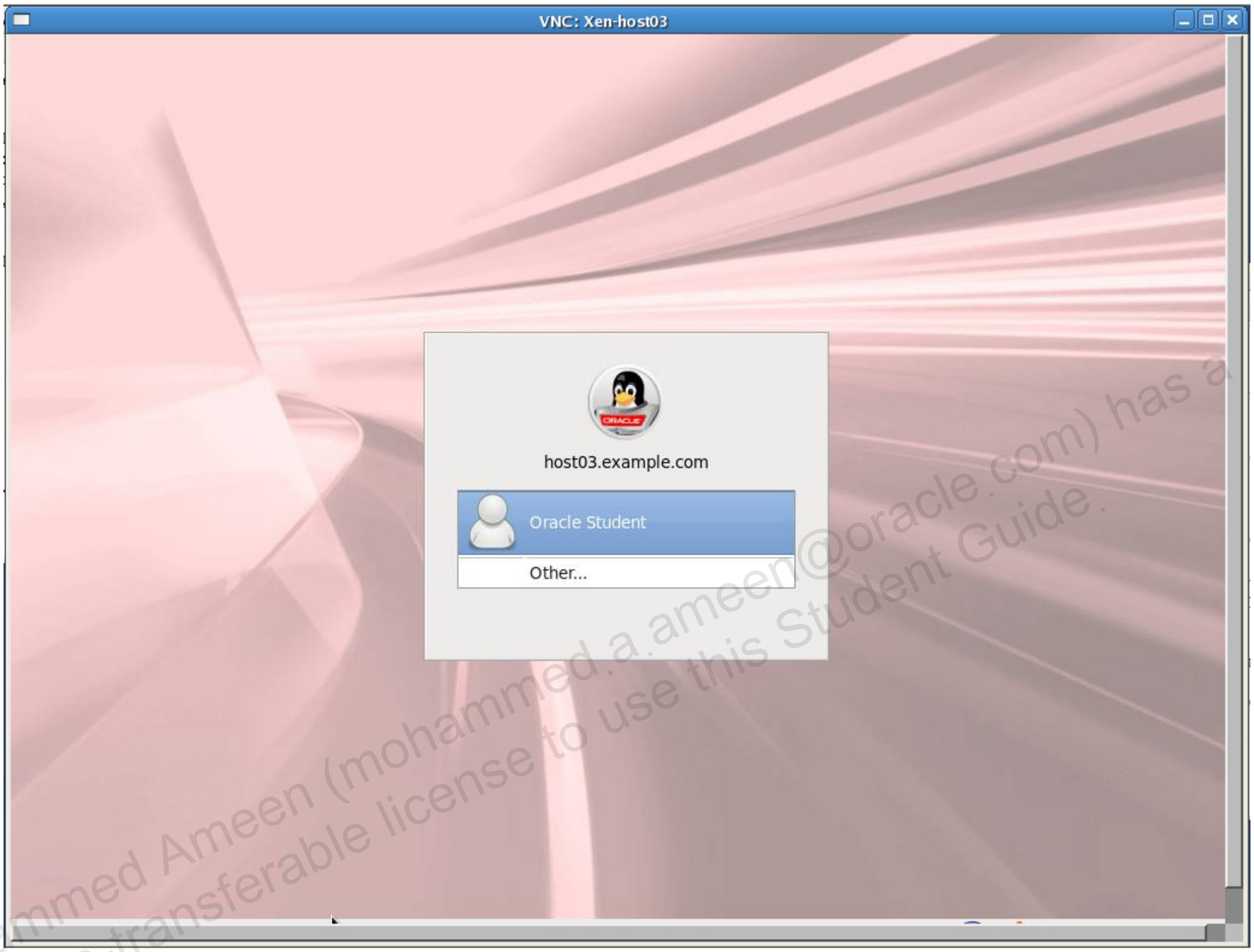


   - After you connect, the GNOME login window appears.

2. Log in from the GNOME desktop.

   a. Select **Oracle Student**.

      - The following window appears:



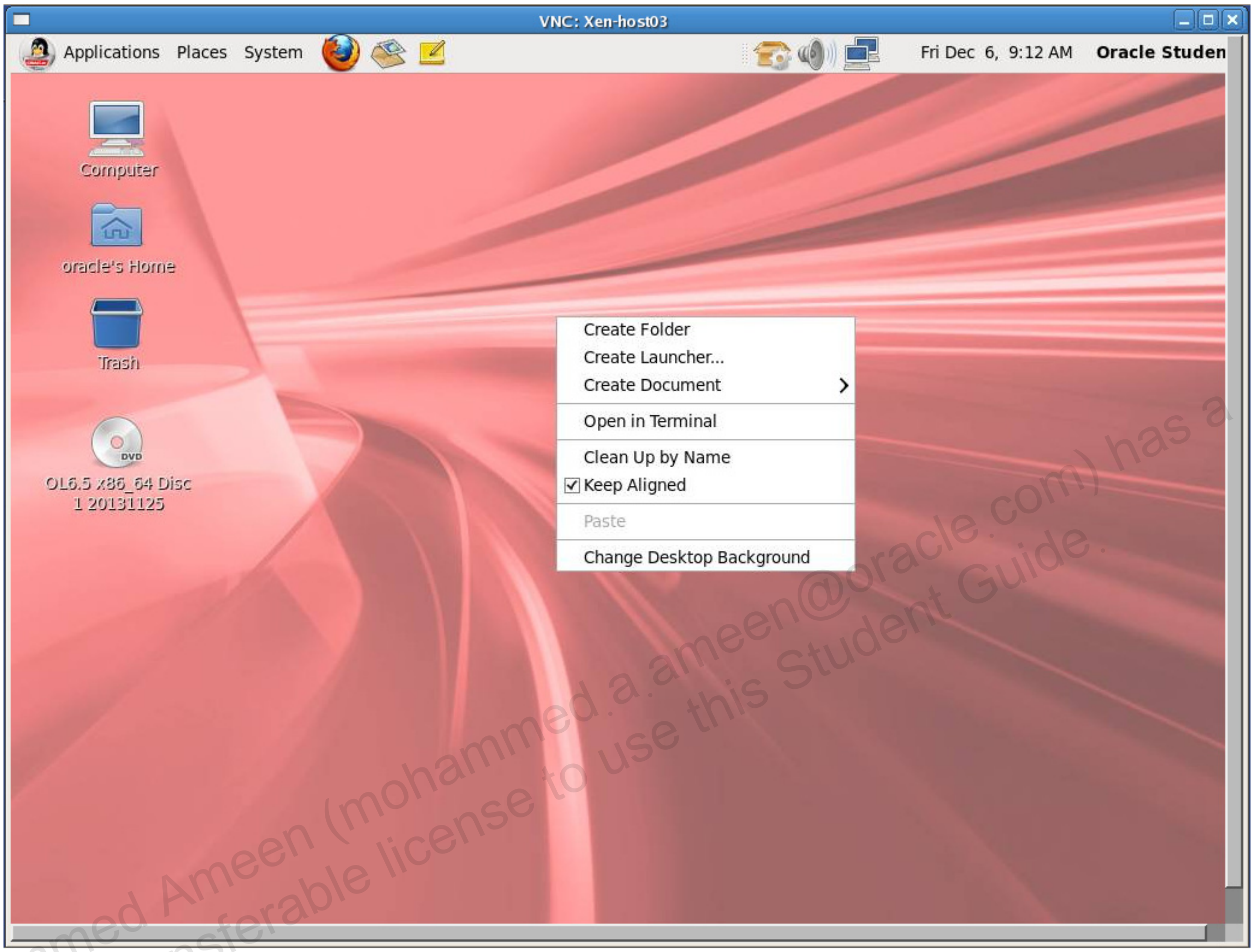
   b. Enter the password `oracle` for `Oracle Student`.

   c. Click **Log In**.

      - After you log in, the GNOME desktop appears.

3. Open a terminal window and become the `root` user.

   a. Right-click the desktop to display the pop-up menu:



   b. Click **Open in Terminal** to display a terminal window.

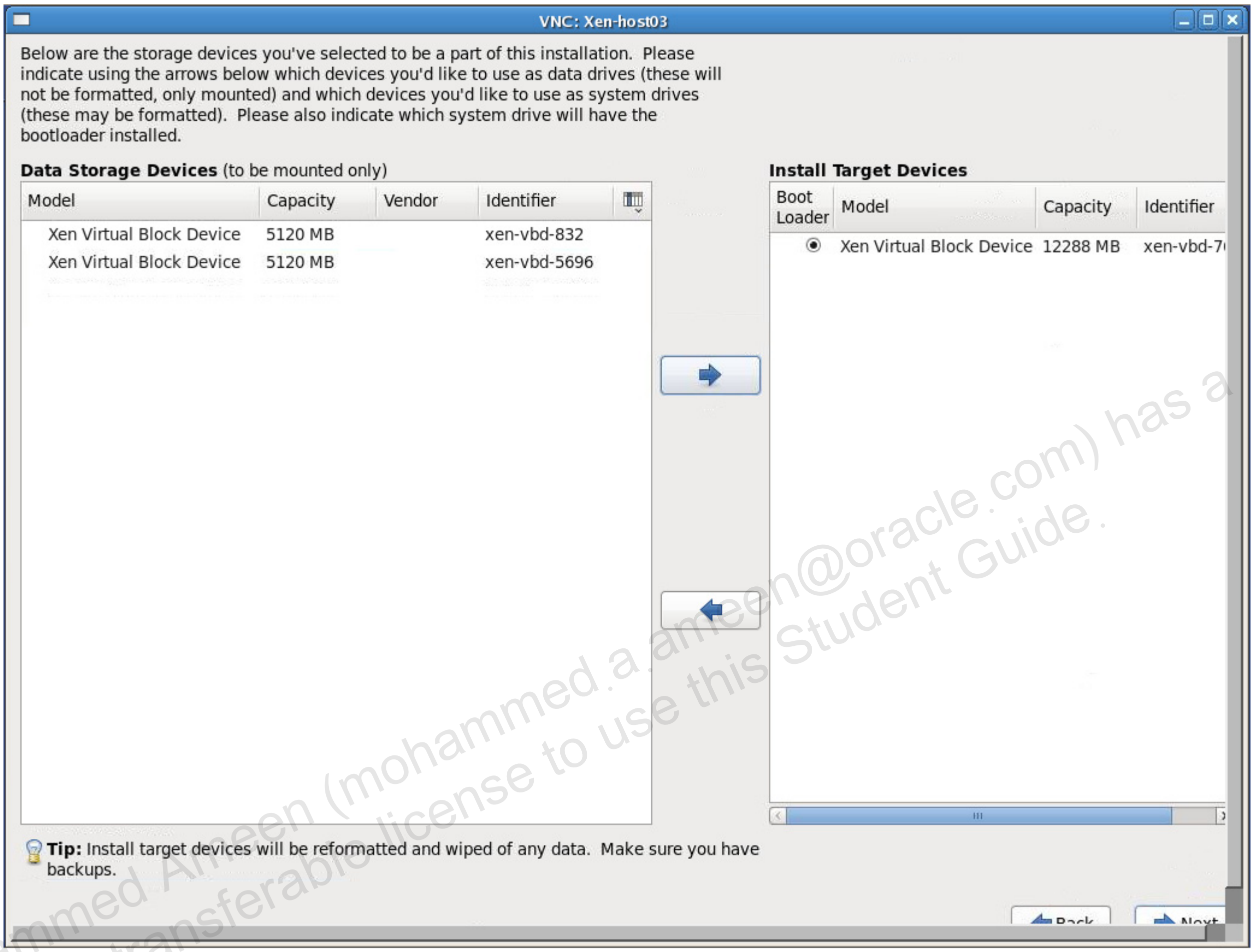
   c. In the terminal window, enter the `su -` command. Then enter the `root` password `oracle` to become `root`.

      - The password is displayed in the following example but does not appear on your screen.

```
$ su -
Password: oracle
# whoami
root
```

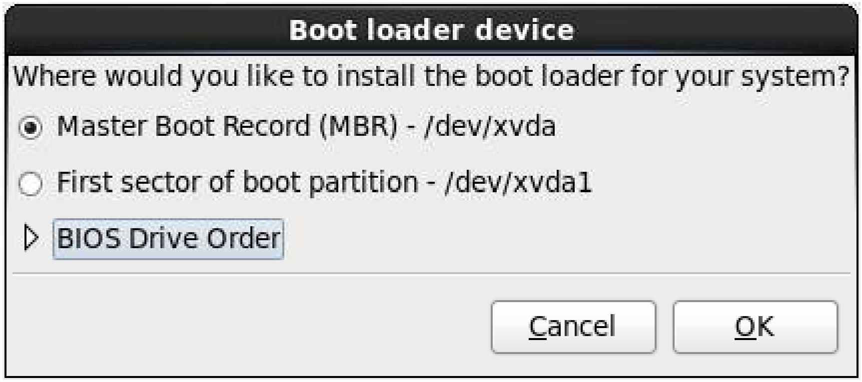
      - The `whoami` command confirms that you are the `root` user.

4. The following screens serve as a reminder of the partition layout that you created when installing Oracle Linux.
   - The Storage Devices that you selected during the installation process are shown here:



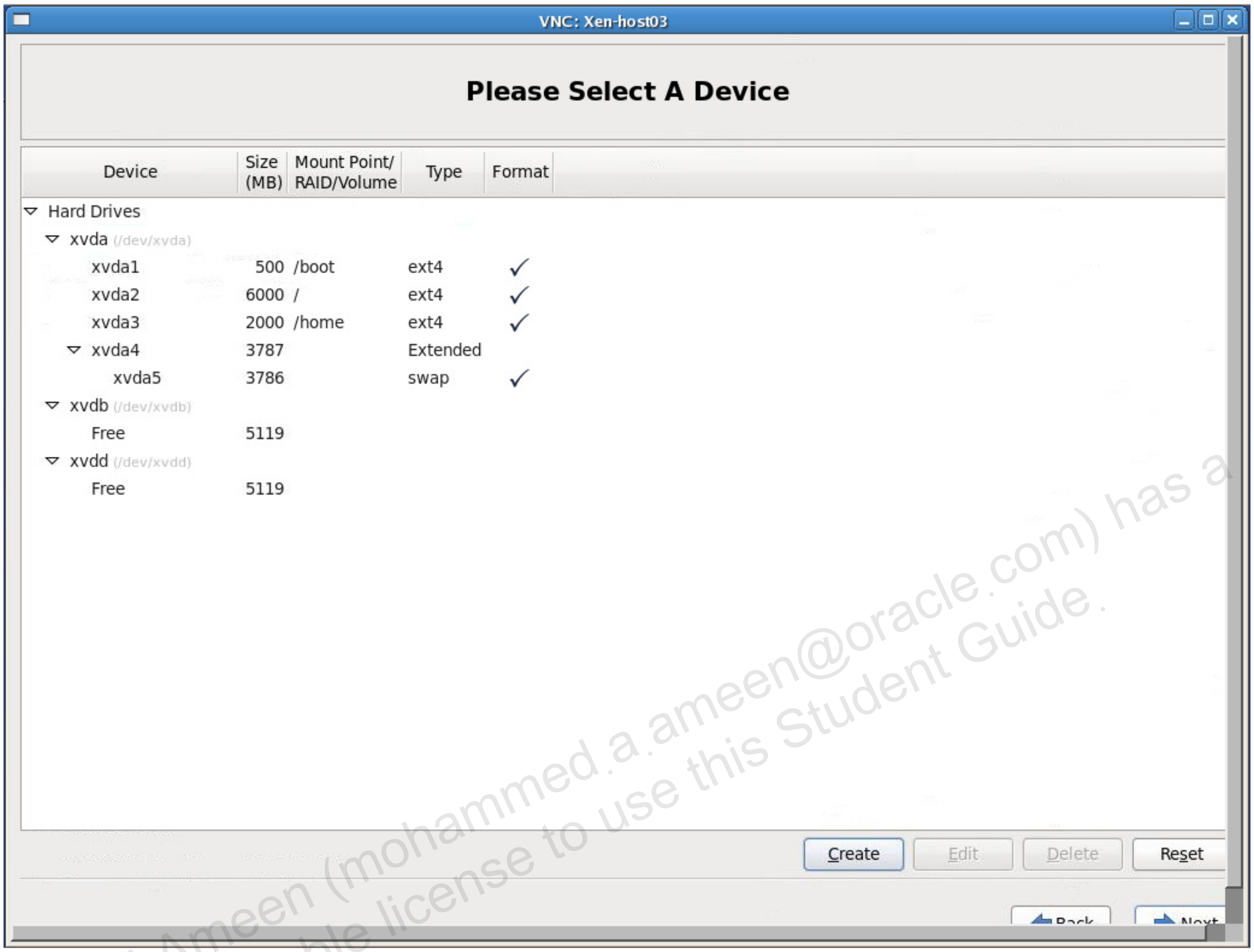
   - You indicated that the GRUB boot loader should be installed on the 12 GB virtual drive.

- You configured the boot loader during the installation process as follows:



- You accepted the defaults, which resulted in installing the GRUB boot loader on the MBR on `/dev/xvda`.
- Recall from the lecture portion of the lesson that only a subset of the GRUB boot loader is written to the MBR. The remainder of the GRUB boot loader is written to the `/boot` partition.

- The partition layout that you created during the installation process is shown here:



Use the `df` command to confirm the mounted partitions.

```
# df -h
Filesystem          Size   Used  Avail  Use%   Mounted on
/dev/xvda2          5.7G   3.0G   2.5G   55%   /
tmpfs              1002M    80K  1002M    1%   /dev/shm
/dev/xvda1          477M    55M   397M   13%   /boot
/dev/xvda3          1.9G   3.9M   1.8G    1%   /home
/dev/sr0            3.7G   3.7G      0  100%   /media/OL6.5 ...
```

- Notice that `/boot` is a separate partition mounted on `/dev/xvda1`.

5. Explore the /boot partition. (Your system may be different from the following sample output.)

   a. Use the ls command to view the /boot/grub directory.

```
# ls /boot/grub
device.map      grub.conf          minix_stage1_5      stage2
e2fs_stage1_5   iso9660_stage1_5   reiserfs_stage1_5
ufs2_stage1_5   fat_stage1_5       jfs_stage1_5
splash.xpm.gz   vstafs_stage1_5    ffs_stage1_5        menu.lst
stage1          xfs_stage1_5
```

   - GRUB works in stages: stage 1, stage 1_5, and stage 2.
   - Notice the various "stage" files in the /boot/grub directory.
   - Notice the splash screen file, splash.xpm.gz, displayed during the boot process.
   - Notice the GRUB configuration file, grub.conf, in this directory.

   b. Use the cat command to view the GRUB configuration file /boot/grub/grub.conf.

```
# cat /boot/grub/grub.conf
# grub.conf generated by anaconda
...
#boot=/dev/sda
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.qz
hiddenmenu
title Oracle Linux Server Unbreakable Enterprise Kernel (3.8.13-
16.2.1.el6uek.x86_64)
      root (hd0,0)
      kernel /vmlinuz-3.8.13-16...
      initrd /initramfs-3.8.13-16...
title Oracle Linux Server Red Hat Compatible Kernel (2.6.32-
431.el6.x86_64)
      root (hd0,0)
      kernel /vmlinuz-2.6.32-431...
      initrd /initramfs-2.6.32-431...
```

   - Each **title** line specifies a bootable kernel.
   - With Oracle Linux, two bootable kernels are included:
     - UEK (3.8.13-16.2.1.el6uek.x86_64)
     - Red Hat Compatible Kernel (2.6.32-431.el6.x86_64)
   - Each title has associated **root**, **kernel**, and **initrd** directives.
   - With **default=0**, the kernel associated with the first title is booted.

c. Use the `grep` command to list the bootable kernels.

```
# grep title /boot/grub/grub.conf
title Oracle Linux Server Unbreakable Enterprise Kernel (3.8.13-16.2.1.el6uek.x86_64)
title Oracle Linux Server Red Hat Compatible Kernel (2.6.32-431.el6.x86_64)
```

d. Use the `ls` command to view the kernel files.

```
# ls /boot/vmlinuz*
vmlinuz-2.6.32-431.el6.x86_64
vmlinuz-3.8.13-16.2.1.el6uek.x86_64
```

e. Use the `ls` command to view the initial RAM disk files.

```
# ls /boot/init*
initramfs-2.6.32-431.el6.x86_64.img
initramfs-3.8.13-16.2.1.el6uek.x86_64.img
```

6. View the kernel boot parameters.

a. Use the `grep` command to view the kernel boot parameters.

```
# grep kernel /boot/grub/grub.conf
...
    kernel /vmlinuz-3.8.13-16.2.1.el6uek.x86_64 ro
root=UUID=... rd_NO_LUKS rd_NO_LVM LANG-en_US.UTF-8 rd_NO_MD
SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM
rhgb quiet
...
```

- Kernel boot parameters are appended to the “kernel” entry in `grub.conf`.
- These parameters are written to the `/proc/cmdline` file and viewable after boot.
- The UUID value is not displayed.

b. Use the `cat` command to view the `/proc/cmdline` file.

```
# cat /proc/cmdline
ro root=UUID=... rd_NO_LUKS rd_NO_LVM LANG-en_US.UTF-8 rd_NO_MD
SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM
rhgb quiet
```

- Notice that the content is the same as the kernel boot parameters in `grub.conf`.

Practice 4-2: Booting Different Kernels

Overview

In this practice, you edit the GRUB configuration file to boot different kernels.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Boot the Red Hat Compatible Kernel.
   a. Use the `uname -r` command to display the current running kernel:

```
# uname -r
3.8.13-16.2.1.el6uek.x86_64
```

   - The current running kernel is the UEK.

   b. Use the `vi` editor to change the **default** entry in `grub.conf` from `0` to `1`, and change the **timeout** entry from `5` to `30`.

```
# vi /boot/grub/grub.conf
...
default=0 (old entry)
timeout=5 (old entry)

default=1 (new entry)
timeout=30 (new entry)
```

   - The **timeout** entry is increased to `30` seconds primarily for the next practice.

   c. Use the `reboot` command to reboot your system:

```
# reboot
...
```

   - After you reboot your system, your VNC session closes.

   d. Connect to **host03** by using VNC.
      1) Run the `vncviewer&` command.

```
# vncviewer&
```

      - The VNC Viewer: Connection Details dialog box is displayed.

      2) Enter the command `localhost:<port_number>`, substituting the correct port number for the **host03** guest. For example, if the port number is 5903, enter the following and click **OK**:

```
localhost:5903
```

   e. Select `Oracle Student` from the GNOME login window, password is `oracle`.
   f. Right-click the GNOME desktop and select **Open in Terminal** from the pop-up menu.

g. In the terminal window, become the `root` user by entering the `su -` command followed by the `root` password `oracle`.

```
$ su -
Password: oracle
# whoami
root
```

h. Use the `uname -r` command to display the new running kernel:

```
# uname -r
2.6.32-431.el6.x86_64
```

- With **`default=1`**, the kernel associated with the second title in `grub.conf` is booted.
- The second bootable kernel is the Red Hat Compatible Kernel.

2. Boot the UEK.

a. Use the `vi` editor to change the **default** entry in `/boot/grub/grub.conf` from `1` back to `0`.

b. Use the `reboot` command to reboot your system.

- After you reboot your system, your VNC session closes.

c. Connect to **host03** by using VNC.

1) Run the command `vncviewer&` from **dom0**.

2) Enter `localhost:<port_number>`, substituting the correct port number for the **host03** guest (for example, `localhost:5903`).

d. Select `Oracle Student` from the GNOME login window. The password is `oracle`.

e. Right-click the GNOME desktop and open a terminal window.

f. Become the `root` user by entering the `su -` command. The password is `oracle`.

g. Use the `uname -r` command to ensure that you are running the UEK:

```
# uname -r
3.8.13-16.2.1.el6uek.x86_64
```

- With **`default=0`**, the kernel associated with the first title in `grub.conf` is booted.
- The first bootable kernel is the UEK.

Practice 4-3: Using the GRUB Menu

Overview

In this practice, you access the GRUB menu and modify the kernel boot parameter to boot into single-user mode.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Display the current run level.

   Use the `who -r` command (or `runlevel` command) to display the current run level:

   ```
   # who -r
       run-level 5  2013-12-06 10:25
   # runlevel
   N 5
   ```

   - The current run level in the example is 5.

2. Reboot your system and display the GRUB menu.

   a. Reboot your VM by using the `reboot` command:

   ```
   # reboot
   ```

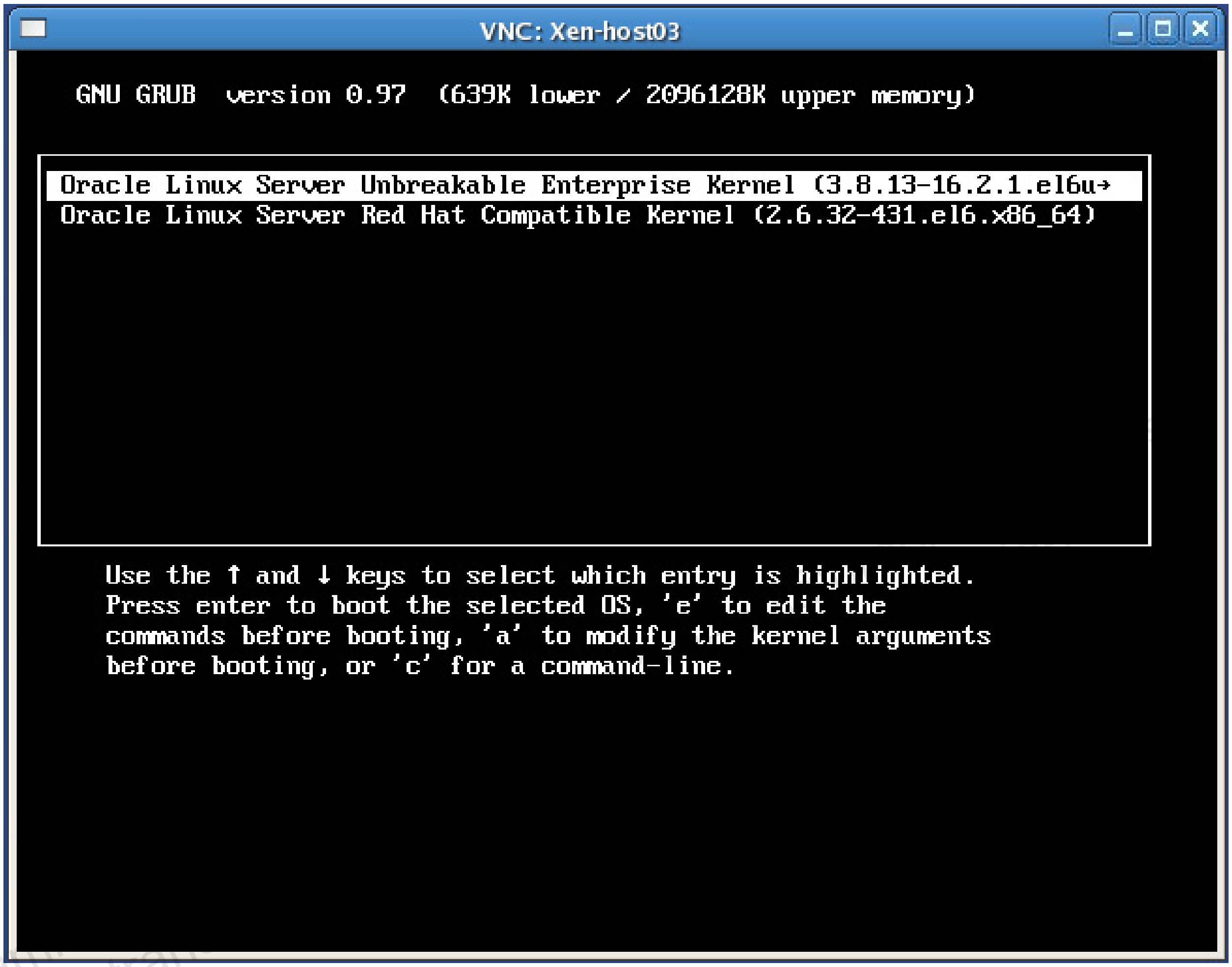
   - After you reboot your system, your VNC session closes.

   b. Run the command `vncviewer&` from **dom0**.

   c. Enter `localhost:<port_number>`, substituting the correct port number for the **host03** guest (for example, `localhost:5903`).

d. Press the **Esc** key before the timeout expires to display the GRUB menu.

- The GRUB menu is shown here:



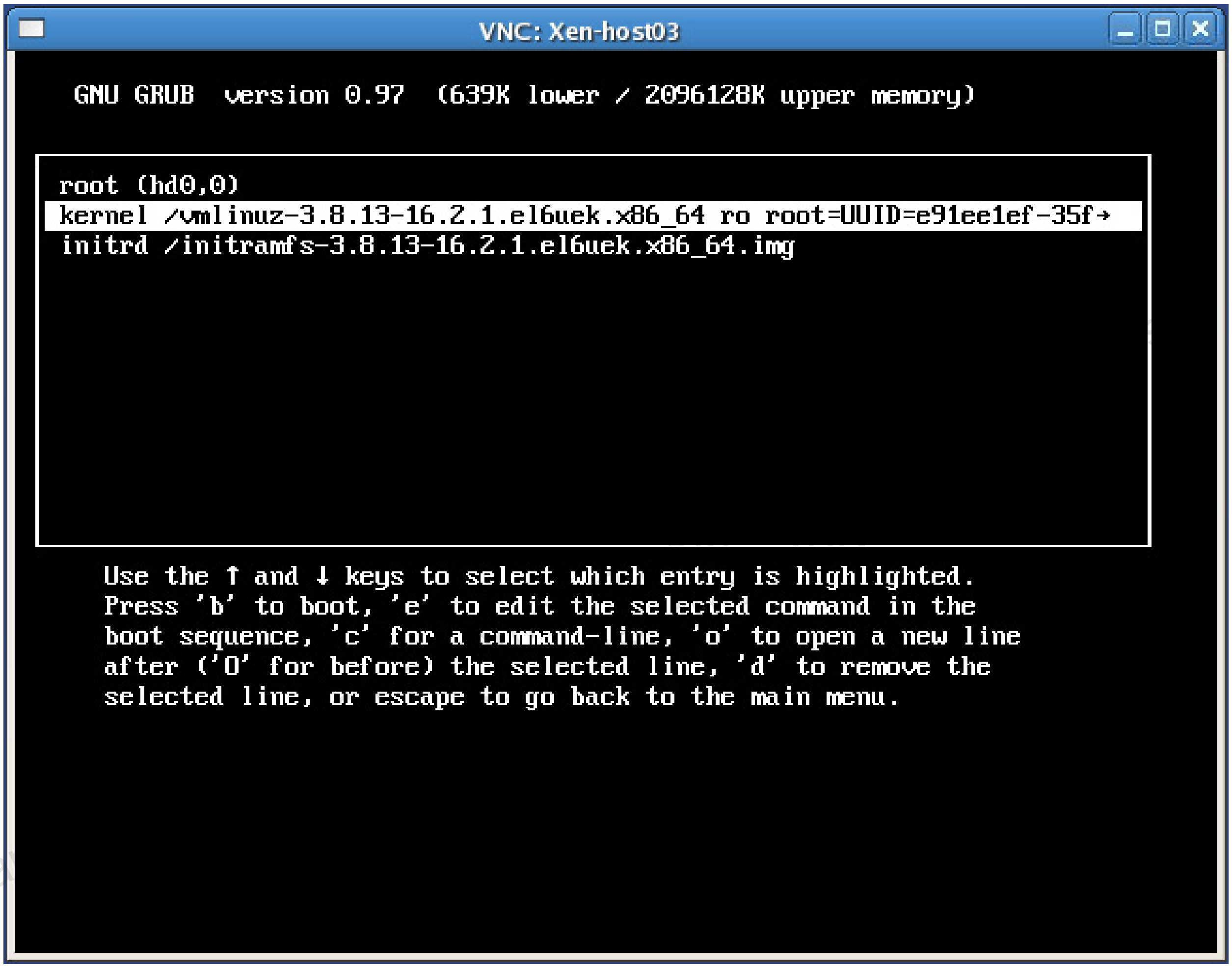
- Each entry in the GRUB menu is associated with a **title** entry in the `grub.conf` file.

3. Edit a bootable kernel entry to boot into single-user mode.

   a. Use the up and down arrow keys if necessary to highlight the first entry (UEK).

   b. Press the **e** key to edit the entry.

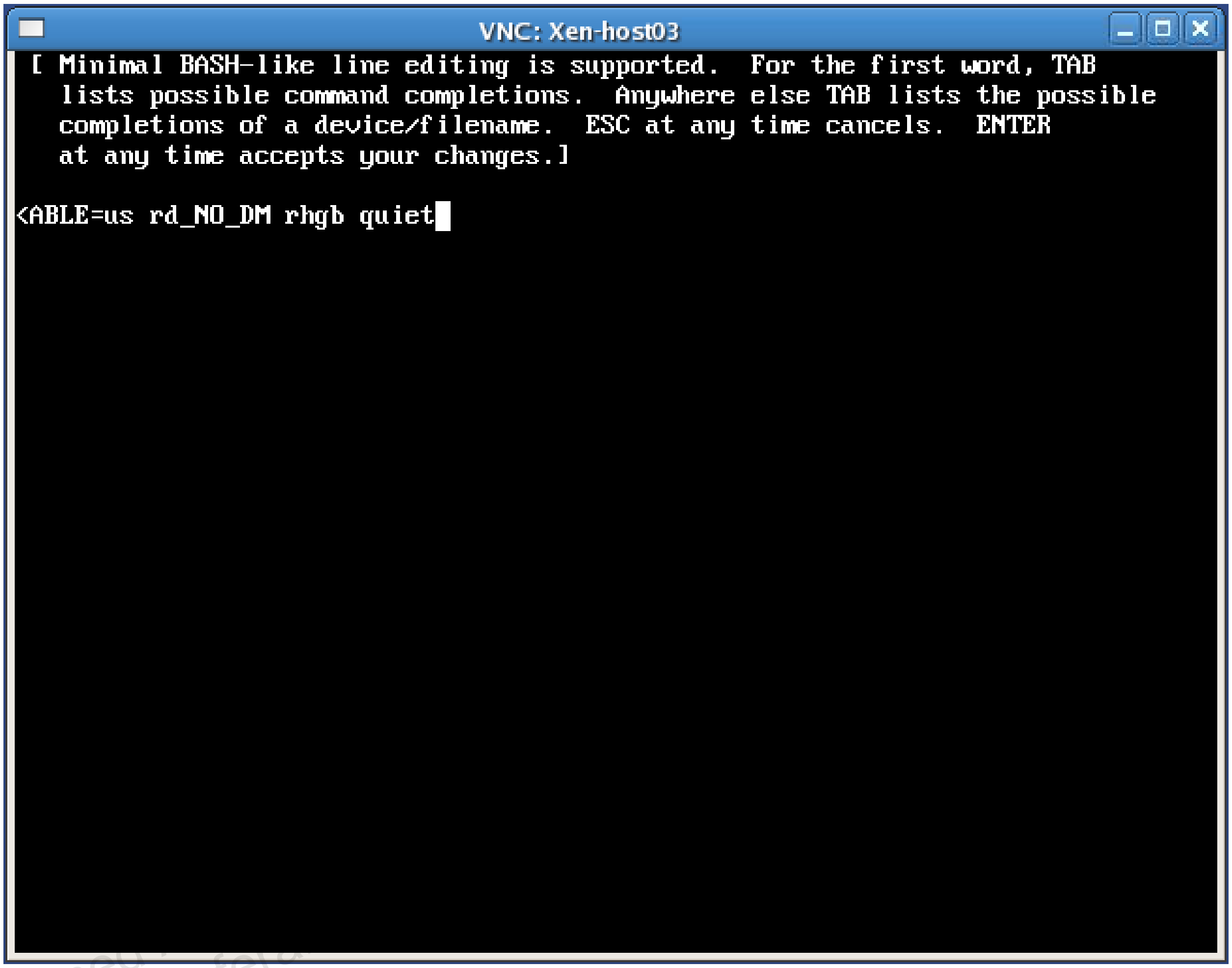
      - The associated **root**, **kernel**, and **initrd** directives for the selected title are shown:



   c. Use the up-arrow and down-arrow keys if necessary to highlight the **kernel** entry (as shown in the preceding screenshot).

   d. Press the **e** key to edit the kernel entry.

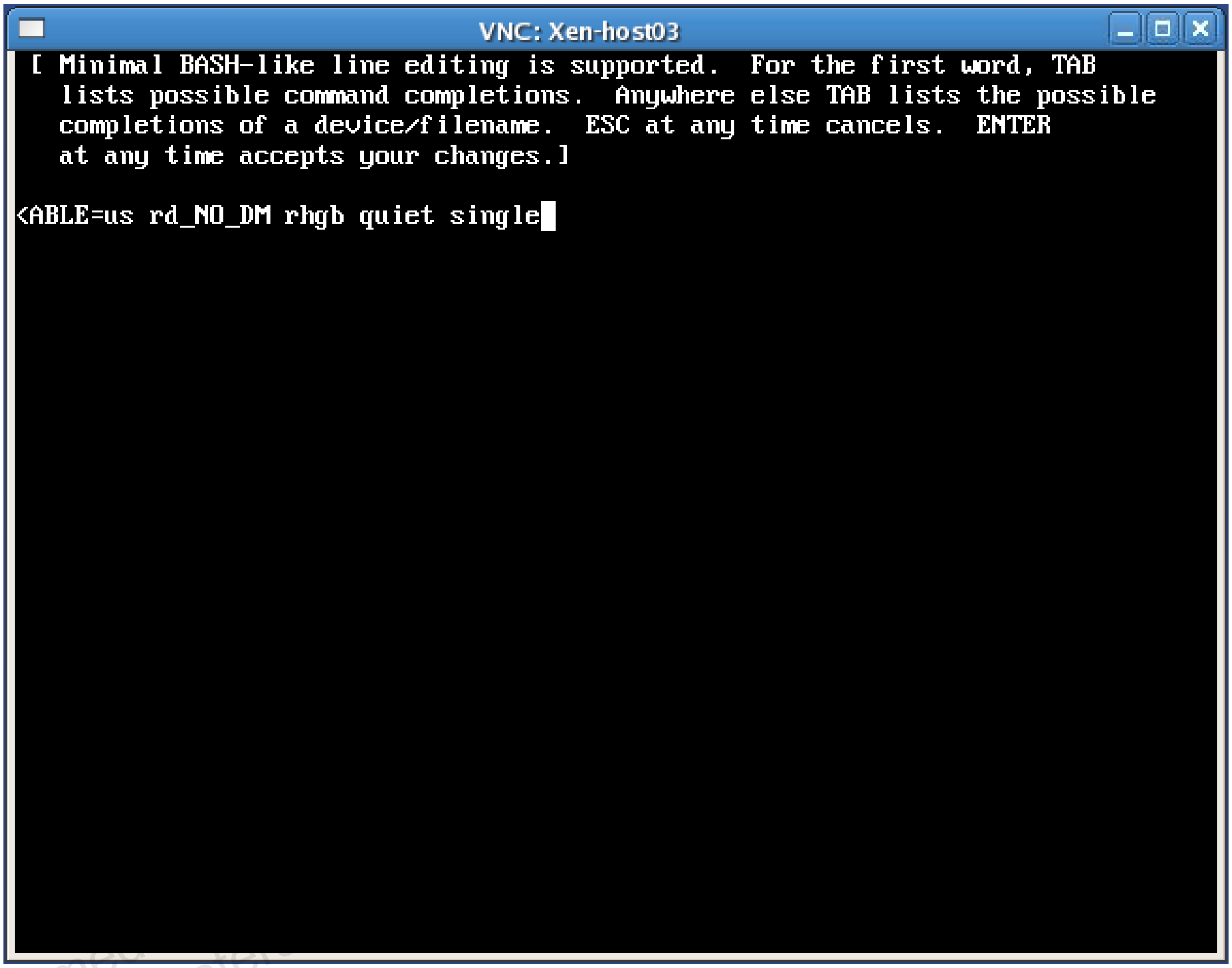
- The end of the **kernel** entry is shown as follows:



e. Use the left and right arrow keys to view the entire kernel entry.

f. After viewing the entire entry, use the right arrow key to move the cursor to the end of the line.

g. Add the word `single` to the end of the kernel line:



h. After adding the word `single` as shown in the preceding screenshot, press the **Enter** key.

i. Press the **b** key to boot.

- Note that the system boots and you are automatically logged in as the `root` user.

j. Use the `who -r` command to determine the current run level:

```
# who -r
    run-level S  2013-12-06 10:37
```

- The current run level in the example is `S`, or single-user mode.

4. Bring the system from single-user mode to the previous run level.

a. Press **Ctrl + D** to bring the system back to run level 5.

- The GNOME login window appears.

b. Select `Oracle Student` in the GNOME login window. The password is `oracle`.

c. Right-click the GNOME desktop and open a terminal window.

d. Become the `root` user by entering the `su -` command. The password is `oracle`.

```
$ su -
Password: oracle
# whoami
root
```

e. Use the `who -r` command to determine the current run level:

```
# who -r
     run-level 5  2013-12-06 10:50                 last=S
```

- The run level in the example is 5.
- The previous run level was single-user mode.

5. Change the **timeout** entry in `/boot/grub/grub.conf` back to `5`.

   Use the `vi` editor to change the **timeout** entry from `30` to `5`.

```
# vi /boot/grub/grub.conf
...
timeout=30 (old entry)

timeout=5  (new entry)
```

Practice 4-4: Changing the Default Run Level

Overview

In this practice, you change the default run level.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Edit `/etc/inittab` and change the default run level to 3.
   a. Use the `vi` editor to change the **default** run level from 5 to 3:

   ```
   # vi /etc/inittab
   ...
   id:5:initdefault: (old entry)
   id:3:initdefault: (new entry)
   ```

   b. Use the `reboot` command to reboot your system:

   ```
   # reboot
   ...
   ```

   - After you reboot your system, your VNC session closes.

   c. Connect to **host03** by using VNC.
      1) Run the `vncviewer&` command.

      ```
      # vncviewer&
      ```

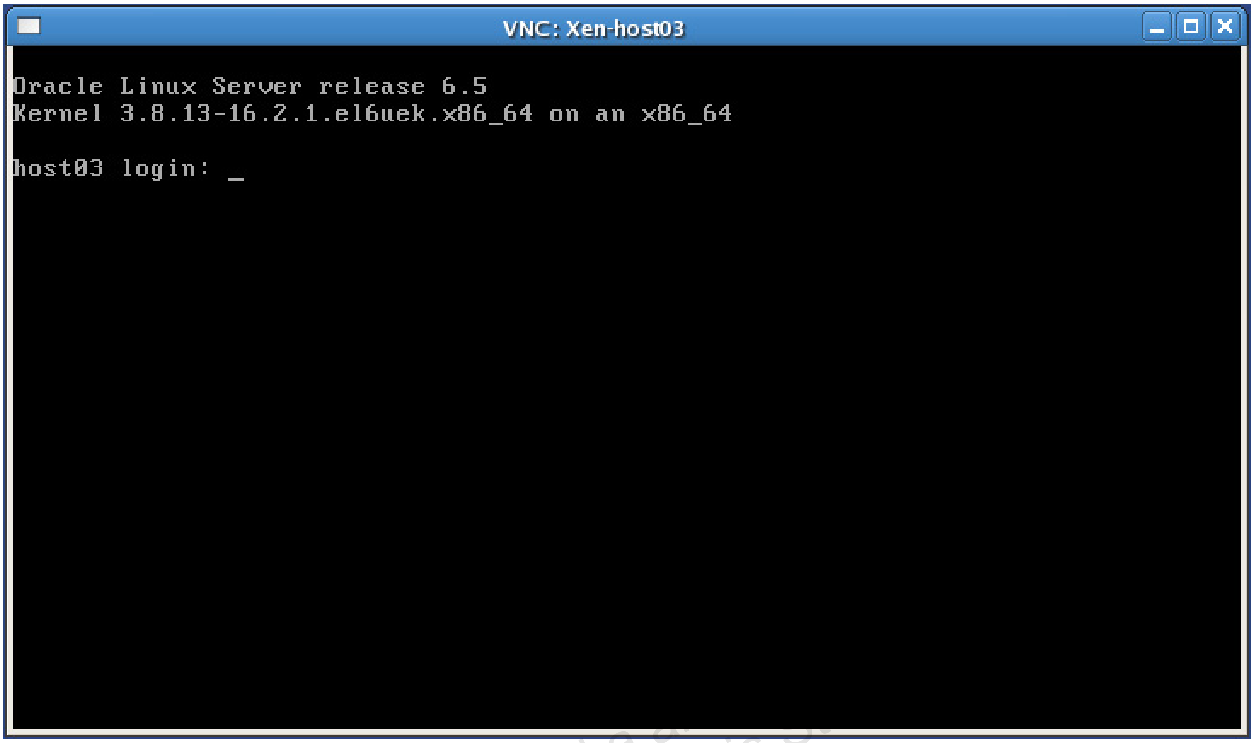
      - The VNC Viewer: Connection Details dialog box is displayed.

      2) Enter the command `localhost:<port_number>`, substituting the correct port number for the **host03** guest. For example, if the port number is 5903, enter the following and click **OK**:

      ```
      localhost:5903
      ```

      - The GNOME login does not appear. Instead, a text-based login window appears.
      - Booting to run level 3 does not start the X Window services.

- The text-based login window appears:



d. Log in as `root`. The password is `oracle`.

e. Use the `who -r` command to display the current run level:

```
# who -r
    run-level 3  2013-12-06 10:52
```

- The default run level in the example is now 3.

2. Edit `/etc/inittab` and change the default run level from 3 back to 5.

a. Use the `vi` editor to change the **default** run level from 3 back to 5:

```
# vi /etc/inittab
...
id:3:initdefault: (old entry)
id:5:initdefault: (new entry)
```

b. Use the `reboot` command to reboot your system:

```
# reboot
...
```

- After you reboot your system, your VNC session closes.

c. Connect to **host03** by using VNC.

1) Run the `vncviewer&` command.

```
# vncviewer&
```

- The VNC Viewer: Connection Details dialog box is displayed.

2) Enter the command `localhost:<port_number>`, substituting the correct port number for the **host03** guest. For example, if the port number is 5903, enter the following and click **OK**:

```
localhost:5903
```

- The GNOME login does appear in run level 5 because the X Window services are started.

d. Select `Oracle Student` from the GNOME login window. The password is `oracle`.

e. Right-click the GNOME desktop and select **Open in Terminal** from the pop-up menu.

f. In the terminal window, become the `root` user by entering the `su -` command followed by the `root` password `oracle`.

```
$ su -
Password: oracle
# whoami
root
```

g. Use the `who -r` command to display the current run level:

```
# who -r
     run-level 5  2013-12-06 10:54
```

- The default run level in the example is now 5.

Practice 4-5: Exploring and Configuring `init` Services

Overview

In this practice, you explore the `init` script directories. You also configure a service to start and stop at a specific run level.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Explore the `/etc/rc.d` directory.
   a. Use the `ls -l` command to display the contents of the directory.

```
# ls -l /etc/rc.d
drwxr-xr-x.   init.d
-rwxr-xr-x.   rc
drwxr-xr-x.   rc0.d
drwxr-xr-x.   rc1.d
drwxr-xr-x.   rc2.d
drwxr-xr-x.   rc3.d
drwxr-xr-x.   rc4.d
drwxr-xr-x.   rc5.d
drwxr-xr-x.   rc6.d
-rwxr-xr-x.   rc.local
-rwxr-xr-x.   rc.sysinit
```

   - `/etc/rc.d/rc.sysinit`: Runs once at boot time
   - `/etc/rc.d/rc`: Starts and stops services when the run level changes
   - `/etc/rc.d/rc.local`: The last script that the `init` program runs

   b. Use the `ls` command to display the contents of the `/etc/rc.d/init.d` directory.

```
# ls /etc/rc.d/init.d
abrt-ccpp   firstboot   messagebus   quota_nld   snmpd
abrtd       functions   netconsole   rdisc       snmptrapd
...
```

   - The directory contains scripts that start and stop services.
   - Note that `/etc/init.d` is a symbolic link to `/etc/rc.d/init.d`.

c. Use the `ls -l` command to display the contents of the `/etc/rc.d/rc5.d` directory.

```
# ls -l /etc/rc.d/rc5.d
lrwxrwxrwx.   K01smartd -> ../init.d/smartd
lrwxrwxrwx.   K02oddjobd -> ../init.d/oddjobd
...
lrwxrwxrwx.   K95firstboot -> ../init.d/firstboot
lrwxrwxrwx.   K99rngd -> ../init.d/rngd
lrwxrwxrwx.   S01sysstat -> ../init.d/sysstat
lrwxrwxrwx.   S02lvm2-monitor -> ../init.d/lvm2-monitor
...
lrwxrwxrwx.   S99local -> ../rc.local
```

- The directory contains symbolic links to scripts in the `/etc/rc.d/init.d` directory.
- This allows services to be stopped (K files) or started (S files) when entering a particular run level (run level 5 in this example).
- Note that `/etc/rc5.d` is a symbolic link to `/etc/rc.d/rc5.d`.

d. Use the `ls -l` command to display the services that are stopped and started when entering run level 3.

```
# ls -l /etc/rc.d/rc3.d
...
```

- Note that `/etc/rc3.d` is a symbolic link to `/etc/rc.d/rc3.d`.

2. Start the **httpd** service when entering run level 3 or run level 5.

a. Use the `find` command to display all **http** files in the `/etc/rc.d` directory.

```
# find /etc/rc.d -name "*http*"
/etc/rc.d/rc2.d/K15httpd
/etc/rc.d/rc0.d/K15httpd
/etc/rc.d/rc3.d/K15httpd
/etc/rc.d/rc6.d/K15httpd
/etc/rc.d/init.d/httpd
/etc/rc.d/rc5.d/K15httpd
/etc/rc.d/rc1.d/K15httpd
/etc/rc.d/rc4.d/K15httpd
```

- Currently, the **httpd** service is configured to stop when entering each run level.

b. Use the `service` command to verify that the **httpd** service is stopped.

```
# service httpd status
httpd is stopped
```

c. Use the `chkconfig` command to query run level information for the **httpd** service.

```
# chkconfig --list httpd
httpd  0:off  1:off  2:off  3:off  4:off  5:off  6:off
```

- This confirms the **httpd** service is not configured to run at any run level.

d. Use the `chkconfig` command to configure the **httpd** service to start at run levels 3 and 5.

```
# chkconfig --level 35 httpd on
# chkconfig --list httpd
httpd  0:off  1:off  2:off  3:on  4:off  5:on  6:off
```

- Now the service is configured to start at run levels 3 and 5.

e. Use the `find` command to display all **http** files in the `/etc/rc.d` directory.

```
# find /etc/rc.d –name "*http*"
/etc/rc.d/rc2.d/K15httpd
/etc/rc.d/rc0.d/K15httpd
/etc/rc.d/rc3.d/S85httpd
/etc/rc.d/rc6.d/K15httpd
/etc/rc.d/init.d/httpd
/etc/rc.d/rc5.d/S85httpd
/etc/rc.d/rc1.d/K15httpd
/etc/rc.d/rc4.d/K15httpd
```

- Now a "start" script exists in the run level 3 directory, `/etc/rc.d/rc3.d/S85httpd` and in the run level 5 directory, `/etc/rc.d/rc5.d/S85httpd`.

f. Use the `service` command to check the status of the **httpd** service.

```
# service httpd status
httpd is stopped
```

- The `chkconfig` command only configures a service to start or stop at a given runlevel.
- The `chkconfig` command does not actually start or stop a service.

g. Use the `reboot` command to reboot your system.

```
# reboot
...
```

- After you reboot your system, your VNC session closes.

h. Connect to **host03** by using VNC, and log in as `Oracle Student` (with password `oracle`).

1) From **dom0**, run the `vncviewer&` command.

```
# vncviewer&
```

- The VNC Viewer: Connection Details dialog box is displayed.

2) Enter the command `localhost:<port_number>`, substituting the correct port number for the **host03** guest. For example, if the port number is 5903, enter the following and click **OK**:

```
localhost:5903
```

3) Select `Oracle Student` in the GNOME login window (password is `oracle`).

i. Open a terminal window and become the `root` user.

1) Right-click the GNOME desktop and select **Open in Terminal** from the pop-up menu.

2) In the terminal window, become the `root` user by entering the `su -` command followed by the `root` password `oracle`.

```
$ su -
Password: oracle
# whoami
root
```

j. Use the `service` command to check the status of the **httpd** service.

```
# service httpd status
httpd (pid 1405) is running...
```

- The httpd service is now running.

3. Stop the **httpd** service when entering run level 3 or run level 5.

a. Use the `chkconfig` command to configure the **httpd** service so that it does not run at run levels 3 and 5.

```
# chkconfig --level 35 httpd off
# chkconfig --list httpd
httpd  0:off  1:off  2:off  3:off  4:off  5:off  6:off
```

- Now the service is configured so that it does not start at run levels 3 and 5.

b. Use the `find` command to display all **http** files in the `/etc/rc.d` directory.

```
# find /etc/rc.d -name "*http*"
/etc/rc.d/rc2.d/K15httpd
/etc/rc.d/rc0.d/K15httpd
/etc/rc.d/rc3.d/K15httpd
/etc/rc.d/rc6.d/K15httpd
/etc/rc.d/init.d/httpd
/etc/rc.d/rc5.d/K15httpd
/etc/rc.d/rc1.d/K15httpd
/etc/rc.d/rc4.d/K15httpd
```

- Now the "start" scripts in the run level 3 directory, `/etc/rc.d/rc3.d/`, and in the run level 5 directory, `/etc/rc.d/rc5.d/`, are gone.

c. Use the `service` command to check the status of the **httpd** service.

```
# service httpd status
httpd (pid 1405) is running...
```

- The `chkconfig` command only configures a service. It does not start or stop the service.

d. Use the `service` command to stop the **httpd** service.

```
# service httpd stop
Stopping httpd:                          [  OK  ]
# service httpd status
httpd is stopped
```

Practice 4-6: Exploring and Configuring Upstart Jobs

Overview

In this practice, you:

- View the files contained in the Upstart package
- View Upstart jobs
- Create a test Upstart job
- Start, stop, and view the status of the `test` job

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Query the Upstart package.
   a. Use the `rpm -q upstart` command to view the installed Upstart package.
      - The `rpm` command is discussed in Lesson 6, "Package Management."

```
# rpm -q upstart
upstart-0.6.5-12.el6.x86_64
```

   b. Use the `rpm -ql upstart` command view the files contained in the Upstart package.

```
# rpm -ql upstart
/etc/dbus-1/system.d/Upstart.conf
/etc/init
/etc/init/init-system.dbus.conf
/sbin/halt
/sbin/init
/sbin/initctl
/sbin/poweroff
/sbin/reboot
/sbin/reload
/sbin/restart
/sbin/runlevel
/sbin/shutdown
/sbin/start
/sbin/status
/sbin/stop
/sbin/telinit
...
```

      - Notice that the `/etc/init` directory is provided by the Upstart package.
      - Notice that some of the key system utilities such as `halt`, `init`, `reboot`, `shutdown`, and others are provided by the Upstart package.

- Notice that the `initctl` utility and the `start`, `stop`, and `status` commands are provided by the Upstart package.
- Also notice that the package provides the `man` pages for these utilities.

2. View Upstart jobs.
   - Upstart jobs are located in the `/etc/init` directory.

   a. Use the `cd` command to change to the `/etc/init` directory. Use the `ls` command to list the contents of the directory.

```
# cd /etc/init
# ls
control-alt-delete.conf   rcS-sulogin.conf
init-system-dbus.conf     readahead-collector.conf
kexec-disable.conf        readahead.conf
plymouth-shutdown.conf    readahead-disable-services.conf
prefdm.conf               serial.conf
quit-plymouth.conf        splash-manager.conf
rc.conf                   start-ttys.conf
rcS.conf                  tty.conf
rcS-emergency.conf
```

   - Notice all files have a `.conf` extension.

   b. Use the `initctl list` command to list Upstart jobs.

```
# initctl list
rc stop/waiting
tty (/dev/tty3) start/running, process ...
tty (/dev/tty2) start/running, process ...
tty (/dev/tty6) start/running, process ...
tty (/dev/tty5) start/running, process ...
tty (/dev/tty4) start/running, process ...
plymouth-shutdown stop/waiting
control-alt-delete stop/waiting
rcS-emergency stop/waiting
...
```

   - Notice there is a one-to-one association for each job defined in `/etc/init` with the exception of the `tty` job.
   - The number of active tty consoles is set by the `ACTIVE_CONSOLES` directive in the `/etc/sysconfig/init` file.

3. Create a `test` job file.

   a. Use the `vi` editor to create `/etc/init/test.conf` with the following entries:

```
# vi test.conf
author "<your_name>"
description "Test Upstart Configuration"
```

- Both `author` and `description` are valid stanzas in Upstart job configuration files.

b. Use the `initctl list` command to list Upstart jobs.

```
# initctl list
rc stop/waiting
test stop/waiting
...
```

- Notice the `test` job is now included in the output of `initctl list`.

c. Use the `vi` editor to edit `/etc/init/test.conf` and add the following entries (in bold):

```
# vi test.conf
author "<your_name>"
description "Test Upstart Configuration"
script
    sleep 1000
end script
```

4. Start, stop, and view the status of the `test` job.

a. Use the `initctl start test` command to start the Upstart `test` job.

```
# initctl start test
test start/running, process ...
```

b. Use the `initctl status test` command to view the status of the Upstart `test` job.

```
# initctl status test
test start/running, process ...
```

c. Use the `initctl stop test` command to stop the Upstart `test` job.

```
# initctl stop test
test stop/waiting
```

d. Use the `status test` command to view the status of the Upstart `test` job.

```
# status test
test stop/waiting
```

e. Use the `start test` command to start the Upstart `test` job.

```
# start test
test start/running, process ...
```

f. Use the `stop test` command to stop the Upstart `test` job.

```
# stop test
test stop/waiting
```

- The `start`, `stop`, and `status` can be run directly without using `initctl`.

5. View Upstart logging information and set logging priority.

   a. Use the `tail` command to view the latest entries written to the `/var/log/messages`, file.

```
# tail /var/log/messages
...
<date_time> host03 init: test main process (...) killed ...
```

   - Notice that by default, only the stopping of Upstart jobs is logged.

   b. Use the `initctl log-priority` command to display the logging priority.

```
# initctl log-priority
message
```

   - Notice the default logging priority is `message`.

   c. Use the `initctl log-priority debug` command to change the logging priority to `debug`.

```
# initctl log-priority debug
```

   d. Use the `start test` command to start the Upstart `test` job.

```
# start test
test start/running, process ...
```

   e. Use the `tail` command to view the latest entries written to the `/var/log/messages`, file.

```
# tail /var/log/messages
...
<date_time> host03 init: Connection from private client
<date_time> host03 init: test goal changed from stop to start
<date_time> host03 init: test state changed from waiting to
starting
<date_time> host03 init: Handling starting event
<date_time> host03 init: test state changed from starting to
pre-start
<date_time> host03 init: test state changed from pre-start to
spawned
<date_time> host03 init: test main process (...)
<date_time> host03 init: test state changed from spawned to
post-start
<date_time> host03 init: test state changed from post-start to
running
<date_time> host03 init: Handling starting event
```

   - Notice with logging priority set to `debug`, state changes (`waiting`, `starting`, `pre-start`, `spawned`, `post-start`, and `running`) are logged.

f. Use the `stop test` command to stop the Upstart `test` job.

```
# stop test
test stop/waiting
```

g. Use the `initctl log-priority message` command to change the logging priority to `message`.

```
# initctl log-priority message
```

6. Start and stop the `test` job based on an event.

a. Use the `vi` editor to edit `/etc/init/test.conf` and add the following entries (in bold):

```
# vi test.conf
author "<your_name>"
description "Test Upstart Configuration"
start on event1
stop on event2
script
    sleep 1000
end script
```

b. Use the `status test` command to display the status of the Upstart `test` job.

```
# status test
test stop/waiting
```

c. Use the `initctl emit event1` command to cause the `test` job to start.

```
# initctl emit event1
```

d. Use the `status test` command to display the status of the Upstart `test` job.

```
# status test
test start/running, process ...
```

- Notice the `test` job started on the `event1` event.

e. Use the `initctl emit event2` command to cause the `test` service to stop.

```
# initctl emit event2
```

f. Use the `status test` command to display the status of the Upstart `test` job.

```
# status test
test stop/waiting
```

- Notice the `test` job stopped on the `event2` event.

g. Do not enter the following; this is information only. You can also start and/or stop a job based on the starting/stopping of another service.

- The following entry in `test.conf` starts the `test` job when *<other-service>* is started:

```
start on started <other-service>
```

- The following entry in `test.conf` stops the `test` job when *<other-service>* is stopped:

  `stop on stopping` *<other-service>*

h. Do not enter the following; this is information only. You can also start and/or stop a job based on the runlevel.

- The following entry in `test.conf` starts the `test` job on run levels 3 and 5:

  `start on runlevel [35]`

- The following entry in `test.conf` stops the `test` job on run level not equal to 3 or 5:

  `stop on runlevel [!35]`

i. Use the `rm` command to remove the `/etc/init/test.conf` file.

```
# rm /etc/init/test.conf
rm: remove regular file '/etc/init/test.conf'? y
```

Practices for Lesson 5: System Configuration

Chapter 5

Practices for Lesson 5: System Configuration

Practices Overview

In these practices, you modify the system configuration files, view and modify kernel settings, and view hardware device and device driver attributes.

Practice 5-1: Exploring `/etc/sysconfig`

Overview

In this practice, you perform the following tasks:

- Explore the `/etc/sysconfig` directory.
- Explore selected files in this directory.
- See that many initialization scripts derive values from files in the directory.
- View the documentation file that describes the entries in the directory.
- Make changes to the content of files in the directory.
- Observe the effect of the changes.

Assumptions

You are the `root` user on the **host03** VM.

Tasks

1. Explore the `/etc/sysconfig` directory.
   a. Use the `ls` command to display the contents of the `/etc/sysconfig` directory.

```
# ls /etc/sysconfig
atd          iptables-config  nfs              selinux
auditd       iptables.old     nspluginwrapper  smartmontools
autoconfig   iptables         ntpd             snmpd
...
```

- System configuration settings are stored in these files.
- The values from many of these files are initialized by their respective initialization script when the service is started.

b. Use the `grep` command to display specific `/etc/sysconfig` entries in initialization scripts.
   - The following commands search for the existence of files in the `/etc/sysconfig` directory contained in initialization scripts in the `/etc/init.d` directory.
   - The purpose of these commands is to show that initialization scripts for various Linux services get configuration information and/or additional command-line arguments from files in the `/etc/sysconfig` directory.

```
# grep /etc/sysconfig/sshd /etc/init.d/sshd
[ -f /etc/sysconfig/sshd ] && . /etc/sysconfig/sshd
# grep /etc/sysconfig/nfs /etc/init.d/nfs
[ -f /etc/sysconfig/nfs ] && . /etc/sysconfig/nfs
# grep /etc/sysconfig/atd /etc/init.d/atd
config=/etc/sysconfig/atd
# grep /etc/sysconfig/init /etc/init.d/functions
  if [ -f /etc/sysconfig/init ]; then
      . /etc/sysconfig/init
...
# grep /etc/sysconfig/network /etc/init.d/*
/etc/init.d/dnsmasq:. /etc/sysconfig/network
/etc/init.d/netconsole:. /etc/sysconfig/network
...
```

- The `/etc/init.d/sshd` script starts and stops the OpenSSH server daemon.
  - The configuration file for the `sshd` service is `/etc/sysconfig/sshd`.
- The `/etc/init.d/nfs` script starts and stops the NFS services.
  - The configuration file for the NFS services is `/etc/sysconfig/nfs`.
- The `/etc/init.d/atd` script starts and stops the `atd` daemon.
  - The `/etc/sysconfig/atd` file specifies additional command-line arguments for `atd`.
- The `/etc/init.d/functions` script contains functions used by most or all scripts in the `/etc/init.d` directory.
  - The `functions` script reads configuration information from the `/etc/sysconfig/init` file, which controls how the system appears and functions during the boot process.
- The `/etc/sysconfig/network` file contains network configuration information which is read by several scripts in the `/etc/init.d` directory.

2. Explore the `/usr/share/doc/init*/sysconfig.txt` file.

   a. Use the `cd` command to change to the `/usr/share/doc/init*` directory.

```
# cd /usr/share/doc/init*
# pwd
/usr/share/doc/initscripts-9.03.40
# ls
changes.ipv6         ipv6-6to4.howto    README-init
sysconfig.txt        COPYING            ipv6-tunnel.howto
static-routes-ipv6   sysvinitfiles
```

   b. Use the `less` command to display the contents of the `sysconfig.txt` file.

```
# less sysconfig.txt
...
```

   c. Search the contents of the `sysconfig.txt` file for **`/etc/sysconfig`**.

      - While viewing the file using the `less` command, use the forward slash (`/`) followed by **`etc/sysconfig`** to search for this string.
      - Press the **n** key (lowercase for “next”) to display the next instance of the string.

   d. Continue to view the **`/etc/sysconfig`** entries in this file.

   e. At the **`/etc/sysconfig/network`** entry, notice some of the variables initialized in this file. Here are some examples:

```
NETWORKING=yes|no
HOSTNAME=<fqdn by default, but whatever hostname you want>
GATEWAY=<gateway IP>
```

   f. Press the **q** key to quit the “less” command.

3. Change the system time zone.

   a. Use the `cat` command to display the contents of the `/etc/sysconfig/clock` file.

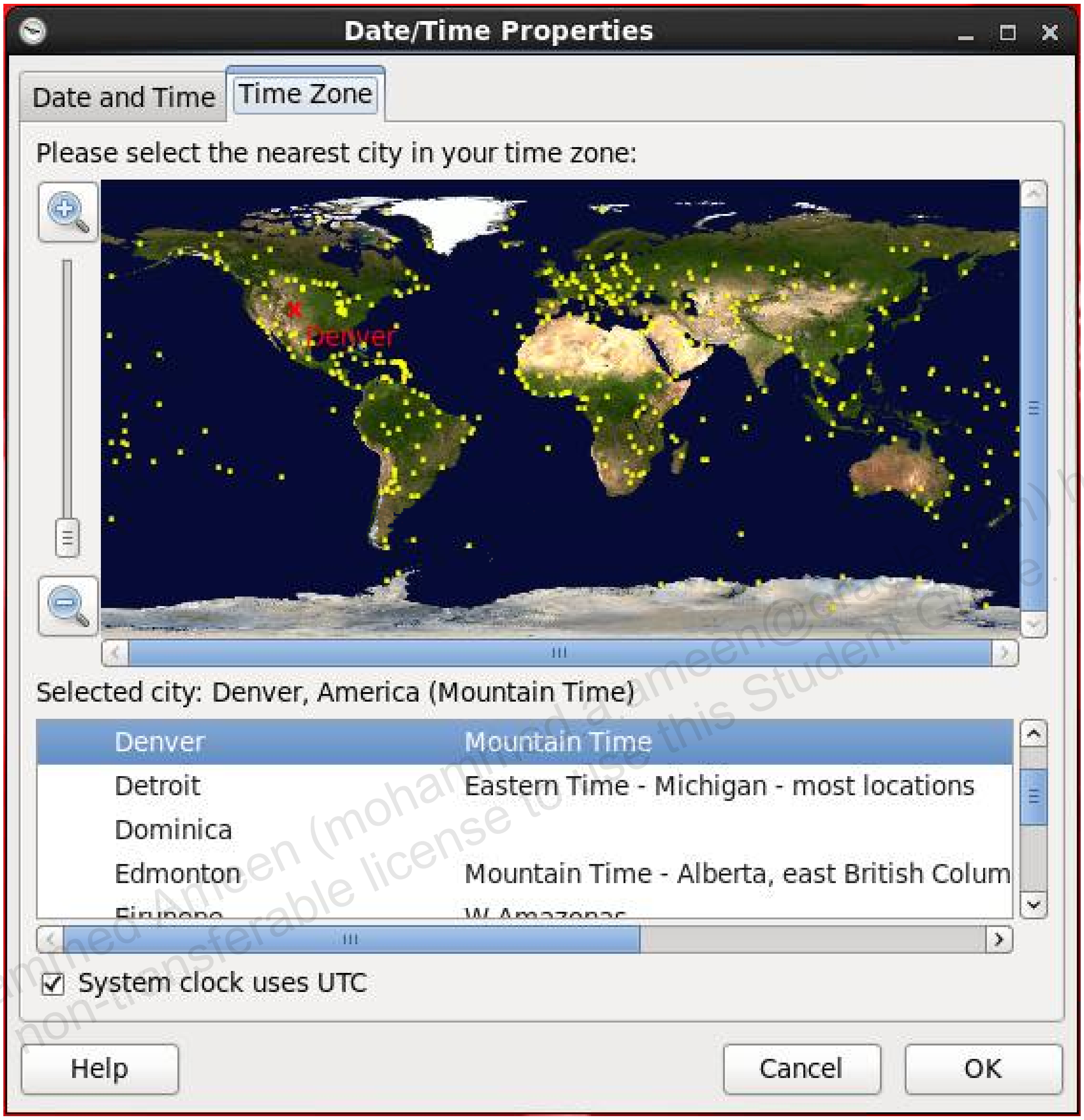
```
# cat /etc/sysconfig/clock
# The time zone of the system...
# This file is only for evaluation by system-config-date...
ZONE=”America/Denver”
```

      - Time zone settings are stored in this file.
      - You can change the time zone and date/time settings after installation by editing this file or by running the `system-config-date` command.

   b. Change the time zone by running the `system-config-date` command.

```
# system-config-date
```

- Running the `system-config-date` command displays the following window.



  - This window has both a **Date and Time** tab and a **Time Zone** tab.

c. Change the **Time Zone** to any different zone (for example, Detroit) and click **OK**.

d. Use the `cat` command to display the contents of the `/etc/sysconfig/clock` file again.

```
# cat /etc/sysconfig/clock
...
ZONE="America/Detroit"
```

  - Note that the `/etc/sysconfig/clock` file is updated with the new time zone.

e. Run the `system-config-date` command again and change the time zone back to the appropriate zone.

f. Display the contents of the `/etc/sysconfig/clock` file to confirm that the change is correct.

- Alternatively, you can edit the `/etc/sysconfig/clock` file, link the `/usr/share/zoneinfo` file to `/etc/localtime`, and then run the `system-config-date` command to confirm the change. See the following page for details: http://www.redhat.com/advice/tips/timezone.html.

Practice 5-2: Exploring `/proc`

Overview

In this practice, you explore the `proc` file system (directory), view various files and directories that represent the current state of the kernel, and change the value of current settings.

Assumptions

You are the `root` user on the **host03** VM.

Tasks

1. Explore the `proc` file system (directory).
   a. Use the `ls` command to display the contents of the `/proc` directory.

```
# ls /proc
1      1305  1612  1840  2012  3    879  kallsyms  self
10     1310  1618  1841  2013  30   883  kcore     slabinfo
1052   1328  1659  1843  2014  353  9    keys      softirqs
...
```

   - Directories with numerical names are named after a program's process ID.

   b. Use the `ls` command to display entries without numerical names.

```
# ls -ld /proc/*[a-z]* | less
dr-xr-xr-x.  acpi
-r--r--r--.  buddyinfo
dr-xr-xr-x.  bus
-r--r--r--.  cgroups
-r--r--r--.  cmdline
-r--r--r--.  cpuinfo
...
```

   - Notice that some entries are files and some entries are directories.

   c. View the `meminfo` file to display information about RAM.

```
# less /proc/meminfo
MemTotal:  2050816 kB
MemFree:   1037256 kB
...
```

d. View the `cpuinfo` file to display information about the processor(s) used by your system.

```
# less /proc/cpuinfo
processor     : 0
vendor_id     : GenuineIntel
cpu_family    : 6
model         : 23
...
```

e. View the `devices` file to display information about the various character and block devices currently configured.

```
# less /proc/devices
Character devices:
  1 mem
  4 /dev/vc/0
  4 tty
...
Block devices:
  1 ramdisk
  2 fd
...
```

f. View the `/proc/fs/ext4` directory to display the ext4 file systems currently configured.

```
# ls /proc/fs/ext4
xvda1  xvda2  xvda3
```

2. Explore process directories in the `proc` file system.

a. Use the `ls` command to display only the directories in `/proc` with numerical names.

```
# ls -d /proc/*[0-9]*
1     1149  13    1464  1525  1698  1819  1846  19    2014  ...
10    1150  1304  1472  1527  17    1822  1848  1903  2015  ...
1052  1167  1305  1483  1542  1707  1825  1849  1905  2054  ...
...
```

b. Show that one process directory exists for each process running on your system

```
# ls -d /proc/*[0-9]* | wc -l
128
# ps -e | wc -l
129
```

- The ps command produces one more entry due to the column headings.

```
# ps -e | less
PID  TTY          TIME   CMD
  1  ?        00:00:00  init
...
```

c. View the contents of the process directory for PID=1.

```
# cd /proc/1
# ls -l
dr-xr-xr-x.  attr
-r--------.  auxv
-r--r--r--.  cgroup
--w-------.  clear_refs
-r--r--r--.  cmdline
-rw-r--r--.  comm
-rw-r--r--.  coredump_filter
-r--r--r--.  cpuset
lrwxrwxrwx.  cwd -> /
-r--------.  environ
lrwxrwxrwx.  exe -> /sbin/init
dr-x------.  fd
...
```

- Notice that some entries are files, some entries are directories, and some entries are symbolic links.

d. Use the less command to display the status of PID=1.

```
# less status
Name:   init
State:  S (sleeping)
Tgid:   1
Pid:    1
PPid:   0
...
```

3. Change the values of kernel settings.

a. Use the cat command to check whether IP Forwarding is enabled.

```
# cat /proc/sys/net/ipv4/ip_forward
0
```

- With the value set to 0, IP Forwarding is disabled.

b. Use the `echo` command to enable IP Forwarding (for example, to set up a Linux router or gateway).

```
# echo 1 > /proc/sys/net/ipv4/ip_forward
# cat /proc/sys/net/ipv4/ip_forward
1
```

c. Use the `echo` command to define the local port range used by TCP and UDP traffic.

```
# echo "32768 61000" > /proc/sys/net/ipv4/ip_local_port_range
```

- Notice that the first local port and the last local port allowed are surrounded by quotation marks.

Practice 5-3: Exploring the sysfs File System

Overview

In this practice, you explore the sysfs file system. You view the virtual block devices and virtual interfaces and determine which power states are supported on your system.

Assumptions

You are the root user on the **host03** VM.

Tasks

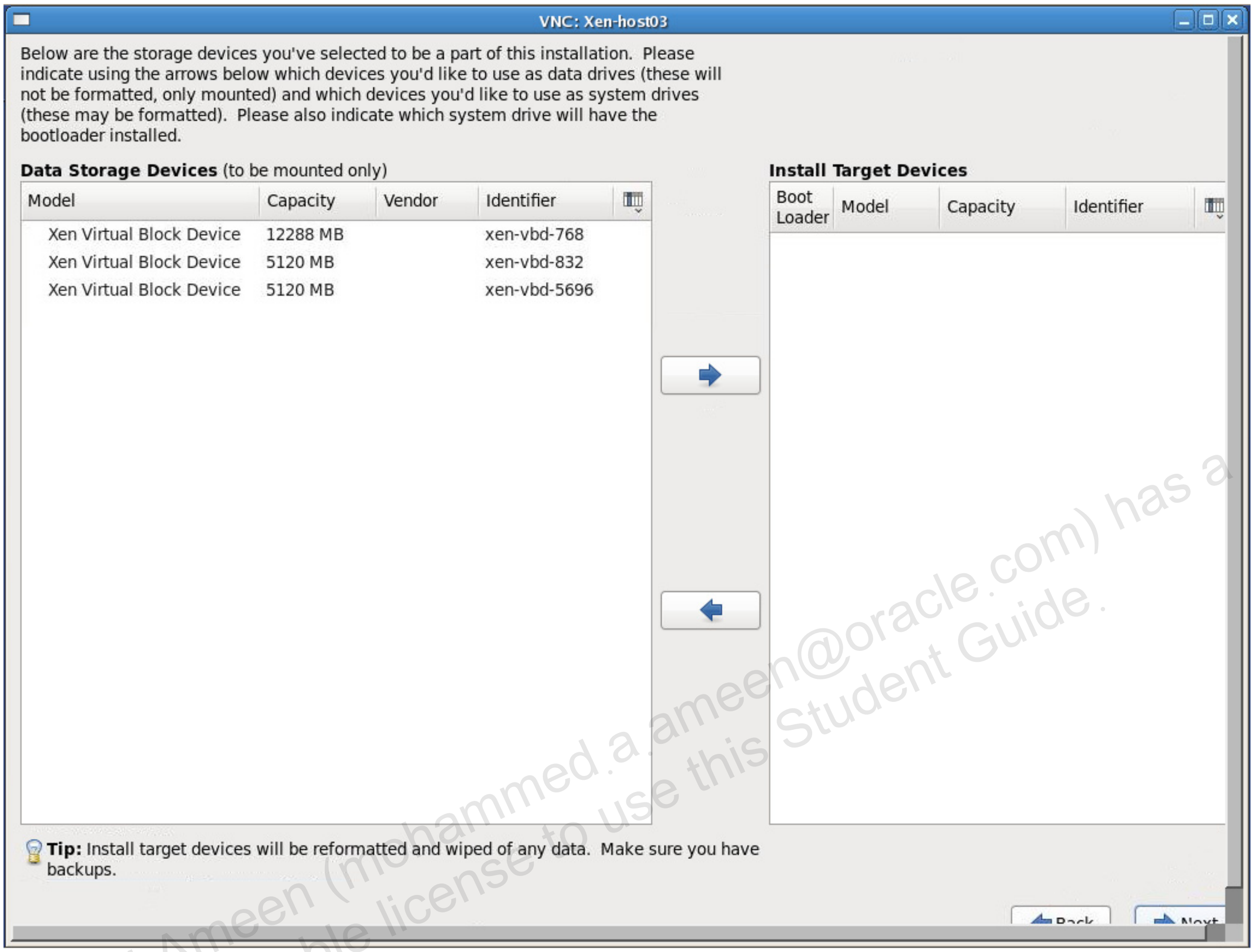
1. Explore the /sys directory
   a. Use the ls command to display the contents of the /sys directory.

```
# ls -l /sys
drwxr-xr-x.  block
drwxr-xr-x.  bus
drwxr-xr-x.  class
drwxr-xr-x.  dev
drwxr-xr-x.  devices
drwxr-xr-x.  firmware
...
```

   b. Use the ls command to display the virtual disk block (**xvd**) devices on your system.

```
# ls -l /sys/block | grep xvd
lrwxrwxrwx.  xvda -> ../devices/vbd-768/block/xvda
lrwxrwxrwx.  xvdb -> ../devices/vbd-832/block/xvdb
lrwxrwxrwx.  xvdd -> ../devices/vbd-5696/block/xvdd
```

- Recall these entries from the installation:



c. Traverse the `/sys/bus` directory and display the virtual interface (**vif**) devices.

```
# cd /sys/bus/xen/devices
# ls vif*
vif-0:
devtype  driver  modalias  net  nodename  power  subsystem
uevent
vif-1:
devtype  driver  modalias  net  nodename  power  subsystem
uevent
```

d. Display the operational state, MAC address, and MTU of **eth0**.

```
# cd vif-0/net/eth0
# pwd
/sys/bus/xen/devices/vif-0/net/eth0
# ls
...
# cat operstate
up
# cat address
00:16:3e:00:01:03
# cat mtu
1500
```

e. Determine which power states are supported.

```
# cd /sys/power
# ls
disk  image_size  pm_test  pm_trace  resume  state
# cat state
mem  disk
```

- `mem` means Suspend-to-RAM.
- `disk` means Suspend-to-Disk.

Practice 5-4: Using `sysctl`

Overview

In this practice, you use the `sysctl` utility and view the `sysctl` configuration file.

Assumptions

You are the `root` user on the **host03** VM.

Tasks

1. Use the `sysctl` utility.
   a. Use the `sysctl` command to disable IP Forwarding.

```
# cat /proc/sys/net/ipv4/ip_forward
1
# sysctl -w net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
# cat /proc/sys/net/ipv4/ip_forward
0
```

   b. Use the `sysctl` command to display the current kernel settings.

```
# sysctl –a
...
debug.exception-trace = 1
...
dev.raid.speed_limit_min = 1000
dev.scsi.logging_level = 0
...
fs.inode-nr = 48928   370
fs.inode-state = 48928   370  0  0  0  0  0
...
kernel.sched_child_runs_first = 0
kernel.sched_latency_ns = 6000000
...
vm.page-cluster = 3
vm.panic_on_oom = 0
...
net.ipv4.route.gc_thresh = -1
net.ipv4.route.gc_timeout = 300
...
```

```
net.ipv6.neigh.default.mcast_solicit = 3
net.ipv6.neigh.default.proxy_delay = 80
...
net.netfilter.nf_log.0 = NONE
net.netfilter.nf_log.1 = NONE
...
```

2. Use the `less` command to view the contents of the `/etc/sysctl.conf` file.

```
# less /etc/sysctl.conf
# Kernel sysctl configuration file for Red Hat Linux
# For binary values, 0 is disabled, 1 is enabled. See sysctl(8) and
# sysctl.conf(5) for more details.
# Controls IP packet forwarding
net.ipv4.ip_forward = 0
# Controls source route verification
net.ipv4.conf.default.rp_filter = 1
...
```

- Changes that are made by using both `echo` and `sysctl` are lost when the system is rebooted.
- To preserve custom settings, add them to the `/etc/sysctl.conf` file.
- Values that are added to this file take effect each time the system boots.

Practices for Lesson 6: Package Management

Chapter 6

Practices for Lesson 6: Package Management

Practices Overview

In these practices, you do the following:

- Use the `rpm` utility to perform software package management.
- Create a local `yum` repository and use the `yum` utility.
- Become familiar with the Oracle Unbreakable Linux Network (ULN).

You access the Oracle Public Yum Server in Practices for Lesson 13, after the network is completely configured.

Practice 6-1: Using the rpm Utility

Overview

In this practice, you use the rpm utility to query installed packages, install packages, upgrade packages, and remove packages.

Assumptions

- You are the root user on the **host03** VM.
- Sample output is provided. Package names and versions shown might differ from actual output.

Tasks

1. Query packages by using rpm.
   a. Query all installed packages.

```
# rpm -qa
...
rng-tools-2-13.el6.x86_64
media-player-info-6-1.el6.noarch
gnome-python2-gnome-2.28.0-3.el6.x86_64
libshout-2.2.2-5.1.el6.x86_64
```

   b. Query whether a specific package (**bash**, for example) is installed.

```
# rpm -qa bash
bash-4.1.2-15.el6_4.x86_64
```

   c. Get detailed information about the **bash** package.

```
# rpm -qi bash
Name        : bash
Version     : 4.1.2
Release     : 15.el6_4
Install Date: Mon 09 Dec 2013
Group       : System Environment/Shells
...
```

   d. List the files in the **bash** package.

```
# rpm -ql bash
/bin/bash
/bin/sh
/etc/skel/.bash_logout
/etc/skel/.bash_profile
...
```

e. Perform a reverse search. That is, determine what package the `/etc/sysconfig` file belongs to.

```
# rpm -qf /etc/sysconfig
filesystem-2.4.30-3.el6.x86_64
```

f. List the configuration files associated with the `cups` package.

```
# rpm -qc cups
/etc/cups/classes.conf
/etc/cups/client.conf
/etc/cups/cupsd.conf
/etc/cups/lpoptions
...
```

2. Install packages by using `rpm`.

a. Use the `df` command to determine the mount point for Oracle Linux installation media.

```
# df -h
Filesystem ... Mounted on
...
/dev/sr0    ... /media/OL6.5 x86_64 Disc 1 20131125
```

- In this example, the Oracle Linux installation media is mounted on `/media/OL6.5 x86_64 Disc 1 20131125`.
- If you connected to **host03** using the `ssh` command rather than using `vncviewer` as instructed, you might not have this mount point.
- Perform the following optional task only if you do not have this mount point; otherwise, proceed to task 2b.

Optional task (skip this optional task if task 2a successfully returned the mount point for the Oracle Linux installation media). Use the `mkdir` command to create the `/media/"OL6.5 x86_64 Disc 1 20131125"` directory (quotes are required because the directory name contains white space). Ignore the error message if the `OL*` directory already exists. Use the `mount` command to mount the Oracle Linux installation media:

```
# mkdir /media/"OL6.5 x86_64 Disc 1 20131125"
# mount -o ro /dev/sr0 /media/"OL6.5 x86_64 Disc 1 20131125"
```

b. Use the `cd` command to change to the `/media/OL*` directory.

```
# cd /media/OL*
```

c. Use the `ls` command to list the contents of the directory.

```
# ls
EFI          Packages      ResilientStorage
EULA         README-en     RPM-GPG-KEY
...
```

- Notice the **Packages** subdirectory.

d. Use the `cd` command to change to the **Packages** directory, and then list the contents of the directory.

```
# cd Packages
# ls
389-ds-base-1.2.11.15-29.el6.x86_64.rpm
...
zsh-4.3.10-7.el6.x86_64.rpm
```

e. Verify that the **zsh** package (Z-Shell) is not already installed.

```
# zsh
-bash: zsh: command not found
# rpm -qa zsh
```

- In this example, the **zsh** package is not installed.

f. Install the **zsh** package by using `rpm`.

```
# rpm -Uvh zsh-4.3.10-7.el6.x86_64.rpm
...
Preparing...      ############################### [100%]
   1:zsh          ############################### [100%]
```

g. Verify that the **zsh** package is now installed.

```
# rpm -qa zsh
zsh-4.3.10-7.el6.x86_64
```

h. Run the `zsh` command, and then display the process ID of `zsh`.

```
# zsh
# ps
PID
7887   pts/0   00:00:00 su
7895   pts/0   00:00:00 bash
16690  pts/0   00:00:00 zsh
16697  pts/0   00:00:00 ps
```

i. Use the `exit` command to log out of `zsh`.

```
# exit
# ps
PID
7887   pts/0   00:00:00 su
7895   pts/0   00:00:00 bash
16697  pts/0   00:00:00 ps
```

3. Remove packages by using `rpm`.

   a. Remove the **zsh** package.

```
# rpm -e zsh
```

   b. Verify that the **zsh** package has been removed.

```
# rpm -qa zsh
# zsh
-bash: /bin/zsh: No such file or directory
```

Practice 6-2: Creating a Local Yum Repository

Overview

In this practice, you install the `createrepo` utility (if necessary), create a Local Yum repository, and create a `repo` file to enable the local repository.

Assumptions

- You are the `root` user on the **host03** VM.
- The Oracle Linux `dvd.iso` image is mounted on `/media`.

Tasks

1. Disable the Public Yum repositories.
   a. Use the `vi` editor to edit the `/etc/yum.repos.d/public-yum-ol6.repo` file and set all “`enabled=1`” and “`enabled=$uek|uekr3`” to “`enabled=0`”.
      - You enable the Public Yum repository in a practice for Lesson 13, after the network is completely configured.

```
# vi /etc/yum.repos.d/public-yum-ol6.repo
[public_ol6_latest]
...
enabled=1                          (old value)
enabled=0                          (new value)
...
[public_ol6_UEK_latest]
...
enabled=$uek                       (old value)
enabled=0                          (new value)
...
[public_ol6_UEKR3_latest]
...
enabled=$uekr3                     (old value)
enabled=0                          (new value)
...
```

   b. Run the `yum clean all` command to clean up the `yum` cache.

```
# yum clean all
Loaded plugins: refresh-packagekit
Cleaning repos:
Cleaning up Everything
```

2. Ensure that the Oracle Linux `dvd.iso` image is mounted on `/media`.

   Use the `df` command to display the mounted file systems.

```
# df -h
Filesystem            Size  Used Avail Use% Mounted on
...
/dev/sr0              3.7G  3.7G     0 100% /media/OL6.5 x86_64
Disc...
```

   - Notice the OL6.5 media is mounted on `/media`.

3. Ensure that the `createrepo` utility is installed. If it is not, install it now.

   a. Use the `rpm` command to check whether the `createrepo` package is installed.

```
# rpm -qa createrepo
```

   - In this example, the package is not installed.

   b. Change to the `/media/OL*/Packages` directory to install `createrepo`.

```
# cd /media/OL*/Packages
# ls creat*
createrepo-0.9.9-18.0.1.el6.noarch.rpm
```

   c. Use the `rpm` command to install the package.

```
# rpm -Uvh createrepo-0.9.9-18.0.1.el6.noarch.rpm
...
error: Failed dependencies:
        python-deltarpm is needed by createrepo...
```

   - The package does not install because a dependent package is not installed.

   d. Attempt to use `yum` to install because `yum` resolves dependencies.

```
# yum install createrepo
Loaded plugins: refresh-packagekit
Setting up Install Process
No package createrepo available
Error: Nothing to do
```

   - The `yum` command fails because no repository is available.

   e. The only alternative is to use `rpm` and install the dependent package first, and then install `createrepo`.

```
# rpm -Uvh python-deltarpm-3.5-0.5.20090913git.el6.x86_64.rpm
error: Failed dependencies:
        deltarpm = 3.5-0.5.20090913git.el6 is needed by python-
deltarpm-3.5-0.5.20090913git.el6.x86_64
```

f. This package failed to install due to another dependency. Use the `rpm` command to install this `deltarpm` package. Then install `python-deltarpm` then `createrepo`.

```
# rpm -Uvh deltarpm-3.5-0.5.20090913git.el6.x86_64.rpm
...
Preparing...             ################################# [100%]
   1:deltarpm            ################################# [100%]
# rpm -Uvh python-deltarpm-3.5-0.5.20090913git.el6.x86_64.rpm
...
Preparing...              ################################# [100%]
   1:python-deltarpm      ################################# [100%]
# rpm -Uvh createrepo-0.9.9-18.0.1.el6.noarch.rpm
...
Preparing...              ################################# [100%]
   1:createrepo           ################################# [100%]
```

g. Verify that `createrepo` is installed and in the search path.

```
# which createrepo
/usr/bin/createrepo
```

4. Create the local repository.

a. Change to the `/media` directory.

```
# cd /media
# ls
OL6.5 x86_64 Disc 1 20131125
```

b. Use the `createrepo` command to create a repository of the current directory.

- This command takes several minutes to complete.

```
# createrepo .
Spawning worker 0 with 26278 pkgs
Workers Finished
Gathering worker results
Saving Primary metadata
Saving file lists metadata
Saving other metadata
Generating sqlite DBs
Sqlite DBs complete
```

c. View the results of the `createrepo` command.

```
# pwd
/media
# ls -l
drwx-r-x-r-x.        OL6.5 x86_64 Disc 1 20131125
drwx-r-x-r-x.        repodata
```

- Notice that the **`repodata`** directory has been created.

d. View the contents of the **repodata** directory.

```
# ls -l repodata
-rw-r--r--   ...other.sqlite.bz2
-rw-r--r--   ...other.xml.gz
-rw-r--r--   ...filelists.sqlite.bz2
-rw-r--r--   ...filelists.xml.gz
...
-rw-r--r--   repomd.xml
```

e. Run the `yum clean all` command to clean up the `yum` cache.

```
# yum clean all
Loaded plugins: refresh-packagekit, security
Cleaning repos: InstallMedia
Cleaning up Everything
```

f. Change to the `yum` repository directory and use the `vi` editor to create the `iso.repo` file:

```
# cd /etc/yum.repos.d
# vi iso.repo
[Myrepo]
name=Oracle Linux
baseurl=file:///media
enabled=1
gpgkey=file:///media/RPM-GPG-KEY-oracle
gpgcheck=1
```

g. In this example, the GPG key is located on the Oracle Linux `dvd.iso` image.

```
# cd /media/OL*
# ls *GPG*
RPM-GPG-KEY   RPM-GPG-KEY-oracle
```

- There are two files that contain the GPG key.
- The files are the same, so either can be used.

```
# diff RPM-GPG-KEY RPM-GPG-KEY-oracle
```

h. Use the `cp` command to copy the `RPM-GPG-KEY-oracle` file from `/media/OL*` to `/media`.

```
# cp /media/OL*/RPM-GPG-KEY-oracle /media
```

i. Manually install the public key.

```
# rpm --import RPM-GPG-KEY-oracle
```

j. Confirm the import of the public key.

```
# rpm -qa gpg-pubkey
gpg-pubkey-...
```

k. Use the `yum repolist` command to list the configured repositories.

```
# yum repolist
...
repo id                    repo name                    status
Myrepo                     Oracle Linux                 26,278
repolist: 26,278
```

Practice 6-3: Using the yum Utility

Overview

In this practice, you use the yum utility to list both installed packages and packages available to be installed, install a package, check for packages with updates available, update a package, and remove a package.

Assumptions

- This practice assumes that you have a yum repository configured.
- This practice is performed from the **host03** VM.
- You are logged on as root on **host03**.

Tasks

1. List packages by using yum.
   a. List all packages installed on your system and all packages available in all configured repositories.

```
# yum list
Loaded plugins: refresh-packagekit, security
Installed Packages
ConsoleKit.x86_64              0.4.1-3.el6          @anaconda-
Ora...
ConsoleKit.libs.x86_64         0.4.1-3.el6          @anaconda-
Ora...
...
zlib-devel.i686                1.2.3-27.el6         Myrepo
zlib-devel.x86_64              1.2.3-27.el6         Myrepo
zsh.x86_64                     4.3.10-7.el6         Myrepo
```

   - Notice that some packages were installed during installation (**@anaconda…**) and some are available in the new repository (**Myrepo**).

   b. List only the installed packages.

```
# yum list installed
Loaded plugins: refresh-packagekit, security
Installed Packages
ConsoleKit.x86_64              0.4.1-3.el6        @anaconda-Ora...
...
zlib.x86_64                    1.2.3-29.el6       @anaconda-Ora...
```

   c. List only the packages that are available to be installed from enabled repositories.

```
# yum list available
Available Packages
389-ds-base.x86_64               1.2.11.15-29.el6       Myrepo
389-ds-base-libs.i686            1.2.11.15-29.el6       Myrepo
...
```

d. List the name of the package to which the `/etc/sysconfig/crond` file belongs.

```
# yum provides /etc/sysconfig/crond
cronie-1.4.4-12.el6.x86_64
...
```

2. Install packages by using `yum`.

   Install the **389-ds-base** package.

   - Answer **y** when prompted.

```
# yum install 389-ds-base
Setting up Install Process
Resolving Dependencies
--> Running transaction check
...
--> Finished Dependency Resolution
Dependencies Resolved
===================================================
Package
===================================================
Installing:
 389-ds-base
Installing for dependencies:
 389-ds-base-libs
...
Transaction Summary
===================================================
Install     14 Package(s)
Total download size: 3.5 M
Installed size: 12 M
Is this ok [y/N]: y
Downloading packages
...
Running rpm_check debug
Running Transaction Test
...
  Installing : svrcore-4.0.4-5.1.el6.x86_64
...
Installed:
 389-ds-base.x86_64 0:1.2.11.15-29.el6
Dependency Installed:
 389-ds-base-libs.x86_64 0:1.2.11.15-29.el6
...
Complete!
```

3. Update packages by using `yum`.
   - You have no packages with updates available.
   - This exercise is an example and shows only the steps needed to update a package.

   a. Check which installed packages have updates available.

```
# yum check-update
DeviceKit-power.x86_64      014-1.el6            Myrepo
...
```

   b. Update the **DeviceKit-power** package.

```
# yum update DeviceKit-power
Setting up Update Process
Resolving Dependencies
--> Running transaction check
...
--> Finished Dependency Resolution
Dependencies Resolved
===================================================
Package
===================================================
Updating:
 DeviceKit-power
Transaction Summary
===================================================
Upgrade     1 Package(s)
Total download size: 90 k
Is this ok [y/N]: y
Downloading packages
DeviceKit-power-014-3.el6.x86_64.rpm
Running rpm_check debug
Running Transaction Test
...
  Updating   : DeviceKit-power-014-3.el6.x86_64
  Cleanup    : DeviceKit-power-014-3.el6.x86_64

Updated:
 DeviceKit-power-014-3.el6.x86_64
Complete!
```

4. Remove packages by using `yum`.

   Remove the **389-ds-base** package.
   - If the following "`yum remove`" command fails, run the "`yum clean all`" command to clean up the `yum` cache, then run the "`yum remove`" command again.

- Answer **y** when prompted.

```
# yum remove 389-ds-base
Setting up Remove Process
Resolving Dependencies
--> Running transaction check
...
--> Finished Dependency Resolution
Dependencies Resolved
===================================================
Package
===================================================
Removing:
 389-ds-base
Transaction Summary
===================================================
Remove     1 Package(s)
Installed size: 4.9 M
Is this ok [y/N]: y
Downloading packages
Running rpm_check debug
Running Transaction Test
...
  Erasing   : 389-ds-base-1.2.9.14-1.el6.x86_64

Removed:
 389-ds-base.x86_64 0:1.2.9.14-1.el6
Complete!
```

Practice 6-4: Unbreakable Linux Network (ULN)

Overview

In this practice, you become familiar with the ULN web interface. You cannot access the ULN without a Customer Support Identifier (CSI). All you can do is read through the tasks in this practice to help understand the capabilities of ULN.

Assumptions

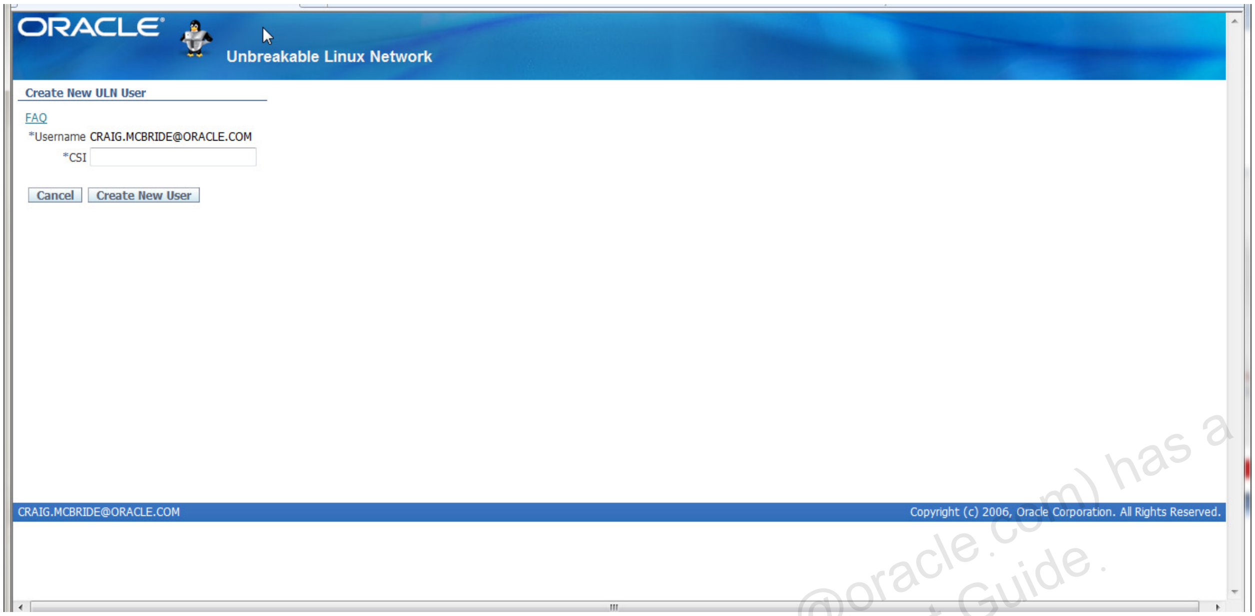
This practice is not intended to be a hands-on exercise.

Tasks

1. Access the Unbreakable Linux Network (ULN).
    a. From a browser, enter the following URL for the Unbreakable Linux Network (ULN): https://linux.oracle.com.

    b. Click **Sign On** to be prompted for login credentials.

c. The first time you access the ULN, use your email address and your Customer Support Identifier (CSI) to log in. You are then required to create a password.



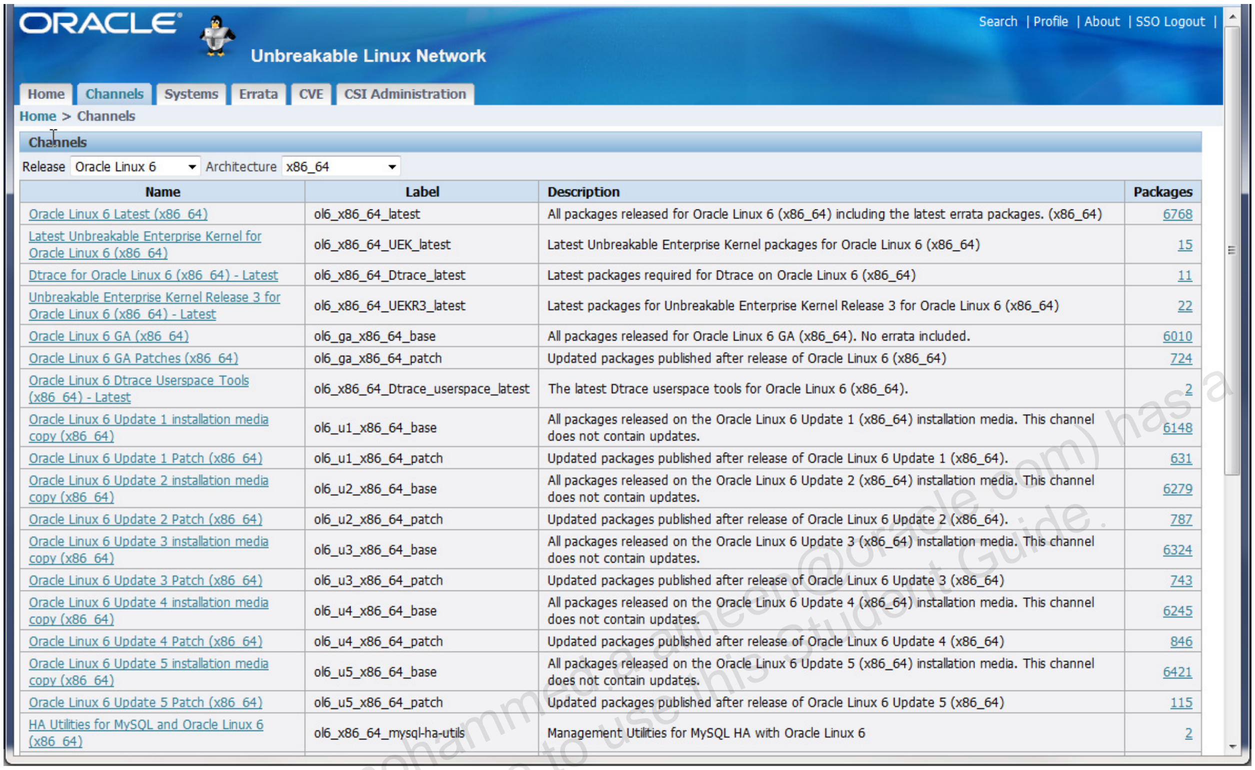
- In the future, use your email address and password to log in to ULN.
- When you register for the first time, you are subscribed to the default channel of `ol6_<arch>_latest`.
- After registering, you can specify a different channel by using the web interface.

2. Use the ULN web interface.

   a. You are first taken to the ULN **Home** page.

- The ULN web interface allows subscribers to register systems and subscribe to channels.

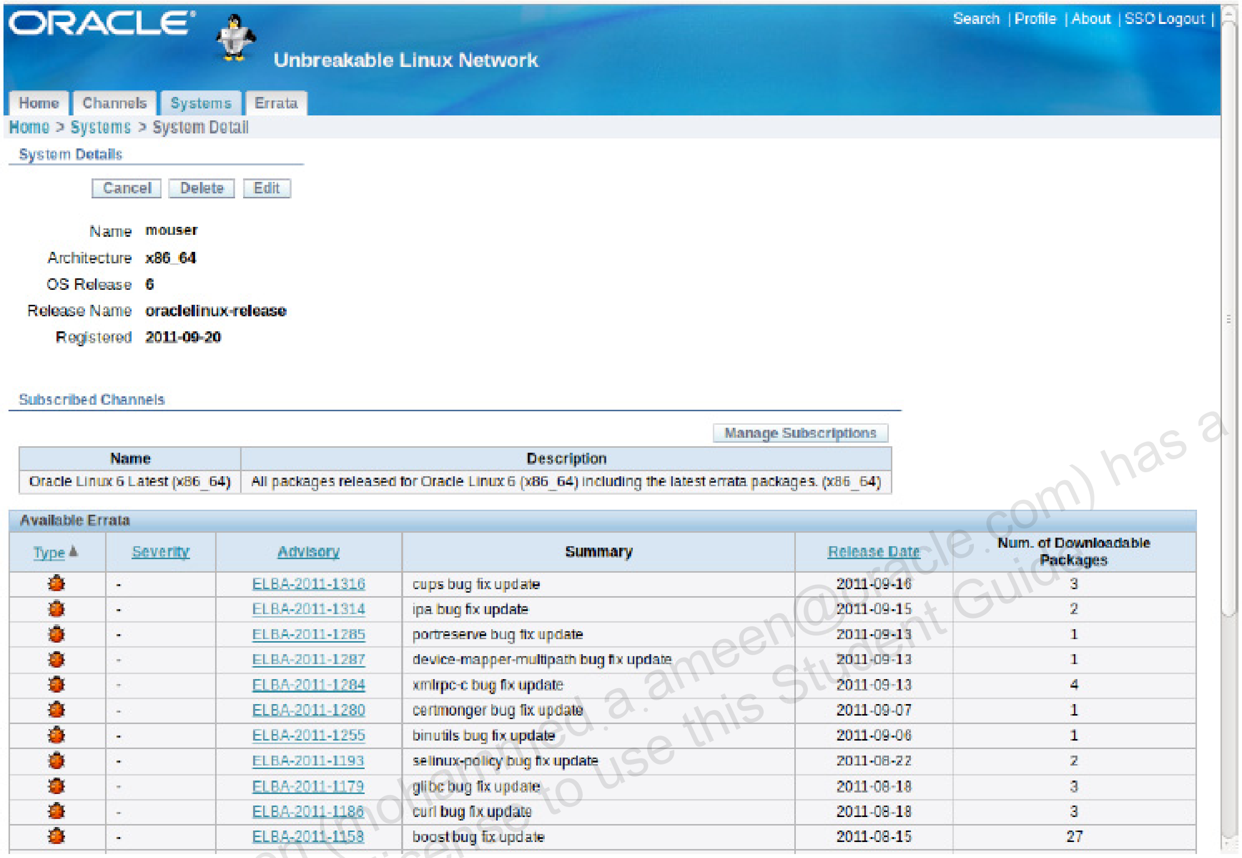
b. Click the **Channels** tab to list of all the channels available via ULN.



Name	Label	Description	Packages
Oracle Linux 6 Latest (x86_64)	ol6_x86_64_latest	All packages released for Oracle Linux 6 (x86_64) including the latest errata packages. (x86_64)	6768
Latest Unbreakable Enterprise Kernel for Oracle Linux 6 (x86_64)	ol6_x86_64_UEK_latest	Latest Unbreakable Enterprise Kernel packages for Oracle Linux 6 (x86_64)	15
Dtrace for Oracle Linux 6 (x86_64) - Latest	ol6_x86_64_Dtrace_latest	Latest packages required for Dtrace on Oracle Linux 6 (x86_64)	11
Unbreakable Enterprise Kernel Release 3 for Oracle Linux 6 (x86_64) - Latest	ol6_x86_64_UEKR3_latest	Latest packages for Unbreakable Enterprise Kernel Release 3 for Oracle Linux 6 (x86_64)	22
Oracle Linux 6 GA (x86_64)	ol6_ga_x86_64_base	All packages released for Oracle Linux 6 GA (x86_64). No errata included.	6010
Oracle Linux 6 GA Patches (x86_64)	ol6_ga_x86_64_patch	Updated packages published after release of Oracle Linux 6 (x86_64)	724
Oracle Linux 6 Dtrace Userspace Tools (x86_64) - Latest	ol6_x86_64_Dtrace_userspace_latest	The latest Dtrace userspace tools for Oracle Linux 6 (x86_64).	2
Oracle Linux 6 Update 1 installation media copy (x86_64)	ol6_u1_x86_64_base	All packages released on the Oracle Linux 6 Update 1 (x86_64) installation media. This channel does not contain updates.	6148
Oracle Linux 6 Update 1 Patch (x86_64)	ol6_u1_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 1 (x86_64).	631
Oracle Linux 6 Update 2 installation media copy (x86_64)	ol6_u2_x86_64_base	All packages released on the Oracle Linux 6 Update 2 (x86_64) installation media. This channel does not contain updates.	6279
Oracle Linux 6 Update 2 Patch (x86_64)	ol6_u2_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 2 (x86_64).	787
Oracle Linux 6 Update 3 installation media copy (x86_64)	ol6_u3_x86_64_base	All packages released on the Oracle Linux 6 Update 3 (x86_64) installation media. This channel does not contain updates.	6324
Oracle Linux 6 Update 3 Patch (x86_64)	ol6_u3_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 3 (x86_64)	743
Oracle Linux 6 Update 4 installation media copy (x86_64)	ol6_u4_x86_64_base	All packages released on the Oracle Linux 6 Update 4 (x86_64) installation media. This channel does not contain updates.	6245
Oracle Linux 6 Update 4 Patch (x86_64)	ol6_u4_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 4 (x86_64)	846
Oracle Linux 6 Update 5 installation media copy (x86_64)	ol6_u5_x86_64_base	All packages released on the Oracle Linux 6 Update 5 (x86_64) installation media. This channel does not contain updates.	6421
Oracle Linux 6 Update 5 Patch (x86_64)	ol6_u5_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 5 (x86_64)	115
HA Utilities for MySQL and Oracle Linux 6 (x86_64)	ol6_x86_64_mysql-ha-utils	Management Utilities for MySQL HA with Oracle Linux 6	2

- For each channel, you can see its detailed description and how many packages are available for download.
- You can also view only the channels for a specific Release and Architecture. The screenshot displays the channels for Oracle Linux 6 x86_64 architecture.

c. Click the **Systems** tab to see registered systems and the number of RPMs available on the subscribed channels to be downloaded and installed.

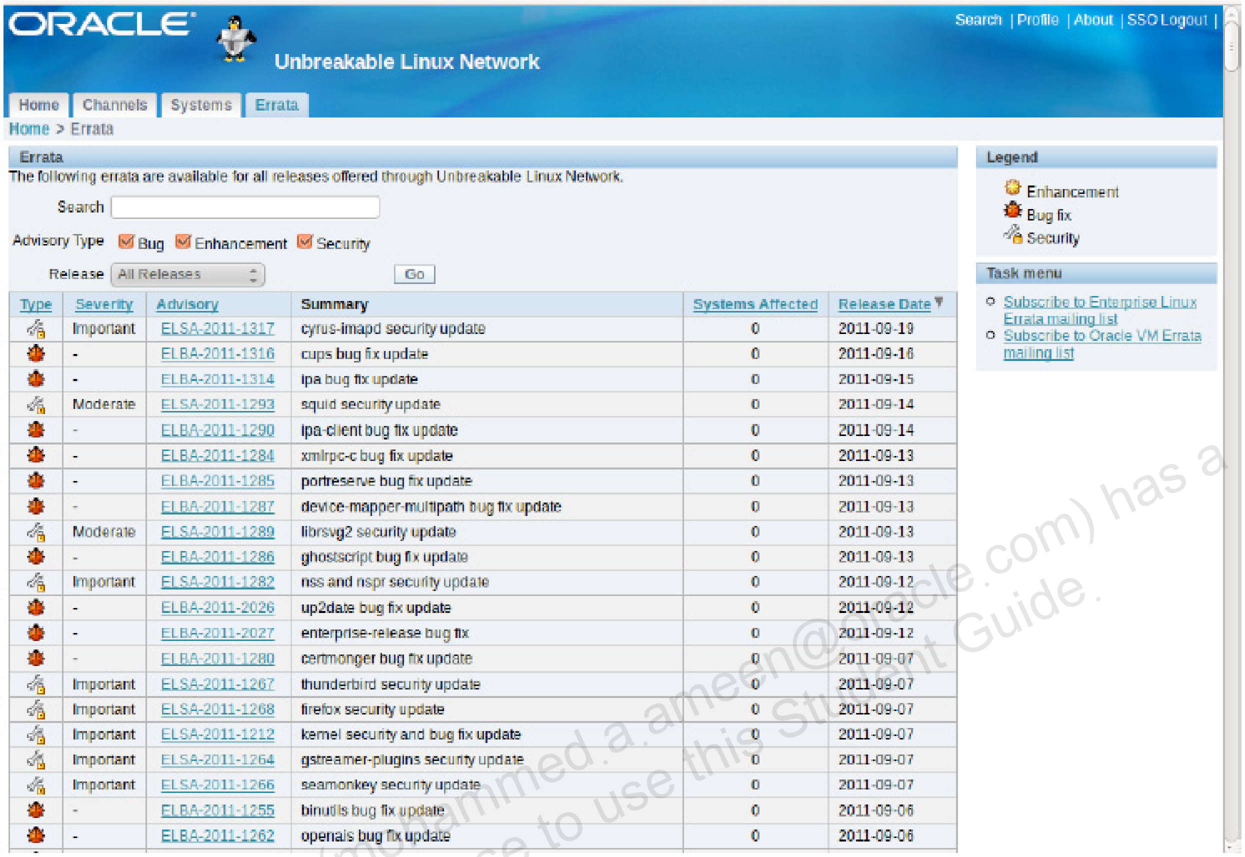


Name	Description
Oracle Linux 6 Latest (x86_64)	All packages released for Oracle Linux 6 (x86_64) including the latest errata packages. (x86_64)

Type	Severity	Advisory	Summary	Release Date	Num. of Downloadable Packages
	-	ELBA-2011-1316	cups bug fix update	2011-09-16	3
	-	ELBA-2011-1314	ipa bug fix update	2011-09-15	2
	-	ELBA-2011-1285	portreserve bug fix update	2011-09-13	1
	-	ELBA-2011-1287	device-mapper-multipath bug fix update	2011-09-13	1
	-	ELBA-2011-1284	xmlrpc-c bug fix update	2011-09-13	4
	-	ELBA-2011-1280	certmonger bug fix update	2011-09-07	1
	-	ELBA-2011-1255	binutils bug fix update	2011-09-06	1
	-	ELBA-2011-1193	selinux-policy bug fix update	2011-08-22	2
	-	ELBA-2011-1179	glibc bug fix update	2011-08-18	3
	-	ELBA-2011-1186	curl bug fix update	2011-08-18	3
	-	ELBA-2011-1158	boost bug fix update	2011-08-15	27

- From this screen, you can also subscribe to additional channels.
- For each channel, you can see a comprehensive listing of each available package. You can also search the list.
- You can get many package details, including a list of the files installed on the system when the RPM is downloaded and a list of the other packages that are necessary for this one to be installed.
- You can also select to download the source RPMs in addition to the binary RPMs.

d. Click the **Errata** tab to list all errata that have been published.



Type	Severity	Advisory	Summary	Systems Affected	Release Date
	Important	ELSA-2011-1317	cyrus-imapd security update	0	2011-09-19
	-	ELBA-2011-1316	cups bug fix update	0	2011-09-16
	-	ELBA-2011-1314	ipa bug fix update	0	2011-09-15
	Moderate	ELSA-2011-1293	squid security update	0	2011-09-14
	-	ELBA-2011-1290	ipa-client bug fix update	0	2011-09-14
	-	ELBA-2011-1284	xmlrpc-c bug fix update	0	2011-09-13
	-	ELBA-2011-1285	portreserve bug fix update	0	2011-09-13
	-	ELBA-2011-1287	device-mapper-multipath bug fix update	0	2011-09-13
	Moderate	ELSA-2011-1289	librsvg2 security update	0	2011-09-13
	-	ELBA-2011-1286	ghostscript bug fix update	0	2011-09-13
	Important	ELSA-2011-1282	nss and nspr security update	0	2011-09-12
	-	ELBA-2011-2026	up2date bug fix update	0	2011-09-12
	-	ELBA-2011-2027	enterprise-release bug fix	0	2011-09-12
	-	ELBA-2011-1280	certmonger bug fix update	0	2011-09-07
	Important	ELSA-2011-1267	thunderbird security update	0	2011-09-07
	Important	ELSA-2011-1268	firefox security update	0	2011-09-07
	Important	ELSA-2011-1212	kernel security and bug fix update	0	2011-09-07
	Important	ELSA-2011-1264	gstreamer-plugins security update	0	2011-09-07
	Important	ELSA-2011-1266	seamonkey security update	0	2011-09-07
	-	ELBA-2011-1255	binutils bug fix update	0	2011-09-06
	-	ELBA-2011-1262	openais bug fix update	0	2011-09-06

Practices for Lesson 7: Ksplice

Chapter 7

Practices for Lesson 7: Ksplice

Practices Overview

In these practices, you do the following:

- View the Ksplice Offline Client Packages from the Unbreakable Linux Network (ULN).
- Upload the Ksplice packages from **dom0** to the **host03** VM.
- Install the Ksplice Offline Client and Kernel Updates to **host03**.
- View the effective kernel version after updates are applied.
- List the active Oracle Ksplice updates in your running kernel.

Practice 7-1: Viewing Ksplice Offline Client Packages from the Unbreakable Linux Network (ULN)

Overview

In this practice, you:

- View the Ksplice Offline Client files available from ULN
- View the `167283.sh` file

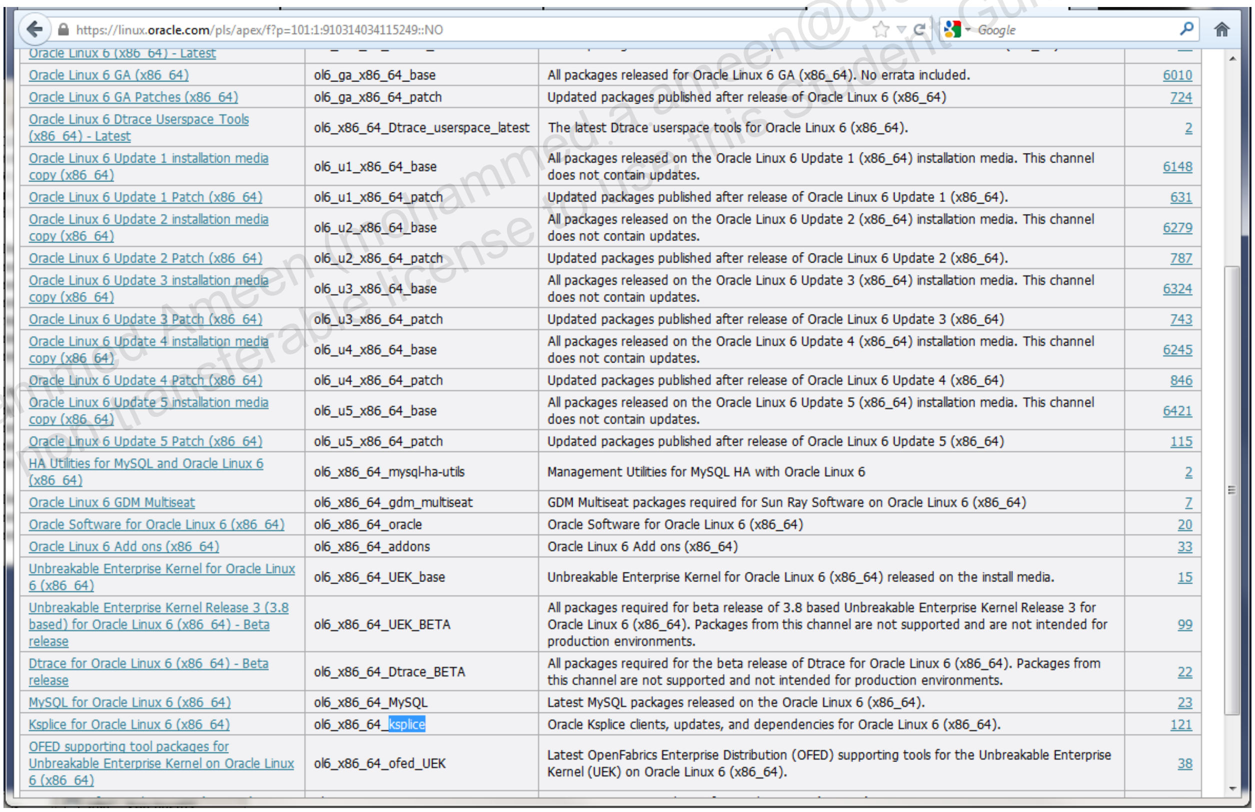
Assumptions

- You are logged on as the `root` user on **dom0**.
- To access ULN, you must be a licensed Oracle customer with an active Oracle Linux support subscription.

Tasks

1. View Ksplice Offline Client files.

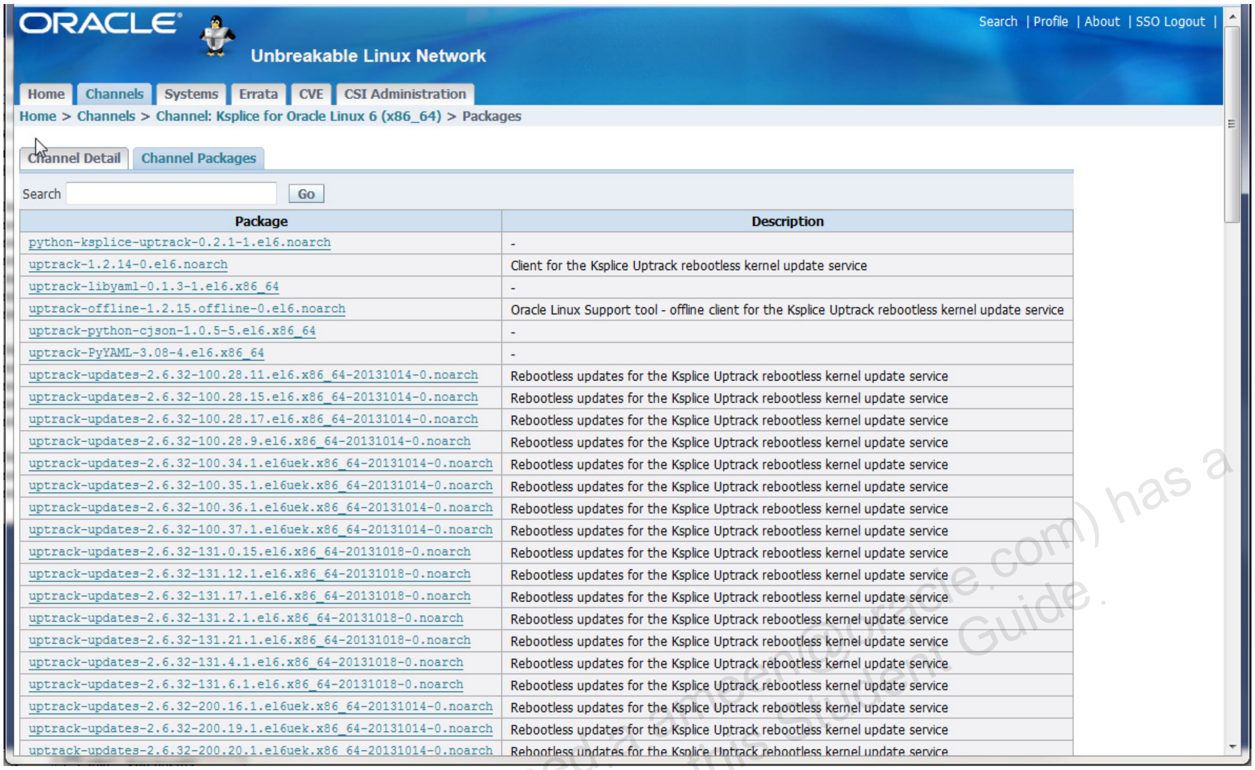
   a. The following screenshot shows ULN channels for Oracle Linux 6 x86_64 architecture.



Oracle Linux 6 (x86_64) - Latest			
Oracle Linux 6 GA (x86_64)	ol6_ga_x86_64_base	All packages released for Oracle Linux 6 GA (x86_64). No errata included.	6010
Oracle Linux 6 GA Patches (x86_64)	ol6_ga_x86_64_patch	Updated packages published after release of Oracle Linux 6 (x86_64)	724
Oracle Linux 6 Dtrace Userspace Tools (x86_64) - Latest	ol6_x86_64_Dtrace_userspace_latest	The latest Dtrace userspace tools for Oracle Linux 6 (x86_64).	2
Oracle Linux 6 Update 1 installation media copy (x86_64)	ol6_u1_x86_64_base	All packages released on the Oracle Linux 6 Update 1 (x86_64) installation media. This channel does not contain updates.	6148
Oracle Linux 6 Update 1 Patch (x86_64)	ol6_u1_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 1 (x86_64).	631
Oracle Linux 6 Update 2 installation media copy (x86_64)	ol6_u2_x86_64_base	All packages released on the Oracle Linux 6 Update 2 (x86_64) installation media. This channel does not contain updates.	6279
Oracle Linux 6 Update 2 Patch (x86_64)	ol6_u2_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 2 (x86_64).	787
Oracle Linux 6 Update 3 installation media copy (x86_64)	ol6_u3_x86_64_base	All packages released on the Oracle Linux 6 Update 3 (x86_64) installation media. This channel does not contain updates.	6324
Oracle Linux 6 Update 3 Patch (x86_64)	ol6_u3_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 3 (x86_64)	743
Oracle Linux 6 Update 4 installation media copy (x86_64)	ol6_u4_x86_64_base	All packages released on the Oracle Linux 6 Update 4 (x86_64) installation media. This channel does not contain updates.	6245
Oracle Linux 6 Update 4 Patch (x86_64)	ol6_u4_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 4 (x86_64)	846
Oracle Linux 6 Update 5 installation media copy (x86_64)	ol6_u5_x86_64_base	All packages released on the Oracle Linux 6 Update 5 (x86_64) installation media. This channel does not contain updates.	6421
Oracle Linux 6 Update 5 Patch (x86_64)	ol6_u5_x86_64_patch	Updated packages published after release of Oracle Linux 6 Update 5 (x86_64)	115
HA Utilities for MySQL and Oracle Linux 6 (x86_64)	ol6_x86_64_mysql-ha-utils	Management Utilities for MySQL HA with Oracle Linux 6	2
Oracle Linux 6 GDM Multiseat	ol6_x86_64_gdm_multiseat	GDM Multiseat packages required for Sun Ray Software on Oracle Linux 6 (x86_64)	7
Oracle Software for Oracle Linux 6 (x86_64)	ol6_x86_64_oracle	Oracle Software for Oracle Linux 6 (x86_64)	20
Oracle Linux 6 Add ons (x86_64)	ol6_x86_64_addons	Oracle Linux 6 Add ons (x86_64)	33
Unbreakable Enterprise Kernel for Oracle Linux 6 (x86_64)	ol6_x86_64_UEK_base	Unbreakable Enterprise Kernel for Oracle Linux 6 (x86_64) released on the install media.	15
Unbreakable Enterprise Kernel Release 3 (3.8 based) for Oracle Linux 6 (x86_64) - Beta release	ol6_x86_64_UEK_BETA	All packages required for beta release of 3.8 based Unbreakable Enterprise Kernel Release 3 for Oracle Linux 6 (x86_64). Packages from this channel are not supported and are not intended for production environments.	99
Dtrace for Oracle Linux 6 (x86_64) - Beta release	ol6_x86_64_Dtrace_BETA	All packages required for the beta release of Dtrace for Oracle Linux 6 (x86_64). Packages from this channel are not supported and not intended for production environments.	22
MySQL for Oracle Linux 6 (x86_64)	ol6_x86_64_MySQL	Latest MySQL packages released on the Oracle Linux 6 (x86_64).	23
Ksplice for Oracle Linux 6 (x86_64)	ol6_x86_64_ksplice	Oracle Ksplice clients, updates, and dependencies for Oracle Linux 6 (x86_64).	121
OFED supporting tool packages for Unbreakable Enterprise Kernel on Oracle Linux 6 (x86_64)	ol6_x86_64_ofed_UEK	Latest OpenFabrics Enterprise Distribution (OFED) supporting tools for the Unbreakable Enterprise Kernel (UEK) on Oracle Linux 6 (x86_64).	38

- Notice the "Ksplice for Oracle Linux 6 (x86_64)" entry towards the bottom of the screen.
- Also notice that this channel contains 121 packages.

b. Clicking the “121” packages link displays the following screen:



Package	Description
python-ksplice-uptrack-0.2.1-1.el6.noarch	-
uptrack-1.2.14-0.el6.noarch	Client for the Ksplice Uptrack rebootless kernel update service
uptrack-libyaml-0.1.3-1.el6.x86_64	-
uptrack-offline-1.2.15.offline-0.el6.noarch	Oracle Linux Support tool - offline client for the Ksplice Uptrack rebootless kernel update service
uptrack-python-cjson-1.0.5-5.el6.x86_64	-
uptrack-PyYAML-3.08-4.el6.x86_64	-
uptrack-updates-2.6.32-100.28.11.el6.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-100.28.15.el6.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-100.28.17.el6.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-100.28.9.el6.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-100.34.1.el6uek.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-100.35.1.el6uek.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-100.36.1.el6uek.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-100.37.1.el6uek.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-131.0.15.el6.x86_64-20131018-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-131.12.1.el6.x86_64-20131018-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-131.17.1.el6.x86_64-20131018-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-131.2.1.el6.x86_64-20131018-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-131.21.1.el6.x86_64-20131018-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-131.4.1.el6.x86_64-20131018-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-131.6.1.el6.x86_64-20131018-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-200.16.1.el6uek.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-200.19.1.el6uek.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service
uptrack-updates-2.6.32-200.20.1.el6uek.x86_64-20131014-0.noarch	Rebootless updates for the Ksplice Uptrack rebootless kernel update service

- This is only a partial list of the 121 packages available from this channel.
- Notice there is an “`uptrack-updates`” RPM for each supported kernel.
- The first six packages are not specific to a kernel version.
- The `uptrack-offline` package is the Ksplice Offline Client.

Note: The `uptrack-offline` package will not install if the `uptrack-1.2.14-0.el6.noarch.rpm` package is already installed.

2. View downloaded packages.

As the `root` user on **dom0**, use the `cd` command to change to the `/OVS/seed_pool/ksplice` directory. Use the `ls` command to display the contents of the directory.

```
[dom0]# cd /OVS/seed_pool/ksplice
[dom0]# ls
167283.sh
uptrack-libyaml-0.1.3-1.el6.x86_64.rpm
uptrack-offline-1.2.15.offline-0.el6.noarch.rpm
uptrack-python-cjson-1.0.5-5.el6.x86_64.rpm
uptrack-PyYAML-3.08-4.el6.x86_64
uptrack-updates-3.8.13-16.2.1.el6uek.x86_64-20131219-
0.noarch.rpm
```

- All files in this directory, with the exception of the `167283.sh` file, have been downloaded from ULN.

- The 167283.sh file has been downloaded from http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/167283.sh.
- The uptrack-offline package and dependencies are provided.
- The package containing kernel updates for the 3.8.13-16.2.1.el6uek.x86_64 kernel is also provided.

3. View the 167283.sh file.

   Use the less command to view the 167283.sh file.

   - Some of the comments in the script are shown as follows:

```
[dom0]# less 167283.sh
#!/bin/sh
...
# yum repository paths
...
# Identify what OS the script is running on.
...
# create the directory for the generated repo files
...
# easiest way to get the list of channel lables ...
...
# determine the correct yum repository directory
...
# Create the required yum repository directory.
...
# figure out which rpms are new
...
# figure out which rpms are there currently
...
# actually get the rpms
...
# run createrepo to generate yum repo metadata
...
```

   - This 167283.sh file is not required for this practice. It is only required to create the Yum repositories for the registered channels.
   - This purpose of this task is just to familiarize you with the contents of this 167283.sh script.

Practice 7-2: Using `sftp` to Upload Ksplice Packages

Overview

In this practice, you perform the following:

- Use `sftp` to upload the Ksplice packages from **dom0** to the **host03** VM.
- The `sftp` command is discussed in Lesson 15 – OpenSSH.

Assumptions

You are the `root` user on **dom0**.

Tasks

1. Use the `sftp` command to transfer the Ksplice packages from **dom0** to **host03**.
   a. From the `/OVS/seed_pool/ksplice` directory on **dom0**, use the `sftp host03` command to connect to **host03** as `root`.
      - Provide `root` password of `oracle` when prompted.

```
[dom0]# sftp host03
root@host03's password: oracle
sftp>
```

   b. Use the `mput *` command to upload all files from the current directory on **dom0** to the `root` user's home directory on **host03**.

```
sftp> mput *
Uploading 167283.sh to /root/167283.sh
...
Uploading uptrack-updates-3.8.13-16.2.1.el6uek.x86_64-20131219-
0.noarch.rpm to /root/uptrack-updates-3.8.13-
16.2.1.el6uek.x86_64-20131219-0.noarch.rpm
...
sftp>
```

   c. Use the `quit` command to exit `sftp`.

```
sftp> quit
```

Practice 7-3: Installing the Ksplice Offline Client and Kernel Updates

Overview

In this practice, you perform the following:

- Install the Ksplice Offline Client packages on **host03**.
- Install the kernel updates package on **host03**.
- View the effective kernel version after updates have been installed.
- List the active Oracle Ksplice updates in your running kernel.

Assumptions

You are the `root` user on **host03**.

Tasks

1. Ensure you have the correct version of the Ksplice Uptrack kernel update.
   a. From **host03**, use the `uname -r` command to display the version of the running kernel.

```
# uname -r
3.8.13-16.2.1.el6uek.x86_64
```

   b. Use the `ls` command to display the contents of the directory.

```
# cd
# ls
...
uptrack-updates-3.8.13-16.2.1.el6uek.x86_64-20131219-
0.noarch.rpm
```

   - Notice the kernel version on **host03** is the same as the Ksplice Uptrack kernel package, `3.8.13-16.2.1.el6uek.x86_64`.

2. Install the Ksplice Offline Client packages.
   a. Use the `rpm -Uvh` command to install the `uptrack-offline` package.

```
# rpm -Uvh uptrack-offline-1.2.15.offline-0.el6.noarch.rpm
error: Failed dependencies:
        uptrack-PyYAML is needed by uptrack-offline-
1.2.15.offline-0.el6.noarch
        uptrack-python-cjson is needed by uptrack-offline-
1.2.15.offline-0.el6.noarch
```

   - Notice the install failed because two dependent packages are missing.

   b. Use the `rpm -Uvh` command to install the `uptrack-PyYAML` package.

```
# rpm -Uvh uptrack-PyYAML-3.08-4.el6.x86_64.rpm
error: Failed dependencies:
        uptrack-libyaml >= 0.1.3-1 is needed by uptrack-
PyYAML...
```

   - Notice the install failed because the `uptrack-libyaml` package is not installed.

c. Use the `rpm -Uvh` command to install the `uptrack-libyaml` package.

```
# rpm –Uvh uptrack-libyaml-0.1.3-1.el6.x86_64.rpm
Preparing...          ################################### [100%]
   1:uptrack-libyaml ################################### [100%]
```

- This package installed successfully with no dependency errors.

d. Perform Task 2b to install the `uptrack-PyYAML` package.

```
# rpm –Uvh uptrack-PyYAML-3.08-4.el6.x86_64.rpm
Preparing...          ################################### [100%]
   1:uptrack-PyYAML  ################################### [100%]
```

- This package installed successfully with no dependency errors.

e. Use the `rpm -Uvh` command to install the `uptrack-python-cjson` package.

```
# rpm –Uvh uptrack-python-cjson-1.0.5-5.el6.x86_64.rpm
Preparing...               ############################## [100%]
   1:uptrack-python-cjson ############################## [100%]
```

- This package installed successfully with no dependency errors.

f. Perform Task 2a to install the `uptrack-offline` package.

```
# rpm –Uvh uptrack-offline-1.2.15.offline-0.el6.noarch.rpm
Preparing...          ################################### [100%]
   1:uptrack-offline ################################### [100%]
There are no existing modules on disk that need basename
migration.
```

- All Ksplice Offline Client packages are now installed.

3. Install the kernel updates package.

Use the `rpm -Uvh` command to install the `uptrack-updates` package.

```
# rpm –Uvh uptrack-updates-3.8.13-16.2.1.el6uek.x86_64-20131219-
0.noarch.rpm
Preparing...                 ############################ [100%]
   1:uptrack-updates-3.8.13-############################ [100%]
The following steps will be taken:
Install [27mo8t1x] CVE-2013-4387: Memory corruption in IPv6 ...
Install [spxsnial] CVE-2013-0342: Denial of service in IPv6 ...
...
Installing [rz5tgzrd] Incorrect NumVFs reporting in PCI I/O ...
Your kernel is fully up to date.
Effective kernel version is 3.8.13-16.3.1.el6uek
```

- All kernel updates are installed.

4. Show effective kernel.

   a. Use the `uname -r` command to display the version of the running kernel.

```
# uname –r
3.8.13-16.2.1.el6uek.x86_64
```

   - Notice the kernel version remains at version `3.8.13-16.2.1.el6uek`.

   b. Use the `uptrack-uname -r` command to display the effective kernel version.

```
# uptrack-uname –r
3.8.13-16.3.1.el6uek.x86_64
```

   - Notice the effective kernel version is `3.8.13-16.3.1.el6uek`.

   c. Use the `uptrack-show` command to list the active Oracle Ksplice updates in your running kernel.

```
# uptrack-show
Installed updates:
[27mo8t1x] CVE-2013-4387: Memory corruption in IPv6 UDP ...
[spxsnial] CVE-2013-0342: Denial of service in IPv6 privacy ...
...
[rz5tgzrd] Incorrect NumVFs reporting in PCI I/O Single Root ...

Effective kernel version is 3.8.13-16.3.1.el6uek
```

Practices for Lesson 8: Automating Tasks

Chapter 8

Practices for Lesson 8: Automating Tasks

Practices Overview

In these practices, you use and configure utilities to automate tasks.

Practice 8-1: Automating Tasks

Overview

In this practice, you use the `crontab` utility and the `at` utility to automate tasks. You also prevent the `oracle` user from being able to use the `at` utility.

Assumptions

You are the `root` user on the **host03** VM.

Tasks

1. Create a `crontab` for the `root` user.
   a. Use the `crontab -l` command to list the contents of your `crontab` file.

```
# crontab -l
no crontab for root
```

   b. Use the `ls` command to view the contents of the `/var/spool/cron` directory.

```
# ls /var/spool/cron
```

   - Notice that there are no files in this directory.

   c. Use the `crontab -e` command to create a `crontab` job that runs the `ls` command every minute.

```
# crontab -e
* * * * * ls
```

   - The `crontab -e` command uses the `vi` editor. Save your changes and exit `vi`. The following messages will be displayed:

```
no crontab for root - using an empty one
crontab: installing new crontab
```

   d. Use the `ls` command to view the contents of the `/var/spool/cron` directory.

```
# ls /var/spool/cron
root
You have new mail in /var/spool/mail/root
```

   - Now there is a `root` file in the `/var/spool/cron` directory.
   - You also have mail, because the output from `cron` jobs is sent to the user's mailbox.

   e. Use the `cat` command to display the contents of the `/var/spool/cron/root` file.

```
# cat /var/spool/cron/root
* * * * * ls
```

f. Use the `mail` command to view the results of your job.

```
# mail
...
>N 1 Cron Daemon  Tue Dec 13 07:17  23/726  "Cron <root@host03>
ls"
 N 2 Cron Daemon  Tue Dec 13 07:18  23/726  "Cron <root@host03>
ls"
 N 3 Cron Daemon  Tue Dec 13 07:19  23/726  "Cron <root@host03>
ls"
&
```

- To view details of mailbox entries, press the associated number and then press **Enter**.
- To redisplay the header, press **h** and then press **Enter**.
- To quit the mail program, press **q** and then press **Enter**.

g. Press **q** and then **Enter** to quit the mail program.

```
& q
```

h. Use the `crontab -r` to remove the `crontab`.

```
# crontab -r
# crontab -l
no crontab for root
```

i. Use the `ls` command to view the contents of the `/var/spool/cron` directory.

```
# ls /var/spool/cron
```

- Notice that the `root` file has been removed from this directory.

2. Create a `crontab` for the `oracle` user.
   - In this task, you switch back and forth between the `root` user and the `oracle` user.

a. Open another tab in your terminal window by pressing **Shift + Ctrl + T**, or right-click and select **Open Tab**.
   - Your terminal window now has a tab where you are logged in as `root` and a tab where you are logged in as the `oracle` user:

b. Click the **oracle@host03** tab and confirm that you are logged in as the `oracle` user.

```
[oracle@host03 ~]$ whoami
oracle
```

c. Use the `crontab -l` command to list the contents of your `crontab` file.

```
[oracle@host03 ~]$ crontab -l
no crontab for oracle
```

d. Use the `crontab -e` command to create a `cron` job that runs the `echo "Hello World"` command every other minute.

```
[oracle@host03 ~]$ crontab -e
*/2 * * * * echo "Hello World"
```

e. Use the `crontab -l` command to list the contents of your `crontab` file.

```
[oracle@host03 ~]$ crontab -l
*/2 * * * * echo "Hello World"
```

f. Use the `mail` command to view the results of your job.

```
[oracle@host03 ~]$ mail
...
>N 1 Cron Daemon  Tue Dec 13 08:40  21/728  "Cron
<oracle@host03> "
 N 2 Cron Daemon  Tue Dec 13 08:42  23/726  "Cron
<oracle@host03> "
&
```

g. To view details of mailbox entries, press the associated number and then press **Enter**.

h. After viewing the details, press **q** and then **Enter** to quit.

```
& 1
...
Hello World
& q
```

i. Use the `ls` command to view the contents of the `/var/spool/cron` directory.

```
[oracle@host03 ~]$ ls /var/spool/cron
ls: cannot open directory /var/spool/cron/: Permission denied
```

- Notice that only the `root` user has permission to view this directory.

j. Click the **root@host03** tab and confirm that you are logged in as the `root` user.

```
[root@host03 ~]# whoami
root
```

k. Use the `ls` command to view the contents of the `/var/spool/cron` directory.

```
[root@host03 ~]# ls /var/spool/cron
oracle
```

- Notice that there is an `oracle` file in the `/var/spool/cron` directory.

l. Use the `cat` command to display the contents of the `oracle` user's `crontab`.

```
[root@host03 ~]# cat /var/spool/cron/oracle
*/2 * * * * echo "Hello World"
```

m. Use `crontab –r` to remove the `crontab` for the `oracle` user.

```
[root@host03 ~]# crontab –u oracle –r
[root@host03 ~]# crontab –u oracle –l
no crontab for oracle
```

3. Use the at command to schedule a one-time task at a specific time.

   a. Click the **oracle@host03** tab and confirm that you are logged in as the oracle user.

```
[oracle@host03 ~]$ whoami
oracle
```

   b. Use at to run the pwd command in 1 minute.

```
[oracle@host03 ~]$ at now + 1 minute
at> pwd
at> <EOT>
```

   - Press **Ctrl + D** for <EOT>.

   c. Use the atq command to list the pending job.

```
[oracle@host03 ~]$ atq
1      2013-11-15  11:19 a oracle
```

   d. After 1 minute has passed, run the atq command again.

```
[oracle@host03 ~]$ atq
```

   - Notice that this time there are no pending jobs.

   e. Use the mail command to view the results of your job.

```
[oracle@host03 ~]$ mail
...
>N 2 Oracle Student Tue Dec 13 10:16 14/509  "Output from your
job"
&
```

   f. To view details of mailbox entries, press the associated number and then press **Enter**.

   g. After viewing the details, press **q** and then **Enter** to quit.

```
& 2
...
/home/oracle
& q
```

4. Restrict usage of the at command.

   - In this task, you also switch back and forth between the root user and the oracle user.

   a. Click the **root@host03** tab and confirm that you are logged in as the root user.

```
[root@host03 ~]# whoami
root
```

   b. Use the vi editor to add user oracle to the /etc/at.deny file.

```
[root@host03 ~]# vi /etc/at.deny
oracle
```

c. Click the **oracle@host03** tab and confirm that you are logged in as the `oracle` user.

```
[oracle@host03 ~]$ whoami
oracle
```

d. Attempt to use `at` to run a command in 1 minute.

```
[oracle@host03 ~]$ at now + 1 minute
You do not have permission to use at.
```

5. In the terminal window, click the **X** in the **oracle@host03** tab to close this tab.

Practices for Lesson 9: Kernel Module Configuration

Chapter 9

Practices for Lesson 9: Kernel Module Configuration

Practices Overview

In these practices, you work with loadable kernel modules.

Practice 9-1: Using Loadable Kernel Modules

Overview

In this practice, you perform the following tasks:

- Use kernel module utilities to list modules.
- Get detailed information about modules.
- Load and unload kernel modules.
- Explore the kernel module configuration directory.

Assumptions

- You are the `root` user on the **host03** VM.
- Sample output is provided. Kernel modules shown might differ from actual output.

Tasks

1. List kernel modules.
   a. Use the `lsmod` command to list kernel modules currently loaded into the kernel.
      - You might see numerous Ksplice modules as a result of completing Practice 7-3.
      - These Ksplice modules are not included in the sample output.

```
# lsmod
Module                  Size    Used by
nls_utf8                1421    1
fuse                   78015    0
autofs4                33910    3
...
nf_conntrack           84594    3 nf_conntrack_ipv4,...
...
```

      - The "Size" column displays the amount of memory the module uses.
      - The "Used by" column gives the total number of processes that are using the module and other modules that it depends on, followed by a list of those dependent modules.
      - In the sample output, `nf_conntrack` depends on modules `nf_conntrack_ipv4`, `nf_conntrack_ipv6`, and other listed modules being loaded before `nf_conntrack` loads.

   b. Use the `modinfo` command to list detailed information about a specific kernel module (for example, `nls_utf8`).

```
# modinfo nls_utf8
filename:   /lib/modules/3.8.13-
16.2.1.el6uek.x86_64/kernel/fs/nls/nls_utf8.ko
license:    Dual BSD/GPL
srcversion:
...
```

- Notice that the kernel modules are loaded from the `/lib/modules/<kernel_version>/kernel` directory.

c. Use the `uname -r` command to display the kernel version.

```
# uname -r
3.8.13-16.2.1.el6uek.x86_64
```

d. List the kernel modules for the kernel version.

```
# ls -R /lib/modules/`uname -r`/kernel
arch
crypto
Documentation
drivers
fs
kernel
lib
mm
net
sound
...
```

- The actual kernel modules have a `.ko` (kernel object) extension.

2. Load kernel modules.

a. Determine if the **nfs** kernel module is currently loaded.

```
# lsmod | grep nfs
```

- In this example, **nfs** is not loaded.

b. Use the `modprobe` command to load the **nfs** kernel module.

- Sample output is provided. Yours might be different with Ksplice kernel updates applied.

```
# modprobe -v nfs
...
insmod /lib/modules/3.8.13-
16.2.1.el6uek.x86_64/kernel/net/sunrpc/sunrpc.ko
insmod /lib/modules/3.8.13-
16.2.1.el6uek.x86_64/kernel/fs/lockd/lockd.ko
insmod /lib/modules/3.8.13-
16.2.1.el6uek.x86_64/kernel/fs/fscache/fscache.ko
insmod /lib/modules/3.8.13-
16.2.1.el6uek.x86_64/kernel/fs/nfs/nfs.ko
```

- The preceding example includes the `-v` (verbose) option.
- Notice that the dependent modules are loaded prior to loading the **nfs** module.
- Also notice that `modprobe` uses the `insmod` command to load the modules.

c. Confirm that the **nfs** module is loaded.

- Sample output is provided. Yours might be different with Ksplice kernel updates applied.

```
# lsmod | grep nfs
nfs                184416      0
fscache             52962      1    nfs
lockd               84481      1    nfs
sunrpc             267672      3    nfs,lockd
```

3. List module dependencies.

- Kernel module dependencies are listed in `/lib/modules/<kernel_version>/modules.dep`.

Use the `grep` command to list the kernel module dependencies for **nfs**.

```
# grep nfs.ko /lib/modules/`uname -r`/modules.dep
kernel/fs/nfs/nfs.ko: kernel/fs/fscache/fscache.ko
kernel/fs/lockd/lockd.ko kernel/net/sunrpc/sunrpc.ko
...
```

- Notice that the dependencies listed in the **`modules.dep`** file for **nfs** correspond with the modules loaded when running the `modprobe nfs` command.

4. Unload kernel modules.

Use the `modprobe -r` command to unload the **nfs** kernel module.

```
# modprobe -rv nfs
rmmod /lib/modules/3.8.13-
16.2.1.el6uek.x86_64/kernel/fs/nfs/nfs.ko
rmmod /lib/modules/3.8.13-
16.2.1.el6uek.x86_64/kernel/fs/fscache/fscache.ko
rmmod /lib/modules/3.8.13-
16.2.1.el6uek.x86_64/kernel/fs/lockd/lockd.ko
...
rmmod /lib/modules/3.8.13-
16.2.1.el6uek.x86_64/kernel/net/sunrpc/sunrpc.ko
```

- The preceding example includes the `-v` (verbose) option.
- Notice that `modprobe -r` uses the `rmmod` command to remove the modules.
- Also notice that the dependent modules are removed when possible.
- In some cases, a module cannot be removed because it is a dependent module for other kernel modules.

5. Explore the `/etc/sysconfig/modules` directory.

- This directory contains files that specify kernel modules to be loaded at boot time.

a. List the contents of the `/etc/sysconfig/modules` directory.

```
# ls -l /etc/sysconfig/modules
...
-rwxr-xr-x.   bluez-uinput.modules
```

- Files in this directory must end in `.modules`.

b. View the contents of the `.modules` file.

```
# cat /etc/sysconfig/modules/bluez-uinput.modules
#!/bin/sh
if [ ! -c /dev/input/uinput ]; then
   exec /sbin/modprobe uinput
fi
```

- Notice that the `.modules` files are executable shell scripts.

6. Explore the `/etc/modprobe.d` directory.
   - Files in this directory end with `.conf` and are used for specifying the options that are to be used with kernel modules.

a. List the contents of the `/etc/modprobe.d` directory.

- Sample output is shown, your output might be different.

```
# ls -l /etc/modprobe.d
-rw-r--r--.    anaconda.conf
-rw-r--r--.    blacklist.conf
-rw-r--r--.    dist-alsa.conf
-rw-r--r--.    dist.conf
-rw-r--r--.    dist-oss.conf
-rw-r--r--.    openfwwf.conf
```

b. Use the `grep` command to search for the occurrence of **alias** in the `dist.conf` file.

```
# grep alias /etc/modprobe.d/dist.conf
...
alias nfs4 nfs
alias rpc_pipefs sunrpc
...
```

- The **alias** command is used to create alternative names for kernel modules.
- The alias name is the first argument; kernel module name is the second argument.

c. Load the **nfs** module by using the **nfs4** alias name.

```
# modprobe nfs4
# lsmod | grep nfs
nfs               184416      0
fscache            52962      1    nfs
lockd              84481      1    nfs
sunrpc            267672      3    nfs,lockd
```

d. Unload the **nfs** module by using the **nfs4** alias name.

```
# modprobe -r nfs4
# lsmod | grep nfs
```

e. Use the `grep` command to search for the occurrence of **options** in the `openfwwf.conf` file.

```
# grep options /etc/modprobe.d/openfwwf.conf
options b43 nohwcrypt=1 qos=0
```

- Notice the **nohwcrypt** and **qos** options for the **b43** kernel module.

f. Load the **b43** kernel module.

```
# modprobe b43
```

g. Use the `modinfo` command to list detailed information about the **b43** kernel module.

```
# modinfo b43
...
parm:    nohwcrypt:Disable hardware encryption. (int)
...
parm:    qos:Enable QOS support (default on) (int)
...
```

- Notice the **nohwcrypt** and **qos** parameters for the **b43** kernel module.

h. Unload the **b43** kernel module.

```
# modprobe -r b43
```

i. Use the `grep` command to search for the occurrence of **install** in the `dist.conf` file.

```
# grep install /etc/modprobe.d/dist.conf
...
install net-pf-3 /bin/true
...
```

- For the **net-pf-3** kernel module, the `/bin/true` command runs instead of loading the kernel module when using `modprobe`.

j. Determine if the **net-pf-3** kernel module is loaded Attempt to load the **net-pf-3** kernel module. Then determine if the module loaded.

```
# lsmod | grep net-pf-3
# modprobe net-pf-3
# lsmod | grep net-pf-3
```

- Notice that the **net-pf-3** kernel module did not load because of the setting in the `/etc/ modprobe.d/dist.conf` file.

Practices for Lesson 10: User and Group Administration

Chapter 10

Practices for Lesson 10: User and Group Administration

Practices Overview

In these practices, you:

- Create user and group accounts
- Understand the benefits of user private groups
- Configure password aging
- Use the User Manager GUI
- Restrict the use of the `su` command
- Allow the use of the `sudo` command

Practice 10-1: User Account Administration

Overview

In this practice, you use command-line utilities to create new user accounts, view files that are updated when adding a new user, modify a user account, set a password for the new user, and log in as the new user.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Add a user.
   a. Use the command-line utility to add `student1` user.

```
# useradd student1
```

   b. Use the `cat` command to view the new `student1` entry in the `/etc/passwd` file.

```
# cat /etc/passwd
...
oracle:x:500:500:Oracle Student:/home/oracle:/bin/bash
student1:x:501:501::/home/student1:/bin/bash
```

   - Notice that the new user's UID and GID incremented by one.
   - Notice that a home directory was created for the new user (`/home/student1`).
   - Notice that the default shell for the new user is **bash** (`/bin/bash`).

   c. Use the `ls` command to verify that the new user's home directory was created.

```
# ls /home
lost+found  oracle  student1
```

   - A home directory was created because **`CREATE_HOME`** in `/etc/login.defs` is set to yes.

   d. View the value of **`CREATE_HOME`** in `/etc/login.defs`.

```
# grep CREATE_HOME /etc/login.defs
CREATE_HOME   yes
```

   e. View the default settings for a new user, stored in `/etc/default/useradd`.

```
# cat /etc/default/useradd
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel
CREATE_MAIL_SPOOL=yes
```

   - Notice that the **`SKEL`** directive is set to `/etc/skel`.

f. View the contents of /etc/skel.

```
# ls -la /etc/skel
-rw-r--r--.  .bash_logout
-rw-r--r--.  .bash_profile
-rw-r--r--.  .bashrc
drwxr-xr-x.  .gnome2
drwxr-xr-x.  .mozilla
```

g. View the contents of the new user's home directory.

```
# ls -la /home/student1
-rw-r--r--.  .bash_logout
-rw-r--r--.  .bash_profile
-rw-r--r--.  .bashrc
drwxr-xr-x.  .gnome2
drwxr-xr-x.  .mozilla
```

- Notice that the contents of **SKEL** (/etc/skel) are copied to the new user's home directory.

h. View the new student1 entry in the /etc/group file.

```
# cat /etc/group
...
oracle:x:500:
student1:x:501:
```

- Because Oracle Linux uses a user private group (UPG) scheme, a new private group (student1, GID=501) was created when the user student1 was created.

i. View the new student1 entry in the /etc/shadow file.

```
# cat /etc/shadow
...
oracle:$6$...:16048:0:99999:7:::
student1:!!:16049:0:99999:7:::
```

j. View the new student1 entry in the /etc/gshadow file.

```
# cat /etc/gshadow
...
oracle:!!::
student1:!::
```

k. Add a new user with the following characteristics:

- Username=student2
- UID=555
- GECOS information="Oracle Student2"
- Default shell=/bin/sh (Bourne shell)

```
# useradd -u 555 -c "Oracle Student2" -s /bin/sh student2
```

l. View the new student2 entry in /etc/passwd.

```
# tail -2 /etc/passwd
student1:x:501:501::/home/student1:/bin/bash
student2:x:555:555:Oracle Student2:/home/student2:/bin/sh
```

- Notice that the UID and GID are 555.
- Notice the GECOS information.
- Notice the default shell.

2. Modify a user account.
   a. Use the usermod command to modify GECOS information for student1 user as follows:

```
# usermod –c "Oracle Student1" student1
```

   b. View the student1 entry in the /etc/passwd file.

```
# grep student1 /etc/passwd
student1:x:501:501:Oracle Student1:/home/student1:/bin/bash
```

3. Assign a password to the new user.
   a. Use the cat command to view the /etc/shadow file.

```
# cat /etc/shadow
...
oracle:$6$...:16048:0:99999:7:::
student1:!!:16049:0:99999:7:::
student2:!!:16049:0:99999:7:::
```

- Notice the "!!" in the student1 and student2 records, indicating that no password has been assigned (and that the accounts are locked).

   b. Use the passwd command to create a password (of password) for the student1 user.

```
# passwd student1
Changing password for user student1.
New password: password
BAD PASSWORD: it is based on a dictionary word
Retype new password: password
passwd: all authentication tokens updated successfully.
```

- Ignore the "BAD PASSWORD" warning, continuing to use password as the password.

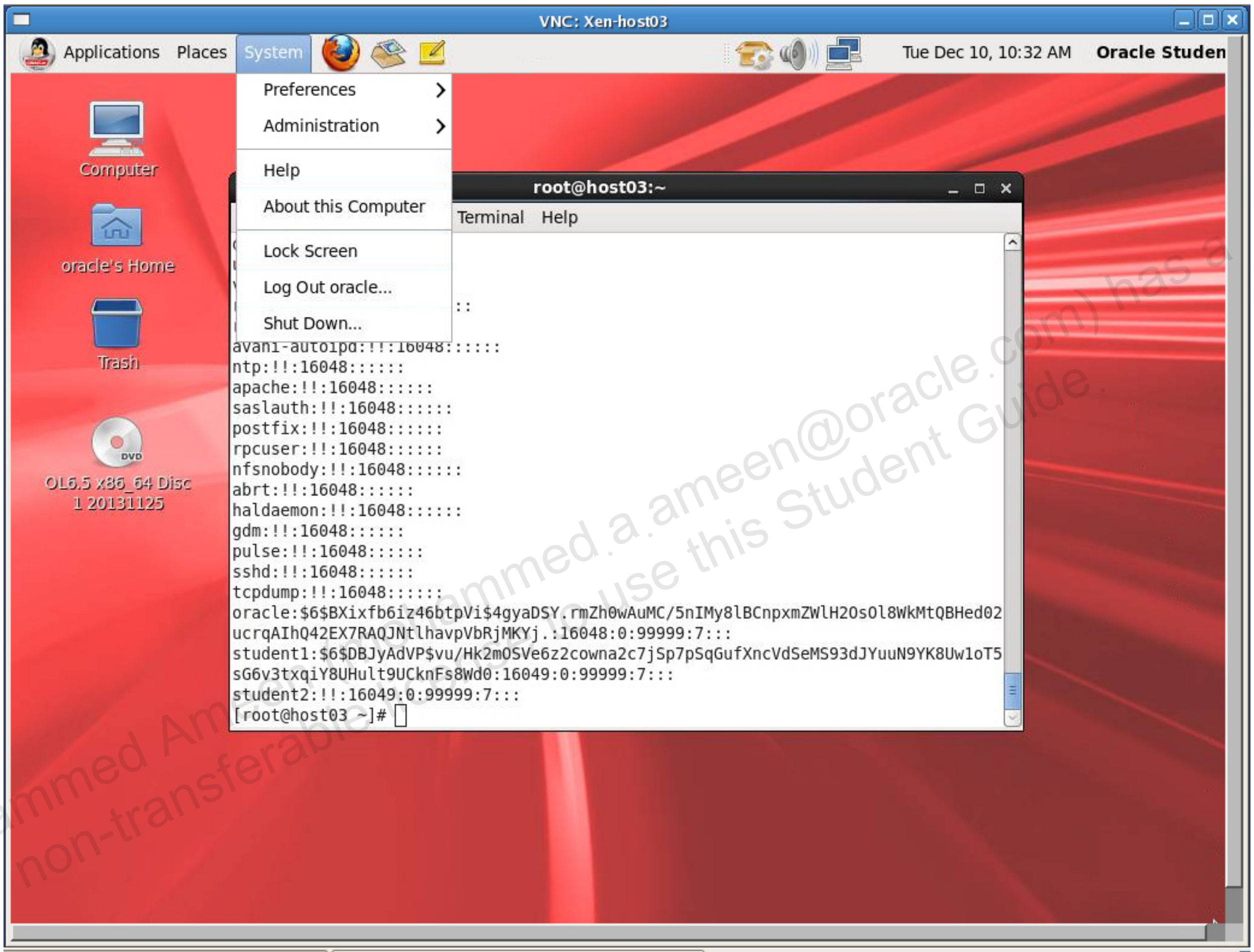
   c. View the /etc/shadow file.

```
# cat /etc/shadow
...
oracle:$6$...:16048:0:99999:7:::
student1:$6$...: 16049:0:99999:7:::
student2:!!: 16049:0:99999:7:::
```

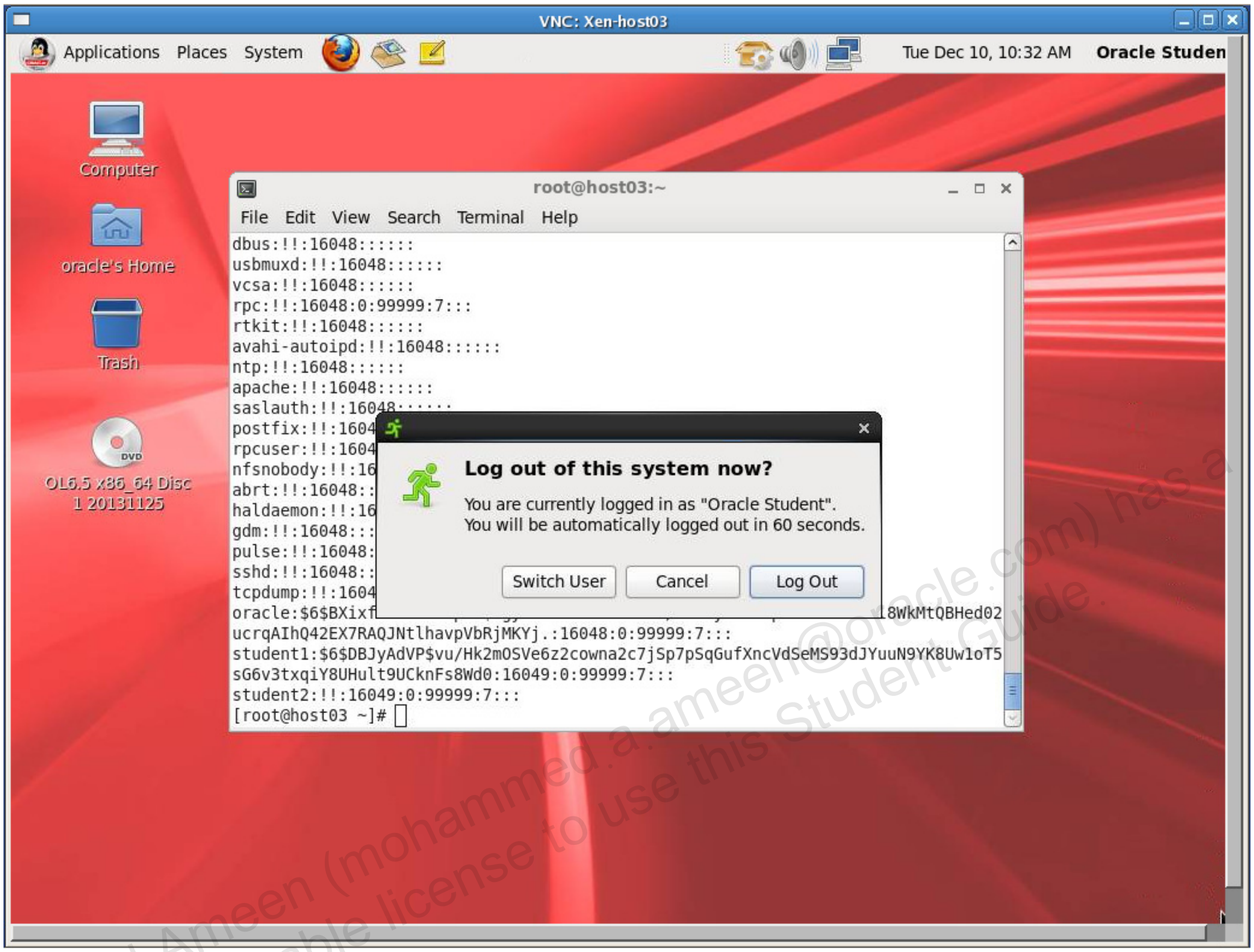
- Now the "`!!`" for `student1` has been replaced with a hashed password value.
- Notice that the `student2` account is still locked because a password has not been assigned.

4. Log in as new user.

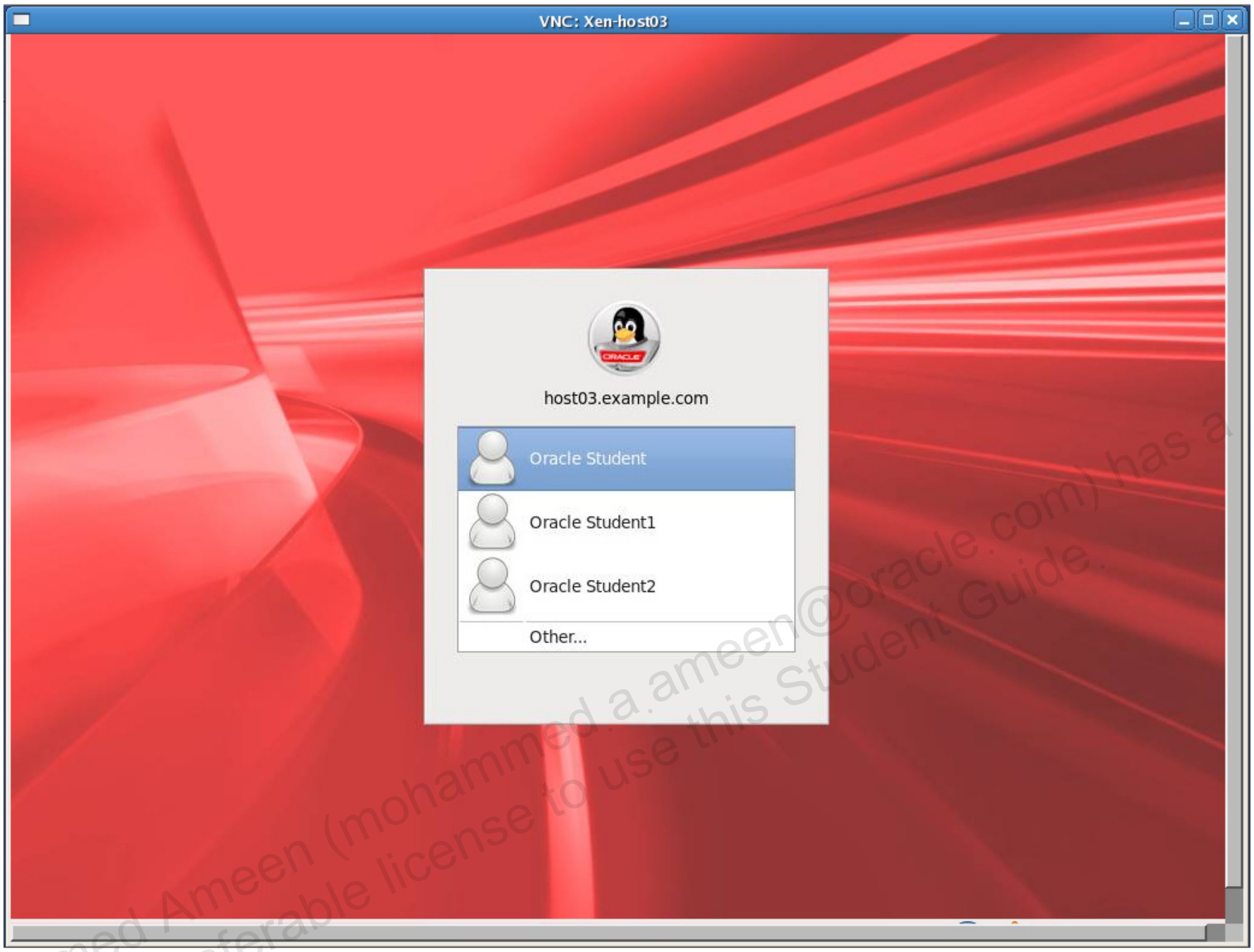
   a. Log out as user `oracle` by selecting **Log out oracle** from the **System** menu, as shown below.



b. Click **Log Out** in the pop-up, as shown below.



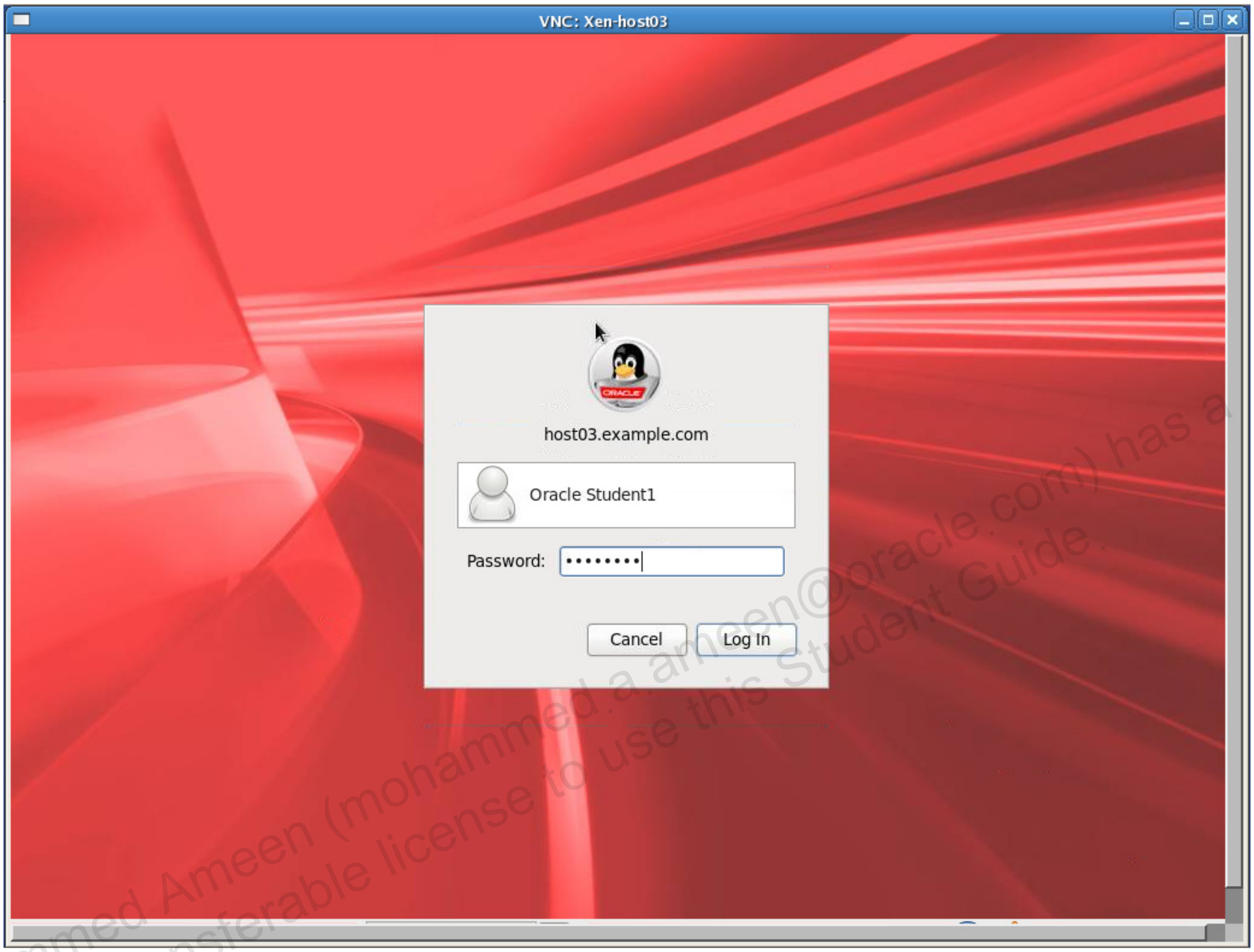
c. Attempt to log in as `student2` by selecting `Oracle Student2` from the menu, as shown below.



- You are prompted for a password, but because no password has been assigned to `student2`, the account is locked.
- Regardless of what you enter for a password, the following screen appears, and you cannot log in as `student2`.

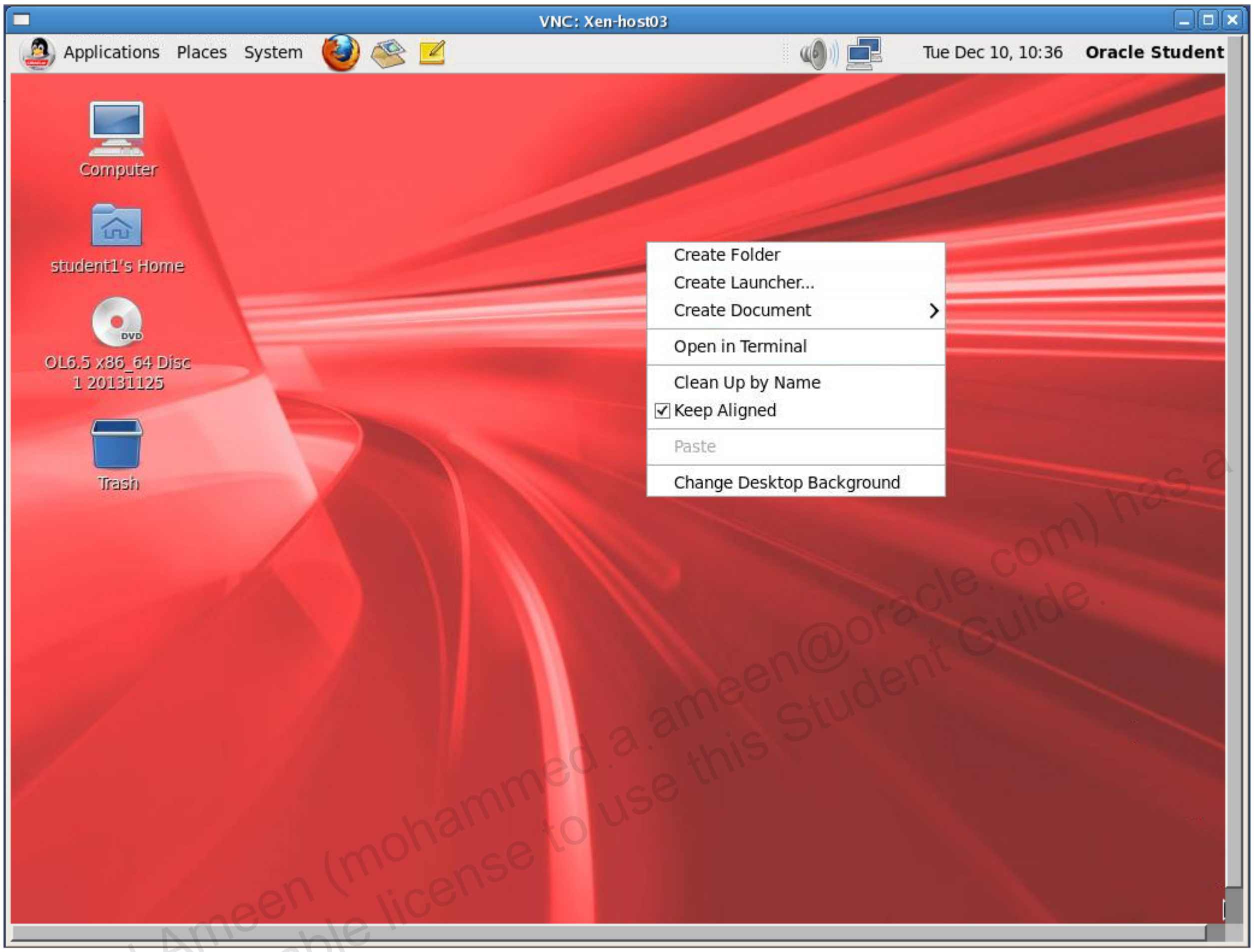
d. Log in as `student1` by selecting `Oracle Student1` from the login menu.

e. Enter `password` when prompted for the password, as shown below.



f. Click **Log In** to successfully log in as `student1`.

g. Right-click the desktop to display the pop-up menu as shown below.



h. Select **Open in Terminal** to open a terminal window.

i. From the terminal window, use the `whoami` command to verify that you are logged in as `student1`.

```
$ whoami
student1
```

j. Use the `su -` command to become the `root`. The password is `oracle`.

```
$ su -
Password: oracle
# whoami
root
```

Practice 10-2: Group Account Administration

Overview

In this practice, you create a new group account and add users to this new group.

Tasks

1. Add a group.
   a. Use the `groupadd` command without any options to add the `students` group.

   ```
   # groupadd students
   ```

   b. View the new `students` entry in the `/etc/group` file.

   ```
   # cat /etc/group
   ...
   oracle:x:500:
   student1:x:501:
   student2:x:555:
   students:x:556:
   ```

   - Notice that the GID for the new group is incremented by one.

2. Add users to the new group.
   a. Use the `usermod` command to add users `student1` and `student2` to the `students` group.

   ```
   # usermod –aG 556 student1
   # tail -1 /etc/group
   students:x:556:student1
   # usermod –aG 556 student2
   # tail -1 /etc/group
   students:x:556:student1,student2
   ```

   - Notice that both `student1` and `student2` have a secondary group membership in the `students` group.

   b. View the primary group membership for `student1` and `student2`.

   ```
   # grep student /etc/passwd
   student1:x:501:501:Oracle Student1:/home/student1:/bin/bash
   student2:x:555:555:Oracle Student2:/home/student2:/bin/sh
   ```

   - Notice that `student1` primary group is still `501`.
   - Notice that `student2` primary group is still `555`.

Practice 10-3: User Private Groups

Overview

In this practice, you use the UPG scheme to give different users write access to files in a single directory.

Tasks

1. Set up a shared /students directory.

   a. Use the mkdir command to create the /students directory.

   ```
   # mkdir /students
   ```

   b. Use the chgrp command to change the group for this /students directory to the students group.

   ```
   # ls -ld /students
   drwxr-xr-x. root  root  /students
   # chgrp -R students /students
   # ls -ld /students
   drwxr-xr-x. root  students  /students
   ```

   - Notice that the group was set to root before issuing the chgrp command.
   - The -R option (recursive) sets the new group for files and directories within /students.

   c. Use the chmod command to set the setgid bit on the directory, and give write permissions to the group.

   ```
   # chmod -R 2775 /students
   # ls -ld /students
   drwxrwsr-x. root  students  /students
   ```

   - Notice the new group permissions on the /students directory.

2. Create files in the /students directory as the oracle user.

   a. Use the usermod command to add the oracle user to the students group.

   ```
   # usermod -aG students oracle
   ```

   b. Use the su command to become the oracle user.

   ```
   # su - oracle
   $ whoami
   oracle
   ```

   c. Use the groups command to display group membership.

   ```
   $ groups
   oracle  students
   ```

   - Notice that the oracle user belongs to two groups, oracle and students.

d. Use the `cd` command to change to the `/students` directory.

```
$ cd /students
```

e. Use the `touch` command to create a file in the `/students` directory.

```
$ touch oracle_file
```

f. Use the `ls` command to display the permissions and ownership of the new file.

```
$ ls -l oracle_file
rw-rw-r--  oracle  students  oracle_file
```

- Notice that the permissions are read-write for the `students` group.

3. Create and edit files within the shared directory as a different user.

a. Use the `su` command to become the `student1` user. The password is `password`.

```
$ su – student1
Password: password
$ whoami
student1
```

b. Use the `groups` command to display group membership.

```
$ groups
student1  students
```

- Notice that the `student1` user belongs to two groups, `student1` and `students`.

c. Use the `cd` command to change to the `/students` directory.

```
$ cd /students
```

d. Use the `touch` command to create a file in the `/students` directory.

```
$ touch student1_file
```

e. Use the `ls` command to display the permissions and ownership of the files.

```
$ ls -l
rw-rw-r--  oracle    students  oracle_file
rw-rw-r--  student1  students  student1_file
```

- Notice that the permissions are read-write for the `students` group.

f. Use the `touch` command to update the time stamp on the `oracle_file`.

```
$ touch oracle_file
```

- Updating the time stamp implies write permissions on the file.

g. Use the `exit` command twice to log out as the `student1` user, and then as the `oracle` user.

```
$ exit
$ whoami
oracle
$ exit
# whoami
root
```

Practice 10-4: Password Aging

Overview

In this practice, you modify the password aging parameters for a user.

Tasks

1. View the password aging information.

   a. List password aging information in /etc/shadow for user student1.

```
# grep student1 /etc/shadow
student1:$6$...:16049:0:99999:7:::
```

   - This example gives 16049 days since the password changed (counted in days since Jan 1, 1970). This will differ depending on when you created the account.
   - 0 is the number of days that need to pass before the password must be changed by the user (0 means **never**).
   - 99999 is the maximum number of days since the password changed that the password can be used. After this amount of days, the password needs to be changed by the user.
   - 7 is the number of days before expire date that the user is warned about the pending password change policy. If the password is not changed after this number of days, the user account is locked.

   b. Use the chage command to view password aging information for user student1.

```
# chage -l student1
Last password change                                  : Dec 10, 2013
Password expires                                      : never
Password inactive                                     : never
Account expires                                       : never
Minimum number of days between password change        : 0
Maximum number of days between password change        : 99999
Number of days or warning before password expires: 7
```

2. Modify the password aging parameters.

   a. Make the following changes to password aging for user `student1`:

      - Change the minimum password age value to `14`.
      - Change the maximum password age value to `30`.

```
# chage student1
Changing the aging information for student1
Enter the new value, or press ENTER for the default
       Minimum Password Age [0]:  14
       Maximum Password Age [99999]:  30
       Last Password Change (YYYY-MM-DD) [2013-12-10]: ENTER
       Password Expiration Warning [7]:  ENTER
       Password Inactive [-1]:  ENTER
       Account Expiration Date (YYYY-MM-DD) [1969-12-31]:  ENTER
```

      - Making these changes means that in 14 days, the user has 30 days to change his or her password.

   b. View the results of the change for `student1` in `/etc/shadow`.

```
# grep student1 /etc/shadow
student1:$6$...:16049:14:30:7:::
```

   c. Use the `chage` command to view password aging information for user `student1`.

```
# chage -l student1
Last password change                               : Dec 10, 2013
Password expires                                   : Jan 09, 2014
Password inactive                                  : never
Account expires                                    : never
Minimum number of days between password change     : 14
Maximum number of days between password change     : 30
Number of days or warning before password expires: 7
```

Practice 10-5: Using the User Manager GUI

Overview

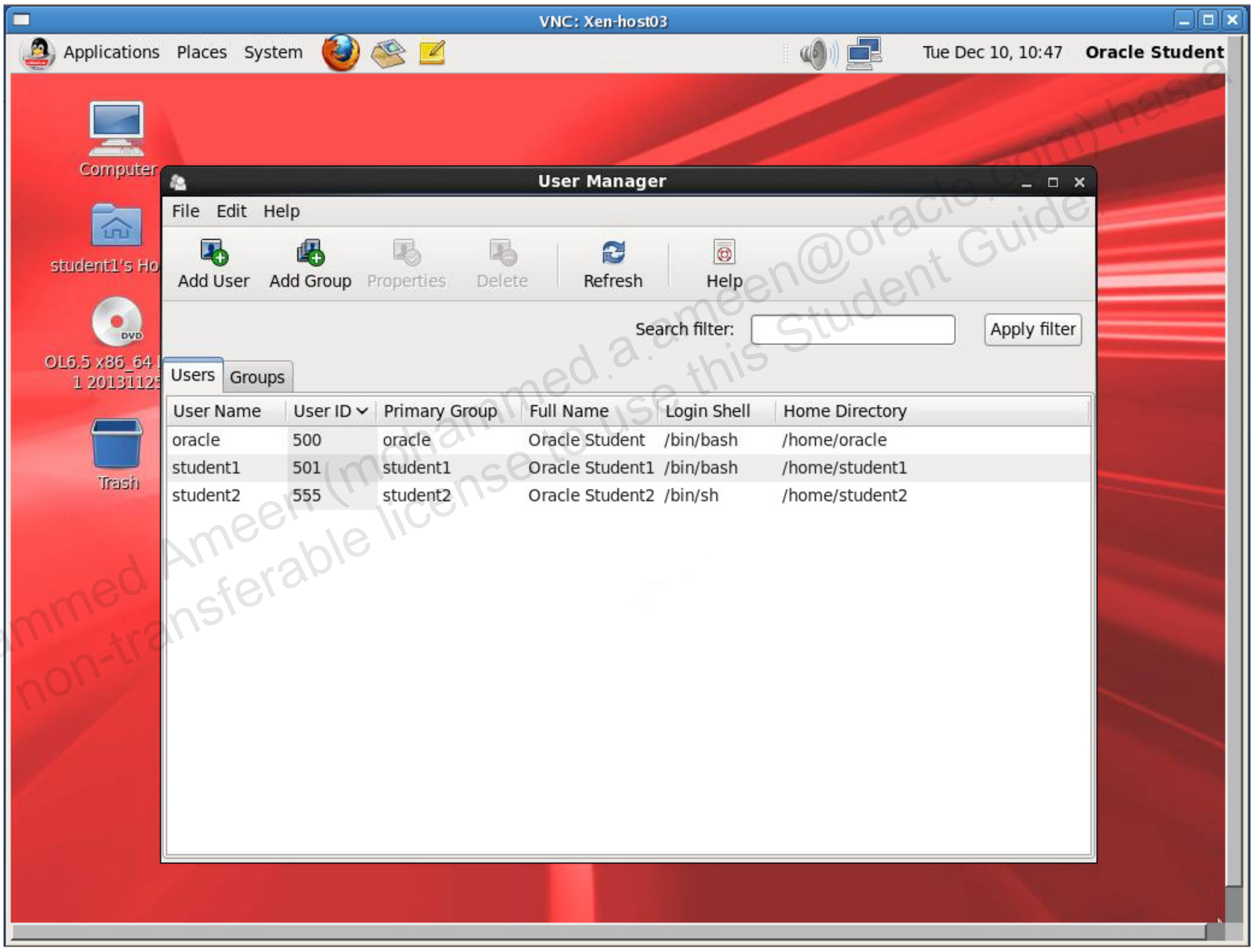
In this practice, you use the User Manager GUI to add a user, and add and delete a group account.

Tasks

1. Add a new user by using the User Manager GUI.

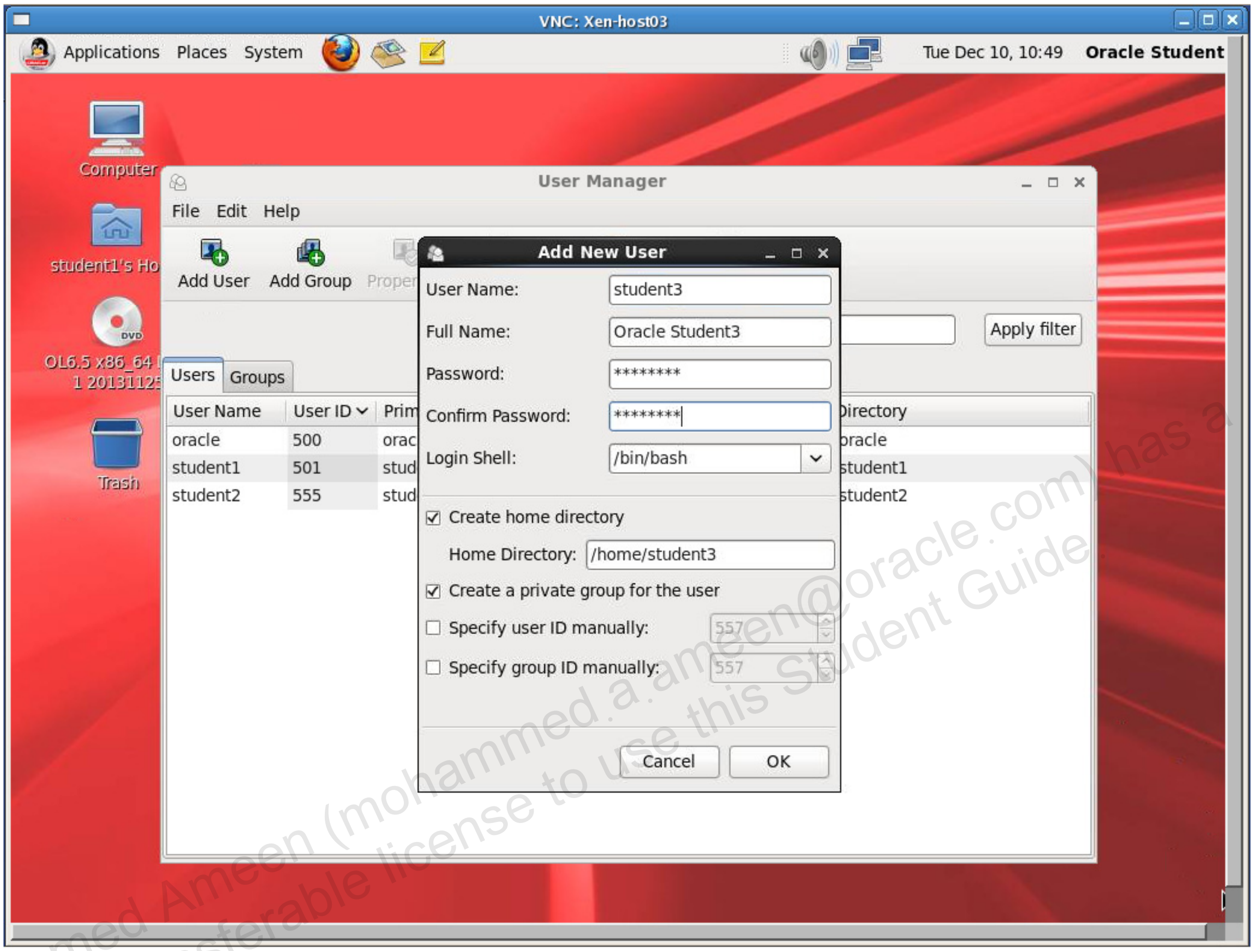
   a. Use the `system-config-users` command to display the User Manager Tool GUI.

```
# system-config-users
```



b. Click **Add User** and provide the new user information, as shown below.

- Use `password` for the **Password** and **Confirm Password** fields.



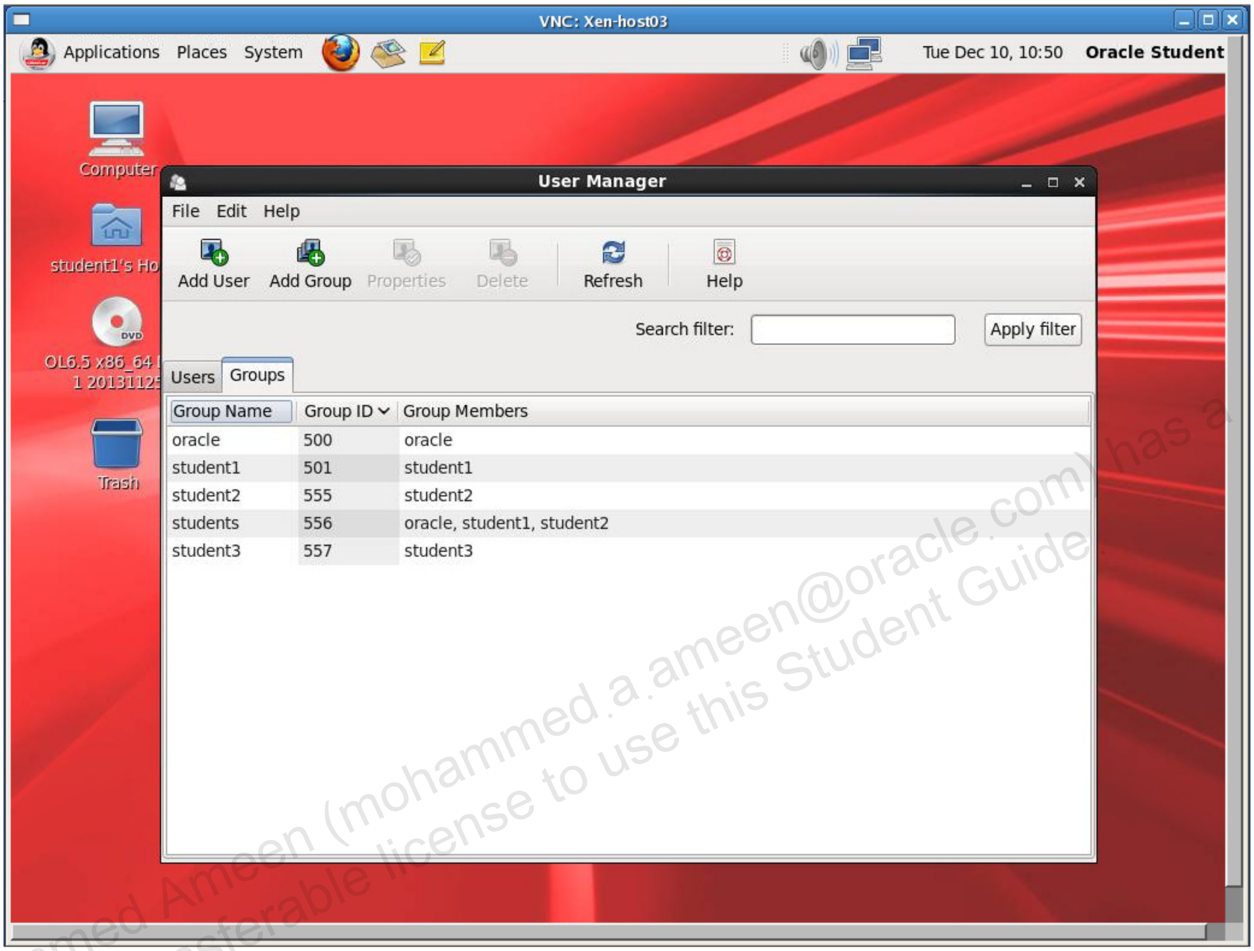
c. Click **OK**.

d. Click **Yes** to use the weak password.

- The updated list of user accounts is displayed.

2. Add a new group.

   a. Click the **Groups** tab to display the list of group accounts, as shown below.



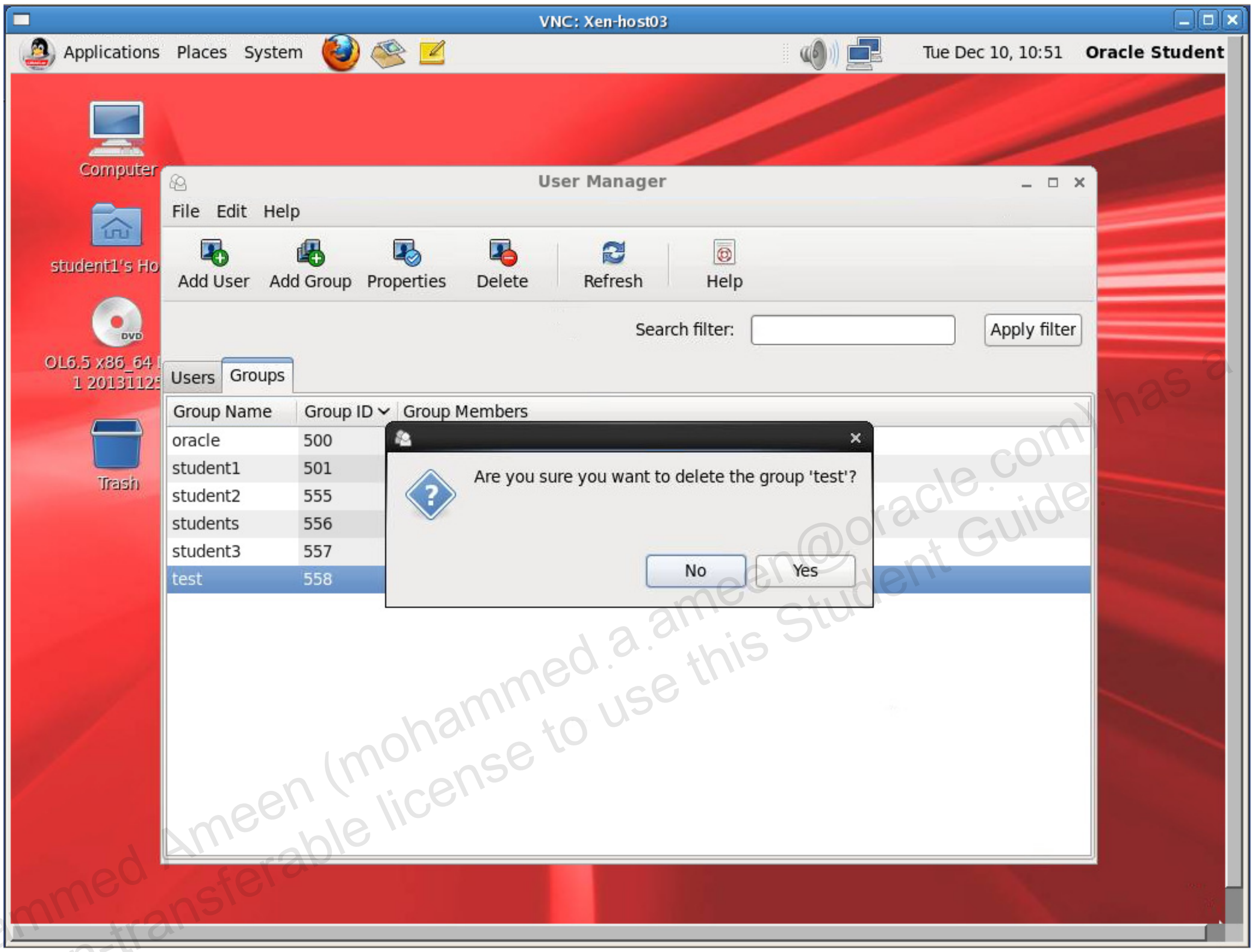
   b. Click **Add Group** and provide a **Group Name** of your choice when prompted.

   c. Click **OK** and notice the updated group list.

3. Delete a group

   a. Select the newly added group and click the **Delete** button.

   b. Click **Yes** to confirm the delete, as shown below.



   c. Select the `student1` group and click the **Delete** button.

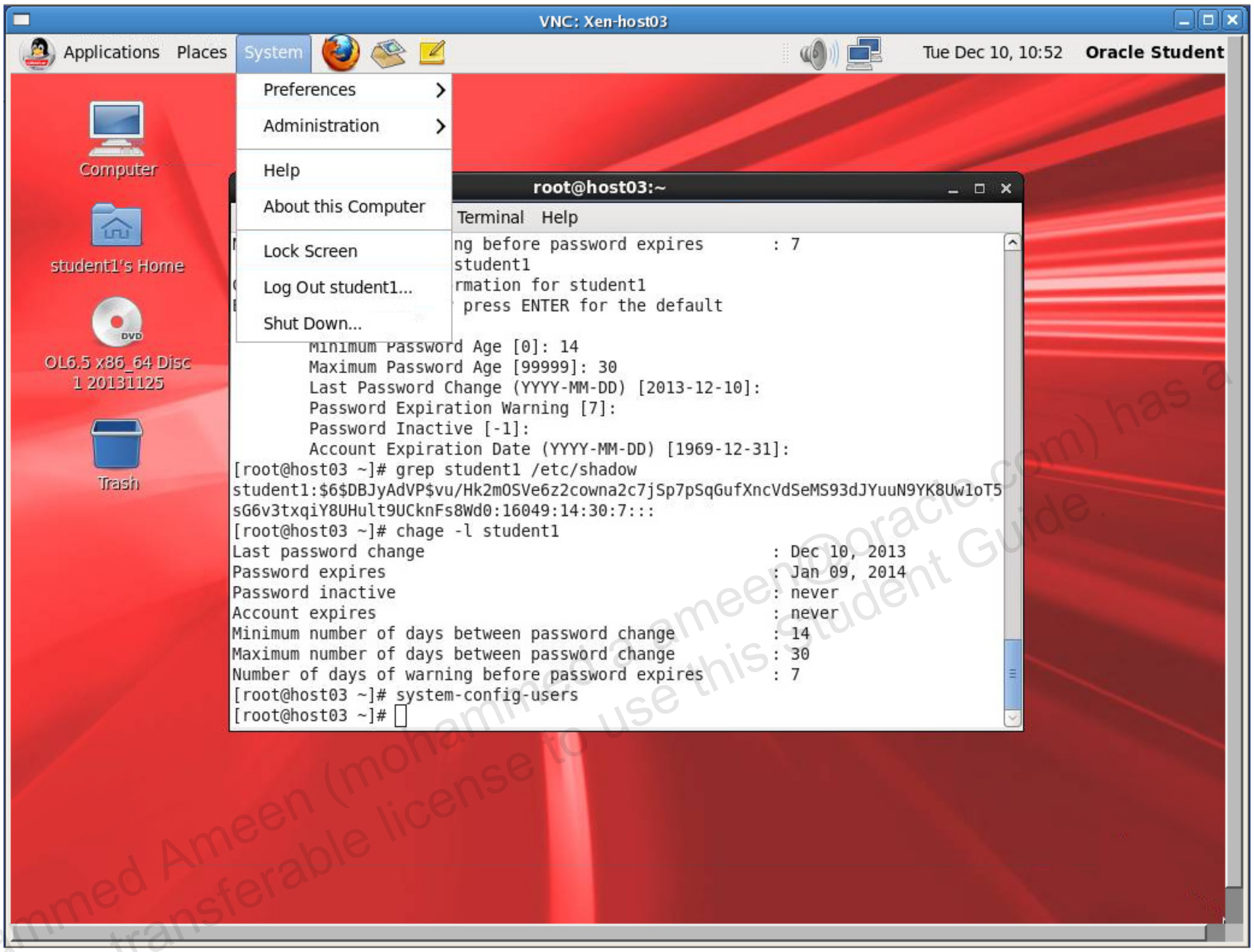
      - Notice that you cannot delete this group, because the user `student1` still exists.

4. Exit the User Manager tool.

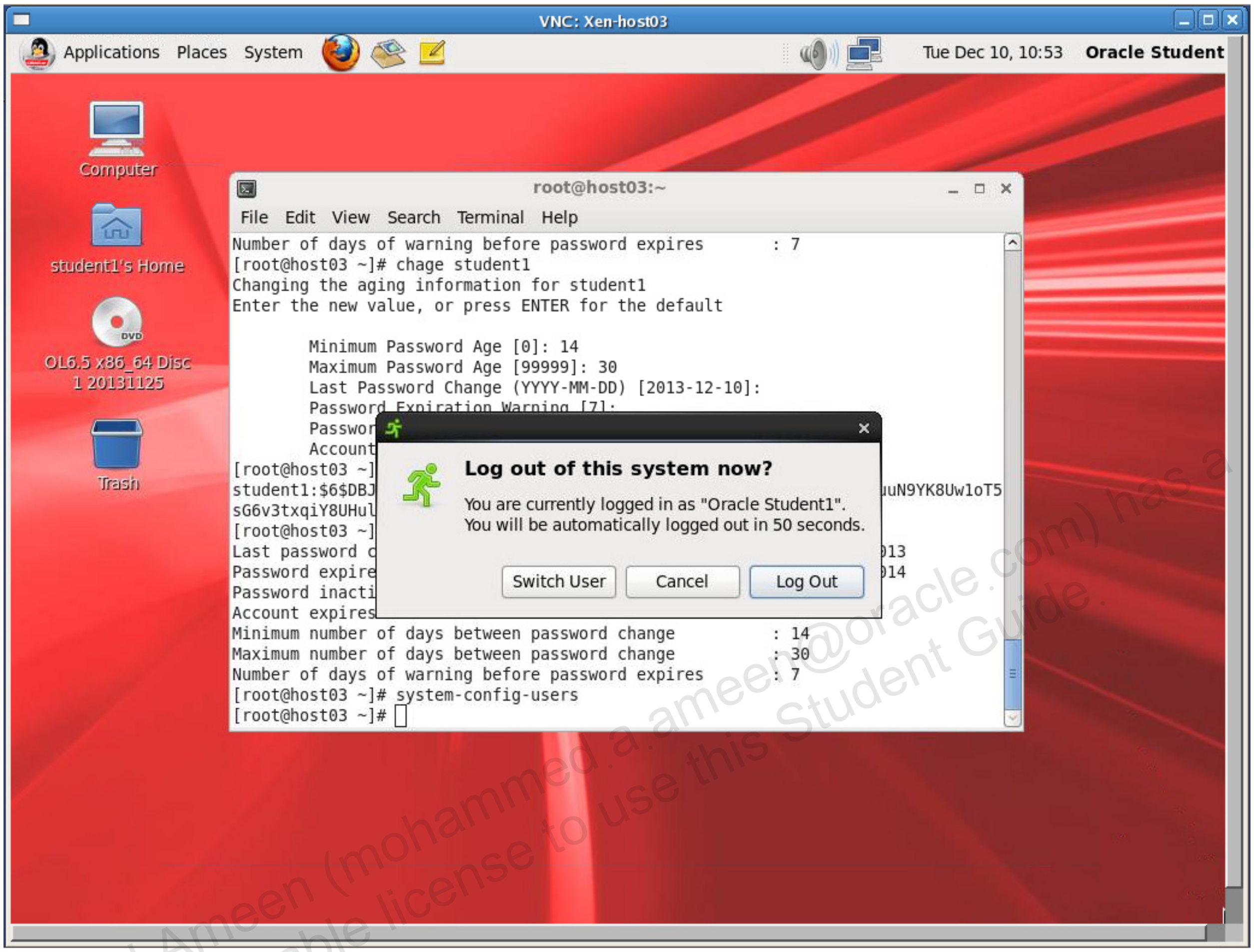
   Select **Quit** from the **File** menu.

5. Log out as user `student1`.

   a. Log out as user `student1` by selecting **Log out student1…** from the **System** menu as shown below.

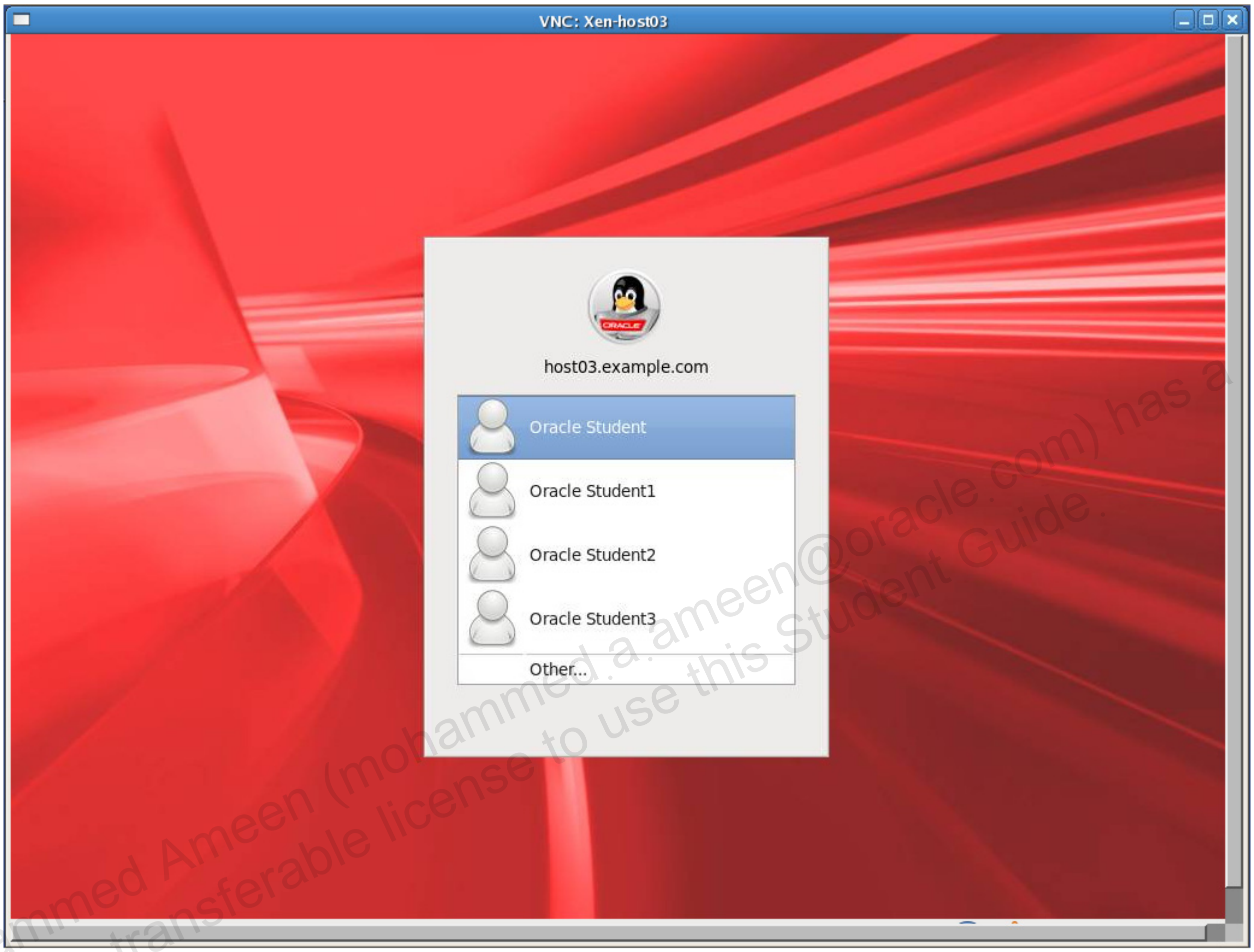


b. Click **Log Out** in the pop-up as shown below.



6. Log in as user `oracle`.

   a. Log in as the `oracle` user by selecting `Oracle Student` from the menu as shown below.



   b. Enter the password `oracle` when prompted for the password.

   c. Right-click on the desktop and select **Open in Terminal** from the pop-up menu.

   d. From the command prompt in the terminal window, use the `su -` command to become the `root` user. Password is `oracle`.

```
$ su -
Password: oracle
# whoami
root
```

Practice 10-6: Restricting the Use of the `su` Command

Overview

In this practice, you limit the use of the `su` command to members of the `wheel` group.

Tasks

1. Add the `oracle` user to the `wheel` group.
   a. Use the `grep` command to display the `wheel` entry in the `/etc/group` file.

```
# grep wheel /etc/group
wheel:x:10:
```

   b. Use the `usermod` command to add the `oracle` user to the `wheel` group. Re-run the `grep` command in Task 1a to verify the modification occurred.

```
# usermod -aG wheel oracle
# grep wheel /etc/group
wheel:x:10:oracle
```

   - Notice the `oracle` user is now a member of the `wheel` group.

2. Restrict the use of the `su` command to members of the `wheel` group.
   - The procedure shown below involves the modification of a Pluggable Authentication Module (PAM) configuration file. PAM is discussed in Lesson 16.

   a. Use the `vi` editor to uncomment (remove the `#` sign) the following entry in the `/etc/pam.d/su` file:

```
# vi /etc/pam.d/su
...
#auth       required   pam_wheel.so   use_uid  (old entry)
auth       required   pam_wheel.so   use_uid   (new entry)
...
```

   - With this entry uncommented, only members of the `wheel` group can use the `su` command (with the exception of the `root` user).

3. Verify that only members of the `wheel` group can use the `su` command.
   a. As the `root` user, use the `su - student1` command to become the `student1` user.

```
# su - student1
```

   - The `su` command is successful because you are the `root` user.

b. Use the `whoami` command to confirm you are the `student1` user. Use the `su` command to become the `root` user. The `root` user password is `oracle`.

```
$ whoami
student1
$ su –
Password: oracle
su: incorrect password
```

- The `su` command fails with an "incorrect password" message, but the real reason the `su` command fails is because your system is configured to allow only members of the `wheel` group to use the `su` command.

c. Use the `exit` command to log out as the `student1` user. Use the `whoami` command to confirm you are the `root` user. Use the `exit` command again, and then `whoami` to verify that you are the `oracle` user.

```
$ exit
logout
# whoami
root
# exit
logout
$ whoami
oracle
```

d. Use the `su` command to become the `root` user. The `root` user password is `oracle`.

```
$ su –
Password: oracle
# whoami
root
```

- The `su` command succeeded this time because the `oracle` user is a member of the `wheel` group.

4. Remove restriction of the `su` command to members of the `wheel` group.

   Use the `vi` editor to comment out (insert the # sign) the following entry in the `/etc/pam.d/su` file:

```
# vi /etc/pam.d/su
...
auth      required   pam_wheel.so   use_uid    (old entry)
#auth      required   pam_wheel.so   use_uid   (new entry)
...
```

- With this entry commented out, all users can use the `su` command.

Practice 10-7: Allowing the Use of the `sudo` Command

Overview

In this practice you authorize the `oracle` user to use `sudo` to run administrative commands.

Tasks

1. Attempt to add a new user without `root` privileges.
   a. Use the `exit` command to log out as the `root` user. Then run the `whoami` command to verify you are the `oracle` user.

   ```
   # exit
   logout
   $ whoami
   oracle
   ```

   b. Use the `useradd` command to add `new_user` user.

   ```
   $ useradd new_user
   bash: /usr/sbin/useradd: Permission denied
   ```

   - Notice you do not have permission to add a new user.

   c. Insert the `sudo` command before the previous `useradd` command (in Task 1b) to add `new_user` user. Enter `oracle` for password when prompted.

   ```
   $ sudo useradd new_user

   We trust you have received the usual lecture from the local
   System Administrator. It usually boils down to these three
   things:

       #1) Respect the privacy of others.
       #2) Think before you type.
       #3) With great power comes great responsibility.

   [sudo] password for oracle: oracle
   oracle is not in the sudoers file. This incident will be
   reported.
   ```

   - Notice that the command fails because the `oracle` user is not in the `sudoers` file.
   - Also notice the incident is reported (in the `/var/log/secure` file).

   d. Use the `su -` command to become the `root` user. The password is `oracle`.

   ```
   $ su -
   Password: oracle
   # whoami
   root
   ```

e. Use the `tail` command to view the end of the `/var/log/secure` file.

```
# tail /var/log/secure
...
<date_time> host03 sudo: oracle : user NOT in sudoers ...
...
```

- Notice the "`sudo`" entry in the `/var/log/secure` file.

2. Add the `oracle` user to the `/etc/sudoers` file.

a. Use the `visudo` command to edit the `/etc/sudoers` file.

- This command opens the `/etc/sudoers` file with the `vi` editor but also locks the file.

```
# visudo
```

b. Add the following line to `/etc/sudoers`, which allows the `oracle` user to use `sudo` to run administrative commands. After adding the line, save the file and exit `visudo`.

```
oracle ALL=(ALL) ALL
```

3. Add a new user by using `sudo`.

a. Use the `exit` command to log out as the `root` user, then run the `whoami` command to verify you are the `oracle` user.

```
# exit
logout
$ whoami
oracle
```

b. Use the `sudo useradd` command to add `new_user` user. Enter `oracle` for password when prompted.

```
$ sudo useradd new_user
[sudo] password for oracle: oracle
```

- No error message implies the command was successful.

c. Use the `grep` command to search for `new_user` in the `/etc/passwd` file.

```
$ grep new_user /etc/passwd
new_user:x:558:558::/home/new_user:/bin/bash
```

4. Run another command, which requires administrative privileges.

a. Use the `service` command to restart the `atd` service.

```
$ service atd restart
rm: cannot remove 'var/run/atd.pid': Permission denied [FAILED]
...
```

- Notice the command failed with "Permission denied"

b. Use `sudo` and run the `service` command again, to restart the `atd` service.

```
$ sudo service atd restart
Stopping atd:                                              [ OK ]
Starting atd:                                              [ OK ]
```

- Notice the command succeeded this time.
- Also notice you were not prompted to enter the `oracle` user's password.
- When a user has been authenticated, the user may then use `sudo` without a password for a short period of time (the default timeout is 5 minutes).
- Do not perform this, but to override the default 5 minute timeout, set the "`timestamp_timeout`" value in `/etc/sudoers` to change the default.

c. In preparation for the next practice, use the `su -` command to become the `root` user. The password is `oracle`.

```
$ su -
Password: oracle
# whoami
Root
```

Practices for Lesson 11: File Systems

Chapter 11

Practices for Lesson 11: File Systems

Practices Overview

In these practices, you:

- Display the partition table, list the mounted file systems, and display the swap space configured during installation
- Partition disk devices, and create and mount file systems on the partitions
- Increase the amount of swap space by creating a swap file

Practice 11-1: Listing the Current Disk Partitions

Overview

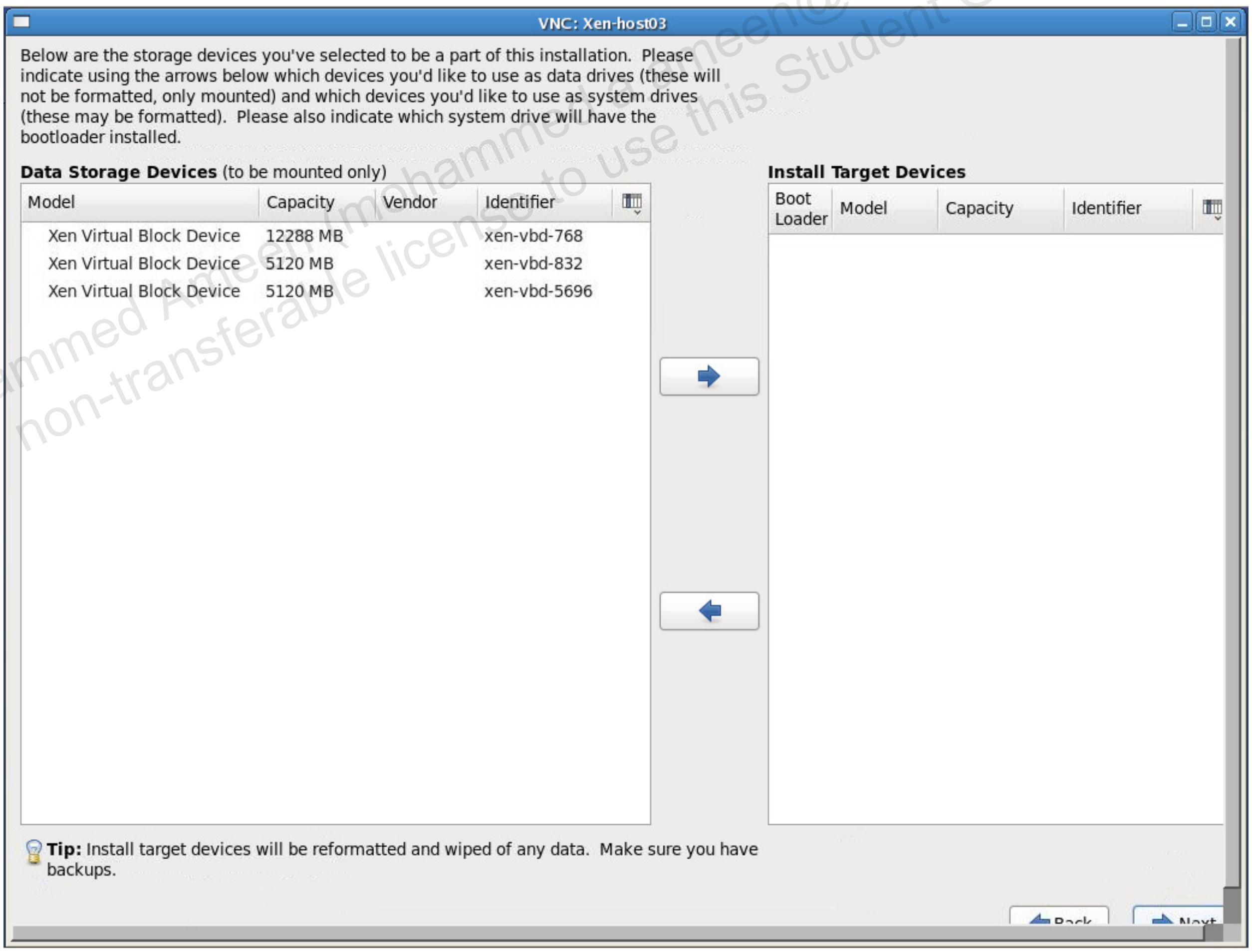
In this practice, you verify the selections made during installation regarding disk partitioning, mount points for file systems, and swap space.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Relate the partition table to selections made during installation.
   - Recall that three virtual disk images were created before initiating the installation:
     - A 12-GB disk image (**system.img**) for the operating system
     - A 5-GB disk image (**u01.img**) for the various storage administration practices
     - A 5-GB disk image (**u02.img**) for the various storage administration practices
   - These three Storage Devices were presented during the installation, as shown in the following:



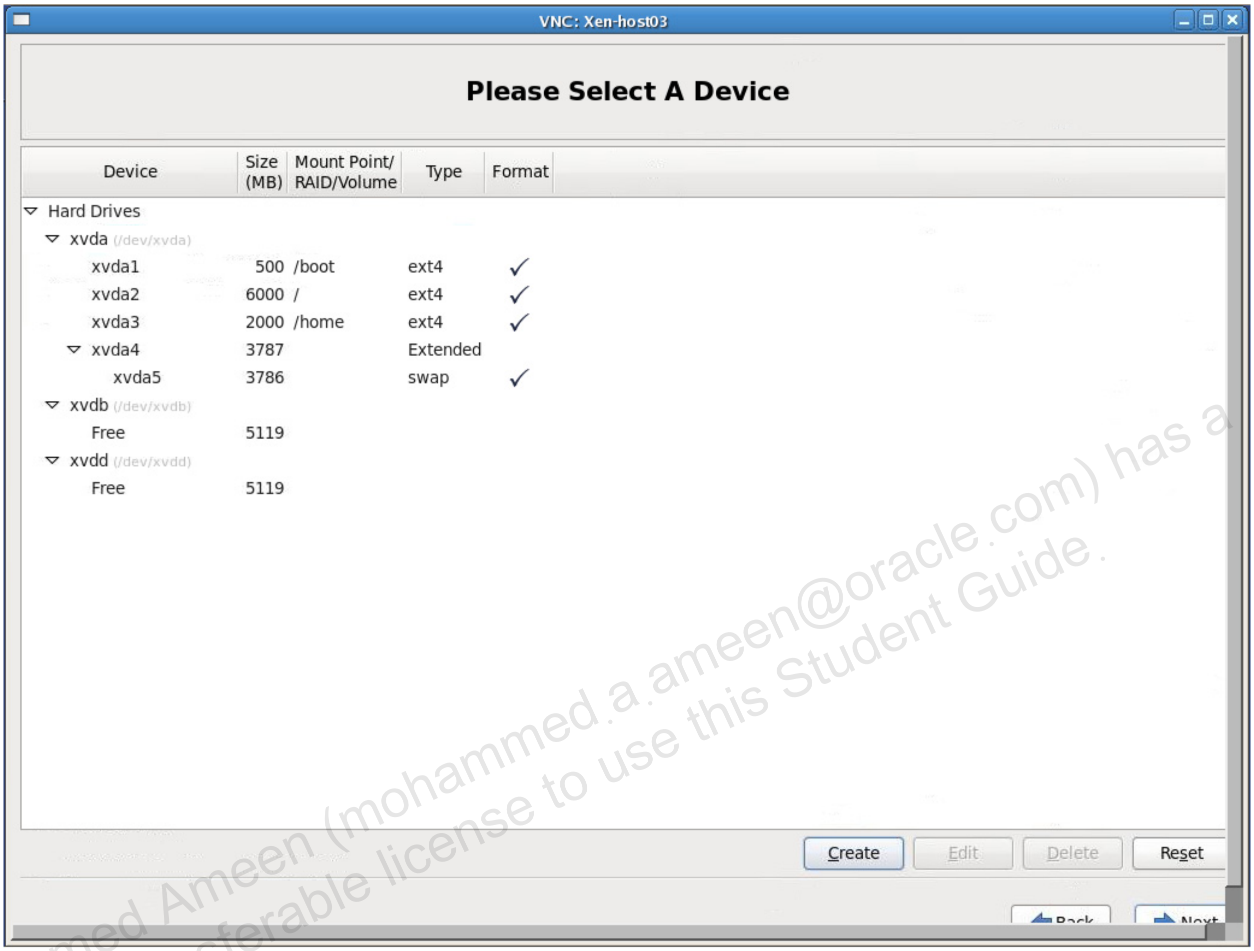
Use the `fdisk` command to display the partition table.

```
# fdisk -l | grep /dev
Disk /dev/xvda: 12.9 GB, 12884901888 bytes
/dev/xvda1   *    1      64   512000   83  Linux
/dev/xvda2       64     829  6144000   83  Linux
/dev/xvda3      829    1084  2048000   83  Linux
/dev/xvda4     1084    1567  3877888    5  Extended
/dev/xvda5     1084    1567  3876864   82  Linux swap / Solaris
Disk /dev/xvdb: 5368 MB,  5368709120 bytes
Disk /dev/xvdd: 5368 MB,  5368709120 bytes
```

- This lists the three storage devices:
  - `/dev/xvda`, approximately 12 GB in size
  - `/dev/xvdb`, approximately 5 GB in size
  - `/dev/xvdd`, approximately 5 GB in size
- This indicates that the first partition on `xvda` contains **Boot** files (marked with *****).
- This also indicates that the fourth partition on `xvda` is an **Extended** partition.
  - A maximum of four primary partitions can be placed on any hard disk.
  - One of the four partitions may be designated as an Extended partition.
  - This Extended partition may then be subdivided into multiple logical partitions.
  - The `/dev/xvda5` is a logical partition as shown as follows:

2. Relate the mounted partitions to selections made during installation.

- Recall the final partition table created during the installation, shown in the following:



Use the `df` command to list the mounted partitions.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
/dev/xvda2     5.7G  3.2G   2.2G   60%  /
tmpfs         1002M  384K  1001M    1%  /dev/shm
/dev/xvda1     477M   55M   397M   13%  /boot
/dev/xvda3     1.9G  5.1M   1.8G    1%  /home
/dev/sr0       3.7G  3.7G      0  100%  /media/OL6.5 x86_64 ...
```

- The first partition on hard drive xvda (**xvda1**) contains **boot** files, mounted on `/boot`.
- The second partition on hard drive xvda (**xvda2**) is the **root** file system, mounted on `/`.
- The third partition on hard drive xvda (**xvda3**) is for user **home** directories, mounted on `/home`.

3. Relate the swap space to selections made during installation.

   Use the `swapon` command to display the swap space.

```
# swapon –s
Filename                          Type            Size     Used   Priority
/dev/xvda5                        partition       3876860 0       -1
```

   - The fifth partition (the first logical partition) on hard drive xvda (**xvda5**) is **swap** space.
   - The `/proc/swaps` file provides the same information.

```
# cat /proc/swaps
Filename                          Type            Size     Used   Priority
/dev/xvda5                        partition       3876860 0       -1
```

Practice 11-2: Partitioning a Storage Device

Overview

In this practice, you create a partition by using the `fdisk` utility, and create a second partition by using the `parted` utility. Recall from the lecture that if the file system is greater than 2 TB, you cannot use `fdisk` and you must use `parted`.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Partition a storage device by using `fdisk`.
   a. Use the `fdisk` command to partition `/dev/xvdb`.

```
# fdisk /dev/xvdb
WARNING: DOS-compatible mode is deprecated. It's strongly
recommended to switch off the mode (command 'c') and change
display units to sectors (command 'u').
Command (m for help):
```

   - DOS does not allow a partition to start (or end) mid-cylinder. Toggling those two items ('c' and 'u') allows partition start and end points to be entered in a sector-exact way, rather than by cylinder.

   b. Perform the recommended actions (switch off DOS-compatible mode and change display units to sectors).

```
Command (m for help): c
DOS Compatibility flag is not set
Command (m for help): u
Changing display/entry units to sectors
```

   c. Display the `fdisk` menu.

```
Command (m for help): m
a   toggle a bootable flag
b   edit bsd disklabel
c   toggle the dos compatibility flag
d   delete a partition
l   list known partition types
m   print this menu
n   add a new partition
o   create a new empty DOS partition table
p   print the partition table
q   quit without saving changes
s   create a new empty sun disklabel
t   change a partition's system id
u   change display/entry units
```

```
v    verify the partition table
w    write table to disk and exit
x    extra functionality (experts only)
```

d. Add a new **primary** partition, giving the partition number **1**.

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
```

- As was previously discussed, a maximum of four primary partitions can be placed on any hard disk.
- One of the four partitions may be designated as an extended partition. This partition may then be subdivided into multiple logical partitions.

e. Continue adding the new partition, using the following parameters:

```
First sector (2048-10485759, default 2048): ENTER
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-10485759, default
10485759): 2100000
```

f. Display the new partition table.

```
Command (m for help): p
Disk /dev/xvdb: 5368 MB, 5368709120 bytes
...
    Device Boot   Start       End      Blocks    Id    System
/dev/xvdb1         2048   2100000     1048976+   83    Linux
```

g. Save the new partition table.

```
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

h. Use the `fdisk` command to list the partition table on `/dev/xvdb`.

```
# fdisk -l /dev/xvdb
Disk /dev/xvdb: 5368 MB, 5368709120 bytes
...
    Device Boot   Start      End     Blocks    Id    System
/dev/xvdb1            1      519    1048976+    83    Linux
```

2. Partition a storage device by using `parted`.

   a. Use the `parted` command to partition `/dev/xvdd`.

```
# parted /dev/xvdd
GNU Parted 2.1
Using /dev/xvdd
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted)
```

   b. Enter **`help`** to view a list of commands.

```
(parted) help
  align-check TYPE N  check partition N for TYPE(min|opt)
alignment
  check NUMBER        do a simple check on the file system
...
```

   c. Enter **`print`** to print the partition table.

```
(parted) print
Model: Xen Virtual Block Device (xvd)
Disk /dev/xvdd: 5369MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Number  Start   End     Size    Type    Filesystem  Flags
```

- Notice that the disk has not been partitioned.

   d. Get **help** for the **`mkpart`** command.

```
(parted) help mkpart
  mkpart PART-TYPE [FS-TYPE] START END    make a partition
  PART-TYPE is one of: primary, logical, extended
  FS-TYPE is one of: ext4, ext3, ext2, fat32, fat16...
  START and END are disk locations, such as 4GB or 10%...
  'mkpart' makes a partition without creating a new file
  system on the partition.  FS-TYPE may be specified to set
  an appropriate partition ID.
```

   e. Use the **`mkpart`** command to create a partition, using the following parameters:

```
(parted) mkpart
Partition type? primary/extended? primary
Filesystem type? [ext2] ENTER
Start? 0
End? 20%
Warning: The resulting partition is not properly aligned for
best performance. Ignore/Cancel? i
```

f. Print the partition table.

```
(parted) print
Model: Xen Virtual Block Device (xvd)
Disk /dev/xvdd: 5369MB
Sector size (logical/physical): 512B/512B
Partition Table: msdos
Number  Start   End     Size    Type     Filesystem  Flags
1       512B    1074MB  1074MB  primary
```

g. Save the partition table and exit.

```
(parted) quit
Information: You may need to update /etc/fstab
```

h. Use the `fdisk` command to list the partition table on `/dev/xvdd`.

- Remember that Linux is case-sensitive. Use lowercase `/dev` in the command as shown.

```
# fdisk -l /dev/xvdd
Disk /dev/xvdd: 5368 MB, 5368709120 bytes
...
Device      Boot   Start      End    Blocks    Id    System
/dev/xvdd1             1     1886   1048575+   83    Linux
```

3. Display the major and minor numbers.

a. Display the partitions in `/proc/partitions`.

```
# cat /proc/partitions
major minor  #blocks  name
 202      0  12582912 xvda
 202      1    512000 xvda1
 202      2   6144000 xvda2
 202      3   2048000 xvda3
 202      4         1 xvda4
 202      5   3876864 xvda5
 202     16   5242880 xvdb
 202     17   1048976 xvdb1
 202     48   5242880 xvdd
 202     49   1048575 xvdd1
...
```

- The kernel uses the major and minor numbers to access a device.

b. Use the `ls` command to list the major and minor numbers for devices in `/dev`.

```
# ls -l /dev/xvd*
brw-rw----. 1 root disk 202,  0 Dec 22 05:19 /dev/xvda
brw-rw----. 1 root disk 202,  1 Dec 22 03:18 /dev/xvda1
brw-rw----. 1 root disk 202,  2 Dec 22 03:18 /dev/xvda2
brw-rw----. 1 root disk 202,  3 Dec 22 03:18 /dev/xvda3
brw-rw----. 1 root disk 202,  4 Dec 22 03:18 /dev/xvda4
brw-rw----. 1 root disk 202,  5 Dec 22 03:18 /dev/xvda5
brw-rw----. 1 root disk 202, 16 Dec 22 05:19 /dev/xvdb
brw-rw----. 1 root disk 202, 17 Dec 22 03:18 /dev/xvdb1
brw-rw----. 1 root disk 202, 48 Dec 22 06:22 /dev/xvdd
brw-rw----. 1 root disk 202, 49 Dec 22 10:25 /dev/xvdd1
```

Practice 11-3: Creating ext File Systems

Overview

In this practice, you create ext3 and ext4 file systems on the new partitions, mount the file systems, and update the file system mount table.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Make an **ext3** file system on `/dev/xvdb1`.
   a. Use the `mkfs` command to make an **ext3** file system on `/dev/xvdb1`.

```
# mkfs -t ext3 /dev/xvdb1
mke2fs 1.43-WIP (20-Jun-2013)
warning: 100 blocks unused.
Filesystem label=
OS type: Linux
...
Writing inode tables: done
Creating journal (8192 blocks): done
...
```

   b. Use the `blkid` command to display the attributes of the `/dev/xvdb1` block device.
      - UUID values are not shown throughout this practice.

```
# blkid /dev/xvdb1
/dev/xvdb1: UUID="..." SEC_TYPE="ext2" TYPE="ext3"
```

2. Make an **ext4** file system on `/dev/xvdd1`.
   a. Use the `mkfs` command to make an **ext4** file system on `/dev/xvdd1` and assign a label of **Test**.

```
# mkfs -t ext4 -L Test /dev/xvdd1
mke2fs 1.43-WIP (20-Jun-2013)
Filesystem label=Test
OS type: Linux
...
Writing inode tables: done
Creating journal (4096 blocks): done
...
```

   b. Display the attributes of the `/dev/xvdd1` block device.

```
# blkid /dev/xvdd1
/dev/xvdd1: LABEL="Test" UUID="..." TYPE="ext4"
```

3. Set the file system label on /dev/xvdb1.

   a. Use the e2label command to set the file system label on /dev/xvdb1 to **Dev**.

      - Remember that Linux is case-sensitive. Use uppercase **Dev** in the command as shown.

```
# e2label /dev/xvdb1 Dev
```

   b. Display the attributes of all the block devices.

```
# blkid
/dev/xvda1: UUID=”...” TYPE=”ext4”
/dev/xvda2: UUID=”...” TYPE=”ext4”
/dev/xvda3: UUID=”...” TYPE=”ext4”
/dev/xvda5: UUID=”...” TYPE=”swap”
...
/dev/xvdb1: UUID=”...” SEC_TYPE=”ext2” TYPE=”ext3” LABEL=”Dev”
/dev/xvdd1: LABEL=”Test” UUID=”...” TYPE=”ext4”
```

      - Notice that both /dev/xvdb1 and /dev/xvdd1 now have labels.

4. Mount the file systems.

   a. Use the mkdir command to create mount points.

```
# mkdir /Test /Dev
```

   b. Use the mount command to mount /dev/xvdb1 on /Dev.

```
# mount /dev/xvdb1 /Dev
```

   c. Mount /dev/xvdd1 on /Test.

```
# mount /dev/xvdd1 /Test
```

   d. Use the df command to display the mounted file systems.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
/dev/xvda2     5.7G  3.2G   2.2G   60%  /
tmpfs         1002M  384K  1001M    1%  /dev/shm
/dev/xvda1     477M   55M   397M   13%  /boot
/dev/xvda3     1.9G  5.1M   1.8G    1%  /home
/dev/sr0       3.7G  3.7G      0  100%  /media/OL6.5 x86_64 ...
/dev/xvdb1    1008M   34M   924M    4%  /Dev
/dev/xvdd1     992M  1.3M   940M    1%  /Test
```

e. Use the `mount` command to display the mounted file systems.

```
# mount
/dev/xvda2 on / type ext4 (rw)
...
/dev/xvda1 on /boot type ext4 (rw)
/dev/xvda3 on /home type ext4 (rw)
...
/dev/xvdb1 on /Dev type ext3 (rw)
/dev/xvdd1 on /Test type ext4 (rw)
```

- Notice that the file systems are mounted read-write (**rw**) by default.

f. Display the mounts in `/proc/mounts`.

```
# cat /proc/mounts
rootfs / rootfs rw 0 0
proc /proc proc rw,relatime 0 0
...
/dev/xvdb1 /Dev ext3 rw,seclabel,relatime,errors=continue...
/dev/xvdd1 /Test ext4 rw,seclabel,relatime,data=ordered 0 0
```

5. Update the file systems mount table.

Use the `vi` editor to add the following entries to `/etc/fstab`.

```
# vi /etc/fstab
LABEL=Dev    /Dev    ext3  defaults  0  0
LABEL=Test   /Test   ext4  defaults  0  0
```

- After updating `/etc/fstab`, the new file systems will mount after a reboot.

Practice 11-4: Increasing Swap Space

Overview

In this practice, you increase the amount of swap space by creating, initializing, and enabling a swap file.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Display the current amount of swap space.
   a. Use the `swapon` command to display the current amount of swap space.

```
# swapon -s
Filename                    Type             Size     Used  Priority
/dev/xvda5                  partition        3876860  0     -1
```

   b. Display the amount of swap space used by viewing the `/proc/meminfo` file.

```
# grep -i swap /proc/meminfo
SwapCached:          0 kB
SwapTotal:     3876860 kB
SwapFree:      3876860 kB
```

   c. Use the `free` command to display the amount of swap space.

```
# free
           total     used            free
...
Swap     3876860        0         3876860
```

2. Create and initialize a swap file.
   a. Use the `dd` command to create a 1 GB swap file, `/swapfile`.

```
# dd if=/dev/zero of=/swapfile bs=1024 count=1000000
1000000+1 records in
1000000+1 records in
1024000000 bytes (1.0 GB) copied, 7.28291 s, 141 MB/s
```

   b. Use the `mkswap` command to initialize the swap file.

```
# mkswap /swapfile
mkswap: /swapfile: warning: don't erase bootbits sectors
        on whole disk. Use -f to force.
Setting up swapspace version 1, size = 999996 KiB
no label, UUID=...
```

3. Enable swapping on the swap file.

   a. Use the `swapon` command to enable swapping on the swap file.

```
# swapon /swapfile
```

   b. Display the updated amount of swap space (repeat step 1).

```
# swapon -s
Filename                Type            Size     Used   Priority
/dev/xvda5              partition       3876860 0      -1
/swapfile               file            999996  0      -2
# grep -i swap /proc/meminfo
SwapCached:          0 kB
SwapTotal:     4876856 kB
SwapFree:      4876856 kB
# free
            total     used            free
...
Swap      4876856        0         4876856
```

Practices for Lesson 12: Storage Administration

Chapter 12

Practices for Lesson 12: Storage Administration

Practices Overview

In these practices, you create and work with logical volumes and RAID devices.

Practice 12-1: Creating Linux LVM Partitions

Overview

In this practice, you create new partitions and change the system ID to Linux LVM.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Un-mount all file systems currently mounted on `/dev/xvdb`.
   a. Determine what is currently mounted on `/dev/xvdb`.

```
# df -h | grep xvdb
/dev/xvdb1    1008M   34M   924M    4%  /Dev
```

   b. Un-mount `/dev/xvdb1`.

```
# umount /dev/xvdb1
```

2. Create a new partition on `/dev/xvdb`.
   a. Use the `fdisk` command to create a new **primary** partition on `/dev/xvdb` by using the following parameters:

```
# fdisk /dev/xvdb
WARNING: DOS-compatible mode is deprecated. It's strongly
recommended to switch off the mode (command 'c') and change
display units to sectors (command 'u').
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (519-2590, default 519): ENTER
Using default value 519
Last cylinder, +cylinders or +size{K,M,G} (519-2590, default
2590): +1G
```

   b. Use the "`t`" command to change the system ID on the new partition.

```
Command (m for help): t
Partition number (1-4): 2
Hex code (type L to list codes): L
 0  Empty       24  NEC DOS          81  Minix / old Lin ...
 1  FAT12       39  Plan 9           82  Linux swap      ...
 2  XENIX root  3c  PartitionMagic   83  Linux           ...
...
 8  AIX         4f QNX4.x 3rd part   8e  Linux LVM       ...
```

```
...
Hex code (type L to list codes): 8e
Changed system type of partition 2 to 8e (Linux LVM)
```

c. Print the new partition table.

```
Command (m for help): p
Disk /dev/xvdb: 5368 MB, 5368709120 bytes
...
    Device  Boot  Start    End    Blocks    Id    System
/dev/xvdb1            1    519   1048976+   83    Linux
/dev/xvdb2          519   1037   1048887+   8e    Linux LVM
```

d. Save the new partition table.

```
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

3. Unmount all file systems currently mounted on /dev/xvdd.

a. Determine what is currently mounted on /dev/xvdd.

```
# df -h | grep xvdd
/dev/xvdd1   1008M   18M   940M    2%  /Test
```

b. Unmount /dev/xvdd1.

```
# umount /dev/xvdd1
```

4. Create a new partition on /dev/xvdd.

a. Use the fdisk command to create a new **primary** partition on /dev/xvdd by using the following parameters:

```
# fdisk /dev/xvdd
WARNING: DOS-compatible mode is deprecated. It's strongly
recommended to switch off the mode (command 'c') and change
display units to sectors (command 'u').
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (1886-9429, default 1886): ENTER
Using default value 1886
Last cylinder, +cylinders or +size{K,M,G} (1886-9429, default
9429): +1G
```

b. Use the "`t`" command to change the system ID on the new partition.

```
Command (m for help): t
Partition number (1-4): 2
Hex code (type L to list codes): 8e
Changed system type of partition 2 to 8e (Linux LVM)
```

c. Save the new partition table.

```
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

d. List the partition table on `/dev/xvdd`.

```
# fdisk -l /dev/xvdd
Disk /dev/xvdd: 5368 MB, 5368709120 bytes
...
Device      Boot  Start    End   Blocks    Id   System
/dev/xvdd1            1   1886  1048575+   83   Linux
/dev/xvdd2         1886   3772  1048656+   8e   Linux LVM
```

5. Re-mount the file systems.

Use the `mount -a` command to mount all file systems mentioned in `/etc/fstab`.

```
# mount -a
```

6. Display the mounted file systems.

```
# df -h
...
/dev/xvdb1   1008M   34M   924M    4%  /Dev
/dev/xvdd1    992M  1.3M   940M    1%  /Test
```

Practice 12-2: Creating a Logical Volume

Overview

In this practice, you create physical volumes, a volume group, and a logical volume. You also use LVM utilities to display information about the LVM entities.

Tasks

1. Initialize the new partitions for use by LVM (create physical volumes).
   a. List the partitions with the Linux LVM (**8e**) system ID.

```
# fdisk -l | grep 8e
/dev/xvdb2          519   1037   1048887+   8e   Linux LVM
/dev/xvdd2         1886   3772   1048656    8e   Linux LVM
```

   b. Use the `pvcreate` command to create physical volumes on both partitions.

```
# pvcreate -v /dev/xvdb2 /dev/xvdd2
   Set up physical volume for "/dev/xvdb2" with ...
   Zeroing start of device /dev/xvdb2
 Writing physical volume data to disk "/dev/xvdb2"
 Physical volume "/dev/xvdb2" successfully created
   Set up physical volume for "/dev/xvdd2" with ...
   Zeroing start of device /dev/xvdd2
 Writing physical volume data to disk "/dev/xvdd2"
 Physical volume "/dev/xvdd2" successfully created
```

2. Display information about the physical volumes.
   a. Use the `pvdisplay` command to display attributes of the physical volumes.

```
# pvdisplay
 "/dev/xvdb2" is a new physical volume of "1.00 GiB"
 --- NEW Physical volume ---
 PV Name            /dev/xvdb2
 VG Name
 PV Size            1.00 GiB
...
"/dev/xvdd2" is a new physical volume of "1.00 GiB"
 --- NEW Physical volume ---
 PV Name            /dev/xvdd2
 VG Name
 PV Size            1.00 GiB
...
```

b. Use the pvs command to report information in a more condensed form.

```
# pvs
 PV          VG  Fmt   Attr PSize PFree
 /dev/xvdb2      lvm2  a-   1.00g 1.00g
 /dev/xvdd2      lvm2  a-   1.00g 1.00g
```

c. Use the pvscan command to scan all disks for physical volumes.

```
# pvscan
  PV /dev/xvdb2                         lvm2 [1.00 GiB]
  PV /dev/xvdd2                         lvm2 [1.00 GiB]
  Total: 2 [2.00 GiB] / in use: 0 [0 ] / in no VG: 2 [2.00...
```

3. Create a volume group.

Use the vgcreate command to create a volume group named **myvolg** from the /dev/xvdb2 physical volume.

```
# vgcreate -v myvolg /dev/xvdb2
    Wiping cache of LVM-capable devices
    Adding physical volume 'dev/xvdb2' to volume group...
    Archiving volume group "myvolg" metadata (seqno 0).
    Creating volume group backup "/etc/lvm/backup/myvolg"...
  Volume group "myvolg" successfully created
```

4. Display information about the volume group.

a. Use the vgdisplay command to display attributes of the volume group.

```
# vgdisplay
 --- Volume group ---
 VG Name              myvolg
 System ID
 Format               lvm2
...
 VG Size              1020.00 MiB
...
```

b. Use the vgs command to report information in a more condensed form.

```
# vgs
 VG       #PV #LV #SN Attr   VSize    VFree
 myvolg     1   0   0 wz--n- 1020.00m 1020.00m
```

c. Use the vgscan command to scan all disks for volume groups and rebuild caches.

```
# vgscan
  Reading all physical volumes.  This may take a while...
  Found volume group "myvolg" using metadata type lvm2
```

d. Display information about the physical volumes.

```
# pvs
 PV          VG      Fmt   Attr PSize    PFree
 /dev/xvdb2  myvolg  lvm2  a-   1020.00m 1020.00m
 /dev/xvdd2          lvm2  a-      1.00g    1.00g
```

- Notice that the `/dev/xvdb2` physical volume is assigned to the **myvolg** volume group.

5. Create a logical volume.

Use the `lvcreate` command to create a 500 MB logical volume named **myvol** from the **myvolg** volume group.

```
# lvcreate -v -L 500m -n myvol myvolg
    Setting logging type to disk
    Finding volume group "myvolg"
    Archiving volume group "myvolg" metadata (seqno 1).
    Creating logical volume myvol
...
  Logical volume "myvol" created
```

6. Display information about the logical volume.

a. Use the `lvdisplay` command to display attributes of the logical volume.

```
# lvdisplay
 --- Logical volume ---
 LV Name            /dev/myvolg/myvol
 VG Name            myvolg
...
 LV Size            500.00 MiB
...
```

b. Use the `lvs` command to report information in a more condensed form.

```
# lvs
 LV    VG     Attr   LSize   Pool Origin Data% Move Log Cpy%...
 myvol myvolg -wi-a- 500.00m
```

c. Use the `lvscan` command to scan all disks for logical volumes.

```
# lvscan
  ACTIVE     '/dev/myvolg/myvol' [500.00 MiB] inherit
```

d. Display information about the physical volumes.

```
# pvs
 PV          VG      Fmt   Attr PSize    PFree
 /dev/xvdb2  myvolg  lvm2  a-   1020.00m 520.00m
 /dev/xvdd2          lvm2  a-      1.00g   1.00g
```

- Notice that the free space in the `/dev/xvdb2` physical volume has been reduced.

e. Display information about the volume group.

```
# vgs
 VG        #PV #LV #SN Attr   VSize    VFree
 myvolg      1   1   0 wz--n- 1020.00m 520.00m
```

- Notice that the free space in the **myvolg** volume group has also been reduced.

Practice 12-3: Creating a File System and Mounting a Logical Volume

Overview

In this practice, you create a file system on the logical volume and mount the logical volume.

Tasks

1. Display the block device name that was automatically created.
   a. List the /dev entries for the **myvol** logical volume.

```
# ls -l /dev/myvolg/myvol
lrwxrwxrwx.  /dev/myvolg/myvol -> ../dm-0
# ls -l /dev/mapper/myvolg-myvol
lrwxrwxrwx.  /dev/mapper/myvolg-myvol -> ../dm-0
```

   - Notice that two entries were automatically created.
   - Notice that both entries are symbolic links to /dev/dm-0.

   b. List the /dev/dm-0 entry.

```
# ls -l /dev/dm-0
brw-rw----.  /dev/dm-0
```

   - Notice that /dev/dm-0 is a block device.

2. Create a file system on the logical volume.
   a. Create an **ext4** file system on the **myvol** logical volume.

```
# mkfs.ext4 /dev/mapper/myvolg-myvol
mke2fs 1.43-WIP (20-Jun-2013)
Filesystem label=
OS type: Linux
...
Writing inode tables: done
Creating journal (8192 blocks): done
...
```

   b. Create a /myvol mount point.

```
# mkdir /myvol
```

   c. Mount the file system.

```
# mount /dev/mapper/myvolg-myvol /myvol
```

d. Display the mounted file systems.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/mapper/myvolg-myvol
               477M  2.3M   449M    1%  /myvol
```

3. Update the file systems mount table.

Use the `vi` editor to add the following line to `/etc/fstab`.

```
/dev/mapper/myvolg-myvol /myvol  ext4  defaults  0  0
```

Practice 12-4: Backing Up Volume Group Metadata

Overview

In this practice, you view the LVM configuration file, view the automatically created metadata backups and archives, and manually create a volume group metadata backup.

Tasks

1. View the configuration of metadata backups and archiving.
   a. Use the `less` command to view the logical volume configuration, `/etc/lvm/lvm.conf`.

```
# less /etc/lvm/lvm.conf
# This is an example configuration file for the LVM2 system.
...
```

   b. Use the search character, `/`, and search for the word `backup`.

```
/backup
```

   - The search results in displaying the following screenshot:

```
root@host03:/etc/lvm
File  Edit  View  Search  Terminal  Help
# Configuration of metadata backups and archiving.  In LVM2 when we
# talk about a 'backup' we mean making a copy of the metadata for the
# *current* system.  The 'archive' contains old metadata configurations.
# Backups are stored in a human readeable text format.
backup {

    # Should we maintain a backup of the current metadata configuration ?
    # Use 1 for Yes; 0 for No.
    # Think very hard before turning this off!
    backup = 1

    # Where shall we keep it ?
    # Remember to back up this directory regularly!
    backup_dir = "/etc/lvm/backup"

    # Should we maintain an archive of old metadata configurations.
    # Use 1 for Yes; 0 for No.
    # On by default.  Think very hard before turning this off.
    archive = 1

    # Where should archived files go ?
    # Remember to back up this directory regularly!
    archive_dir = "/etc/lvm/archive"
:
```

   - Notice that metadata backups are enabled (`backup = 1`) and backups are stored in the `/etc/lvm/backup` directory.
   - Also notice that archives are enabled (`archive = 1`) and the archives are stored in the `/etc/lvm/archive` directory.

c. Press `q` to quit the `less` command.

```
:q
#
```

d. You can also use the `lvm dumpconfig` command to view the configuration settings.

```
# lvm dumpconfig
config {
        checks=1
        abort_on_errors=0
        profile_dir="/etc/lvm/profile"
}
...
backup {
        backup=1
        backup_dir="/etc/lvm/backup"
        archive=1
        archive_dir="/etc/lvm/archive"
...
```

2. View the metadata backup and archive files.

a. Use the `cd` command to change to the `/etc/lvm/backup` directory. Use the `ls` command to display the contents of the directory.

```
# cd /etc/lvm/backup
# ls
myvolg
```

- Notice the backup of the `myvolg` volume group was created automatically.

b. Use the `less` command to view the contents of the `myvolg` metadata backup file.

```
# less myvolg
...
description = "Created *after* executing 'lvcreate -v -L 500...
...
myvolg {
...
       physical_volumes {
               pv0 {
...
                        device = "/dev/xvdb2"
...
       logical_volumes {
               myvol {
...
```

- Notice the description states the backup file was created “after” executing the lvcreate command string.
- Also notice the metadata backup file includes information on the physical and logical volumes.

c. Use the cd command to change to the /etc/lvm/archive directory. Use the ls command to display the contents of the directory.

```
# cd /etc/lvm/archive
# ls
myvolg_00000-...        myvolg_00001-...
```

- Notice the archive files for the myvolg volume group were created automatically.

d. Use the less command to view the contents of the myvolg-00000* archive file.

```
# less myvolg_00000*
...
description = "Created *before* executing 'vgcreate -v myvolg...
...
myvolg {
...
       physical_volumes {
               pv0 {
...
                        device = "/dev/xvdb2"
...
```

- Notice the description states the archive file was created “before” executing the vgcreate command string.

e. Use the less command to view the contents of the myvolg-00001* archive file.

```
# less myvolg_00001*
...
description = "Created *before* executing 'lvcreate -v -L 500...
...
myvolg {
...
       physical_volumes {
               pv0 {
...
                        device = "/dev/xvdb2"
...
```

- Notice the description states the archive file was created “before” executing the lvcreate command string.

3. Create a metadata backup of the `myvolg` volume group.

   a. Use the `vgcfgbackup` command to back up the metadata for the `myvolg` volume group.

      - Include the `–f file_today` argument to the `vgcfgbackup` command.

```
# vgcfgbackup –f file_today myvolg
Volume group "myvolg" successfully backed up.
```

   b. Use the `diff` command to display the differences in the newly created backup file and the existing `myvolg` metadata backup file.

      - Note that the newly created backup was created in the current directory.

```
# diff file_today /etc/lvm/backup/myvolg
...
< description = "vgcfgbackup –f file_today myvolg"
---
> description = "Created *after* executing 'lvcreate –v –L ...
...
```

      - Notice that only the description and the creation information are different.

Practice 12-5: Creating a Logical Volume Snapshot

Overview

In this practice, you create a snapshot volume, mount the snapshot, and remove the snapshot volume.

Tasks

1. Create a snapshot volume.
   a. Copy `/boot/init*` to the mounted logical volume, `/myvol`.

```
# cp /boot/init* /myvol
# ls /myvol
initramfs-2.6.32-431.el6.x86_64.img
initramfs-3.8.13-16.2.1.el6uek.x86_64.img
lost+found
```

   b. Create a 500 MB snapshot named `myvol-snapshot` of the **myvol** logical volume.

```
# lvcreate -L 500m –s –n myvol-snapshot myvolg/myvol
  Logical volume "myvol-snapshot" created
```

   c. Use the `ls –l` command to list the contents of the `/etc/lvm/backup` directory and the contents of the `/etc/lvm/archive` directory.

```
# ls –l /etc/lvm/backup
-rw-------. <date_time> myvolg
# ls –l /etc/lvm/archive
-rw-------. <date_time> file_today
-rw-------. <date_time> myvolg_00000...
-rw-------. <date_time> myvolg_00001...
-rw-------. <date_time> myvolg_00002...
```

   - Notice a new `myvolg` backup file was automatically created when the snapshot was created (note the time stamp).
   - Also notice a new archive file was automatically created (note the time stamp on the `myvolg_00002*` file).

   d. List the logical volumes.

```
# lvs
 LV             VG     Attr    LSize   Pool Origin Data% Move...
 myvol          myvolg owi-aos 500.00m
 myvol-snapshot myvolg swi-a-s 500.00m      myvol    0.00
```

e. List the contents of the /dev/myvolg and /dev/mapper directories.

```
# ls -l /dev/myvolg
lrwxrwxrwx.  myvol -> ../dm-0
lrwxrwxrwx.  myvol-snapshot -> ../dm-1
# ls -l /dev/mapper
crw-rw----.  control
lrwxrwxrwx.  myvolg-myvol -> ../dm-0
lrwxrwxrwx.  myvolg-myvol-real -> ../dm-2
lrwxrwxrwx.  myvolg-myvol--snapshot -> ../dm-1
lrwxrwxrwx.  myvolg-myvol--snapshot-cow -> ../dm-3
```

2. Mount the snapshot.

a. Mount the snapshot onto /mnt.

```
# mount -t ext4 /dev/myvolg/myvol-snapshot /mnt
```

b. Display the mounted file systems.

```
# df -h
Filesystem     Size  Used  Avail   Use%  Mounted on
...
/dev/mapper/myvolg-myvol
               477M   41M   411M     9%  /myvol
/dev/mapper/myvolg-myvol--snapshot
               477M   41M   411M     9%  /mnt
```

c. List the files on /mnt.

```
# ls /mnt
initramfs-2.6.32-431.el6.x86_64.img
initramfs-3.8.13-16.2.1.el6uek.x86_64.img
lost+found
```

- Notice that these are the same files that were copied onto /myvol.

d. Remove the files on /myvol.

```
# rm /myvol/init*
rm: remove regular file '/myvol/init...'? y
...
```

e. List the files on /mnt.

```
# ls /mnt
initramfs-2.6.32-431.el6.x86_64.img
initramfs-3.8.13-16.2.1.el6uek.x86_64.img
lost+found
```

- Notice that these files are still present.
- Removing files from the original volume does not change the snapshot's content.

3. Remove the snapshot.

   a. Unmount the snapshot from `/mnt`.

```
# umount /mnt
```

   b. Use the `lvremove` command to remove the snapshot.

   - Answer **y** when asked, "Do you really want to ..."

```
# lvremove -v myvolg/myvol-snapshot
    Using logical volume(s) on command line
Do you really want to remove active logical volume myvol-
snapshot? [y/n]: y
...
  Logical volume "myvol-snapshot" successfully removed
```

   c. List the logical volumes.

```
# lvs
 LV             VG     Attr   LSize   Pool Origin Data% Move...
 myvol          myvolg -wi-ao 500.00m
```

   - Notice that the **myvol-snapshot** logical volume has been deleted.

   d. List the contents of the `/dev/myvolg` and `/dev/mapper` directories.

```
# ls –l /dev/myvolg
lrwxrwxrwx.  myvol -> ../dm-0
# ls –l /dev/mapper
crw-rw----.  control
lrwxrwxrwx.  myvolg-myvol -> ../dm-0
```

   - Notice that the "snapshot" entries have been deleted.

   e. Use the `ls –l` command to list the contents of the `/etc/lvm/backup` directory and the contents of the `/etc/lvm/archive` directory.

```
# ls –l /etc/lvm/backup
-rw-------. <date_time> myvolg
# ls –l /etc/lvm/archive
-rw-------. <date_time> file_today
-rw-------. <date_time> myvolg_00000...
-rw-------. <date_time> myvolg_00001...
-rw-------. <date_time> myvolg_00002...
-rw-------. <date_time> myvolg_00003...
```

   - Notice a new `myvolg` backup file was automatically created when the snapshot was removed (note the time stamp).
   - Also notice a new archive file was automatically created (note the time stamp on the `myvolg_00003*` file).

Practice 12-6: Increasing the Capacity of a Logical Volume

Overview

In this practice, you increase the size of a logical volume and add a physical volume to a volume group.

Tasks

1. Increase the size of a logical volume.
   a. List the volume group.

```
# vgs
 VG        #PV #LV #SN Attr   VSize    VFree
 myvolg      1   1   0 wz--n- 1020.00m 520.00m
```

   - Notice that the **myvolg** volume group has 520 MB of free space.

   b. List the logical volume.

```
# lvs
 LV              VG     Attr   LSize   Pool Origin Data% Move...
 myvol           myvolg -wi-ao 500.00m
```

   - Notice that the **myvol** logical volume is 500 MB in size.

   c. Display the mounted file systems.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/mapper/myvolg-myvol
               477M  2.3M   449M    1%  /myvol
```

   - Notice that the size of the file system is 477 MB.

   d. Use the `lvextend` command to increase the size of the **myvolg/myvol** logical volume and the file system by 500 MB.

```
# lvextend -L +500M -r myvolg/myvol
  Extending logical volume myvol to 1000.00 MiB
  Logical volume myvol successfully resized
resize2fs 1.43-WIP (20-Jun-2013)
Filesystem at /dev/mapper/myvolg-myvol is mounted on /myvol; on-
line resize required
...
The filesystem on /dev/mapper/myvolg-myvol is now 1024000 blocks
long.
```

   - The `-r` option causes the file system to be resized.

e. List the volume group.

```
# vgs
 VG        #PV #LV #SN Attr   VSize     VFree
 myvolg      1   1   0 wz--n- 1020.00m 20.00m
```

- Notice that the **myvolg** volume group now has only 20 MB of free space.

f. List the logical volume.

```
# lvs
 LV              VG      Attr   LSize    Pool Origin Data% Move...
 myvol           myvolg -wi-ao 1000.00m
```

- Notice that the **myvol** logical volume is now 1,000 MB in size.

g. Display the mounted file systems.

```
# df -h
Filesystem    Size  Used  Avail  Use%  Mounted on
...
/dev/mapper/myvolg-myvol
              961M  2.5M   914M    1%  /myvol
```

- Notice that the size of the file system is now 961 MB.

h. Use the `ls -l` command to list the contents of the `/etc/lvm/backup` directory and the contents of the `/etc/lvm/archive` directory.

```
# ls -l /etc/lvm/backup
-rw-------. <date_time> myvolg
# ls -l /etc/lvm/archive
-rw-------. <date_time> file_today
-rw-------. <date_time> myvolg_00000...
-rw-------. <date_time> myvolg_00001...
-rw-------. <date_time> myvolg_00002...
-rw-------. <date_time> myvolg_00003...
-rw-------. <date_time> myvolg_00004...
```

- Notice a new `myvolg` backup file was automatically created when the logical volume was extended (note the time stamp).
- Also notice a new archive file was automatically created (note the time stamp on the `myvolg_00004*` file).

2. Add a physical volume to a volume group.

a. List the physical volumes.

```
# pvs
 PV          VG      Fmt   Attr PSize    PFree
 /dev/xvdb2  myvolg  lvm2  a--  1020.00m 20.00m
 /dev/xvdd2          lvm2  a--     1.00g  1.00g
```

- Notice that the `/dev/xvdd2` physical volume is not assigned to a volume group.

b. List the volume group.

```
# vgs
  VG       #PV #LV #SN Attr   VSize    VFree
  myvolg     1   1   0 wz--n- 1020.00m 20.00m
```

- Notice that the **myvolg** volume group is 1020 MB in size.

c. Use the `vgextend` command to add the `/dev/xvdd2` physical volume to the **myvolg** volume group.

```
# vgextend -v myvolg /dev/xvdd2
    Checking for volume group "myvolg"
...
    Adding physical volume '/dev/xvdd2' to volume group 'myvolg'
...
  Volume group "myvolg" successfully extended
```

d. List the physical volumes.

```
# pvs
  PV          VG      Fmt   Attr PSize    PFree
  /dev/xvdb2  myvolg  lvm2  a--  1020.00m   20.00m
  /dev/xvdd2  myvolg  lvm2  a--  1020.00m 1020.00m
```

- Notice that the `/dev/xvdd2` physical volume is now assigned to the **myvolg** volume group.

e. List the volume group.

```
# vgs
  VG       #PV #LV #SN Attr   VSize VFree
  myvolg     2   1   0 wz--n- 1.99g 1.02g
```

- Notice that the **myvolg** volume group now has two PVs (physical volumes).
- Notice that VSize and VFree have increased.

f. List the logical volume.

```
# lvs
  LV              VG     Attr   LSize    Pool Origin Data% Move...
  myvol           myvolg -wi-ao 1000.00m
```

g. Use the `ls -l` command to list the contents of the `/etc/lvm/backup` directory and the contents of the `/etc/lvm/archive` directory.

```
# ls -l /etc/lvm/backup
-rw-------. <date_time> myvolg
# ls -l /etc/lvm/archive
-rw-------. <date_time> file_today
-rw-------. <date_time> myvolg_00000...
-rw-------. <date_time> myvolg_00001...
-rw-------. <date_time> myvolg_00002...
-rw-------. <date_time> myvolg_00003...
-rw-------. <date_time> myvolg_00004...
-rw-------. <date_time> myvolg_00005...
```

- Notice a new `myvolg` backup file was automatically created when the physical volume was added to the volume group (note the time stamp).
- Also notice a new archive file was automatically created (note the time stamp on the `myvolg_00005*` file).

3. Increase the size of the **myvolg/myvol** logical volume and the file system.

a. Increase the size of the **myvolg/myvol** logical volume and the file system by 500 MB.

```
# lvextend -L +500M -r myvolg/myvol
  Extending logical volume myvol to 1.46 GiB
  Logical volume myvol successfully resized
resize2fs 1.43-WIP (20-May-2013)
...
The file system on /dev/mapper/myvolg-myvol is now 1536000
blocks long.
```

b. List the logical volume.

```
# lvs
 LV              VG     Attr   LSize   Pool Origin Data% Move...
 myvol           myvolg -wi-ao 1.46g
```

c. Display the mounted file systems.

```
# df -h
Filesystem    Size  Used  Avail  Use%  Mounted on
...
/dev/mapper/myvolg-myvol
              1.5G  2.8M   1.4G    1%  /myvol
```

- Notice that the size of the file system is now 1.5 GB.

d. Use the `ls -l` command to list the contents of the `/etc/lvm/backup` directory and the contents of the `/etc/lvm/archive` directory.

```
# ls -l /etc/lvm/backup
-rw-------. <date_time> myvolg
# ls -l /etc/lvm/archive
-rw-------. <date_time> file_today
-rw-------. <date_time> myvolg_00000...
-rw-------. <date_time> myvolg_00001...
-rw-------. <date_time> myvolg_00002...
-rw-------. <date_time> myvolg_00003...
-rw-------. <date_time> myvolg_00004...
-rw-------. <date_time> myvolg_00005...
-rw-------. <date_time> myvolg_00006...
```

- Notice a new `myvolg` backup file was automatically created when the logical volume was extended (note the time stamp).
- Also notice a new archive file was automatically created (note the time stamp on the `myvolg_00006*` file).

Practice 12-7: Restoring Volume Group Metadata

Overview

In this practice, you restore volume group metadata from a backup.

Tasks

1. View the current physical volumes, volume group, and logical volumes configuration.
   a. Use the `pvs` command to display information about the physical volumes.

```
# pvs
 PV          VG      Fmt  Attr PSize    PFree
 /dev/xvdb2  myvolg  lvm2 a-   1020.00m      0
 /dev/xvdd2  myvolg  lvm2 a-   1020.00m 540.00m
```

   b. Use the `vgs` command to display information about the volume groups.

```
# vgs
 VG       #PV #LV #SN Attr   VSize  VFree
 myvolg     2   1   0 wz--n- 1.99g  540.00m
```

   c. Use the `lvs` command to display information about the logical volumes.

```
# lvs
 LV    VG     Attr    LSize   Pool Origin Data% Move Log Cpy%...
 myvol myvolg -wi-ao- 1.46g
```

2. Restore the volume group metadata from a backup.
   a. Use the `vgcfgrestore` command to restore the `myvolg` volume group metadata.

```
# vgcfgrestore myvolg
Restored volume group myvolg
```

   b. Repeat Task 1a, 1b, 1c (execute the `pvs`, `vgs`, `lvs` commands).

```
# pvs
 PV          VG      Fmt  Attr PSize    PFree
 /dev/xvdb2  myvolg  lvm2 a-   1020.00m      0
 /dev/xvdd2  myvolg  lvm2 a-   1020.00m 540.00m
# vgs
 VG       #PV #LV #SN Attr   VSize  VFree
 myvolg     2   1   0 wz--n- 1.99g  540.00m
# lvs
 LV    VG     Attr    LSize   Pool Origin Data% Move Log Cpy%...
 myvol myvolg -wi-ao- 1.46g
```

   - Notice that the LVM information is the same as the output in Task 1.

Practice 12-8: Creating a RAID Device

Overview

In this practice, you remove the logical volume and LVM entities, create a RAID array device, and create a file system and mount the RAID device. You remove the RAID device at the end of the practice.

Tasks

1. Remove the logical volume, volume group, and physical volume.
   a. Unmount the **myvol** logical volume.

```
# umount /myvol
```

   b. Use the `lvremove` command to remove the **myvol** logical volume.
      - Answer **y** when asked, "Do you really want to ..."

```
# lvremove myvolg/myvol
Do you really want to remove active logical volume myvol? [y/n]:
y
  Logical volume "myvol" successfully removed
```

   c. Use the `vgremove` command to remove the **myvolg** volume group.

```
# vgremove myvolg
  Volume group "myvolg" successfully removed
```

   d. Use the `pvremove` command to remove the physical volumes.

```
# pvremove /dev/xvdb2 /dev/xvdd2
  Labels on physical volume "/dev/xvdb2" successfully wiped
  Labels on physical volume "/dev/xvdd2" successfully wiped
```

   e. Remove the `/myvol` mount point.

```
# rmdir /myvol
```

   f. Using the `vi` editor, remove the following line from `/etc/fstab`.

```
# vi /etc/fstab
/dev/mapper/myvolg-myvol /myvol  ext4  defaults  0  0
```

2. Create a new RAID array.
   - Disk partitions used in a RAID device should have the partition type changed to **fd** (Linux raid autodetect). This step is omitted however in this practice.

a. Use the `mdadm` command to create a RAID-1 device, `/dev/md0`, from the `/dev/xvdb2` and `/dev/xvdd2` partitions.

- Answer **y** when asked, "Continue creating array?"

```
# mdadm --create /dev/md0 --level=1 --raid-devices=2 /dev/xvdb2
/dev/xvdd2
...
Continue creating array? y
mdadm: Defaulting to version 1.2 metadata
mdadm: array /dev/md0 started.
```

b. View the `/proc/mdstat` file to check the status of the MD RAID devices.

```
# cat /proc/mdstat
Personalities : [raid1]
md0 : active raid1 xvdd2[1] xvdb2[0]
      1047616 blocks supper 1.2 [2/2] [UU]
      [==========>.......]  resync = ...
unused devices: <none>
```

c. Run the previous command again to confirm the resync has completed.

```
# cat /proc/mdstat
Personalities : [raid1]
md0 : active raid1 xvdd2[1] xvdb2[0]
      1047616 blocks supper 1.2 [2/2] [UU]
unused devices: <none>
```

- Ensure the resync is complete before continuing.

d. Use the `mdadm` command to view information about the RAID device.

```
# mdadm --query /dev/md0
/dev/md0: 1023.06MiB raid1 2 devices, 0 spares. Use mdadm --
detail for more detail.
# mdadm --detail /dev/md0
/dev/md0:
        Version : 1.2
  Creation Time : Tue Dec 10...
     Raid Level : raid1
     Array Size : 1047616 (1023.23 MiB 1072.76 MB)
  Used Dev Size : 1047616 (1023.243MiB 1072.76 MB)
   Raid Devices : 2
  Total Devices : 2
...
    Number   Major   Minor   RaidDevice   State
       0      202      18        0        active sync /dev/xvdb2
       1      202      50        1        active sync /dev/xvdd2
```

3. Create a file system on the RAID device.

a. Create an **ext4** file system on `/dev/md0`.

```
# mkfs.ext4 /dev/md0
mke2fs 1.43-WIP (20-Jun-2013)
Filesystem label=
OS type: Linux
...
Writing inode tables: done
Creating journal (4096 blocks): done
...
```

b. Create a mount point named `/raid`.

```
# mkdir /raid
```

c. Mount the file system.

```
# mount /dev/md0 /raid
```

d. Display the mounted file systems.

```
# df -h
Filesystem    Size  Used  Avail  Use%  Mounted on
...
/dev/md0      991M  1.3M   939M   1%   /raid
```

4. Create the **mdadm** configuration file, `/etc/mdadm.conf`.

Use the `vi` editor to create `/etc/mdadm.conf` and add the following entry.

- This step does not need to be performed in this practice. You are directed to remove this file in the next task.

```
# vi /etc/mdadm.conf
ARRAY /dev/md0 devices=/dev/xvdb2,/dev/xvdd2
```

- With this file, the RAID array is properly detected and initialized after a reboot.

5. Remove the RAID array.

a. Remove the `/etc/mdadm.conf` file.

```
# rm /etc/mdadm.conf
rm: remove regular file '/etc/mdadm.conf'? y
```

b. Unmount the **raid** volume.

```
# umount /raid
```

c. Use the `mdadm` command to deactivate the array and release all resources.

```
# mdadm --stop /dev/md0
mdadm: stopped /dev/md0
```

d. Display the attributes of the `/dev/xvdb3` block device.

```
# blkid | grep raid
/dev/xvdb2: UUID=”...” UUID_SUB=”...”
LABEL=”host03.example.com:0” TYPE=”linux_raid_member”
/dev/xvdd2: UUID=”...” UUID_SUB=”...”
LABEL=”host03.example.com:0” TYPE=”linux_raid_member”
```

- Notice `/dev/xvdb2` and `/dev/xvdd2` have a type of “`linux_raid_member`.”

e. Use the `mdadm` command to overwrite the **md** superblock on `/dev/xvdb2` and `/dev/xvdd2`.

```
# mdadm --zero-superblock /dev/xvdb2
# mdadm --zero-superblock /dev/xvdd2
```

f. Display the attributes of `/dev/xvdb2` and `/dev/xvdd2` to ensure that the **md** superblock was overwritten.

```
# blkid /dev/xvdb2
# blkid /dev/xvdd2
```

- No output indicates the superblock was overwritten.

Practices for Lesson 13: Network Configuration

Chapter 13

Practices for Lesson 13: Network Configuration

Practices Overview

In these practices, you:

- Configure the `eth1` network interface by editing network interface configuration files
- Use NetworkManager to configure network interfaces
- Use the `system-config-network` text-based utility
- Access the Public Yum Repository from **host03** VM
- Simulate upgrading your system (you *do not* actually perform the upgrade)

Practice 13-1: Configuring the `eth1` Network Interface

Overview

In this practice, you:

- Display the available network interfaces on your system
- View the network interface configuration files
- Configure a static IP address for the `eth1` network interface
- Update your `/etc/hosts` file
- Create a default gateway and display your route table
- Ensure that connectivity to **dom0** and the other VM guests

Assumptions

You are the `root` user on **host03** VM.

If you were unable to complete the OS install and FirstBoot configuration on **host03** in Practice 3, substitute **host02** for **host03** in this practice and all future practices in this course. Use the IP address of `192.168.1.102` for the `eth1` interface if configuring **host02**.

Tasks

1. Use the `ifconfig` command to display your available network interfaces.

```
# ifconfig
eth0      Link encap:Ethernet HWaddr 00:16:3E:00:01:03
          inet addr:192.0.2.103 ...
          inet6 addr: ...
...
eth1      Link encap:Ethernet HWaddr 00:16:3E:00:02:03
          inet6 addr:...
...

lo        Link encap:Local Loopback
          inet addr:127.0.0.1
...
```

- Notice that you have two Ethernet interfaces (`eth0` and `eth1`) and the loopback interface (`lo`).
- The `eth0` Ethernet interface has an IPv4 address, but `eth1` does not.

2. View the network interface configuration files.

   a. Use the `cd` command to change to the `/etc/sysconfig/network-scripts` directory.

```
# cd /etc/sysconfig/network-scripts
```

b. Use the `ls` command to view the contents of this directory.

```
# ls
ifcfg-eth0  ifdown-ipv6  ifup         ifup-plip...
ifcfg-eth1  ifdown-isdn  ifup-aliases  ifup-plush...
ifcfg-lo    ifdown-post...
...
```

- Notice that you have a configuration file for `eth0`, `ifcfg-eth0`.
- Notice that you have a configuration file for `eth1`, `ifcfg-eth1`.
- Notice that you have a configuration file for the loopback interface, `ifcfg-lo`.
- Several interface control scripts exist in this directory to activate and deactivate network interfaces.

3. Configure `eth1` with a static IP address of `192.168.1.103`.

Use the `vi` editor to edit the `ifcfg-eth1` file as follows (the use of quotes around the assigned values is optional):

```
DEVICE=eth1
BOOTPROTO=none
ONBOOT=yes
USERCTL=no
IPV6INIT=no
PEERDNS=yes
TYPE=Ethernet
HWADDR=00:16:3e:00:02:03
IPADDR=192.168.1.103
NETMASK=255.255.255.0
BROADCAST=192.168.1.255
UUID=... (leave this as is)
NM_CONTROLLED=yes
```

4. Specify routing and host information for all network interfaces.

a. Use the `cd` command to change to the `/etc/sysconfig` directory.

```
# cd /etc/sysconfig
```

b. If necessary, use the `vi` editor to edit the `network` file as follows:

- This file might not require editing.

```
NETWORKING=yes
HOSTNAME=host03.example.com
GATEWAY=192.0.2.1
```

5. Use the `vi` editor to edit the `/etc/hosts` file as follows:

```
127.0.0.1       localhost.localdomain localhost
192.0.2.1       example.com                dom0
192.0.2.101     host01.example.com         host01
192.0.2.102     host02.example.com         host02
192.0.2.103     host03.example.com         host03
```

6. Use the `service` command to restart the network.

```
# service network restart
Shutting down interface eth0...
Shutting down interface eth1...
Shutting down loopback interface...
Bringing up loopback interface...
Bringing up interface eth0...
Bringing up interface eth1...
...
```

7. Use the `ifconfig` command to display the status of the interfaces.

```
# ifconfig
eth0      Link encap:Ethernet HWaddr 00:16:3E:00:01:03
          inet addr:192.0.2.103 Bcast:192.0.2.255 Mask:...
          inet6 addr: ...
          UP BROADCAST RUNNING MULTICAST  MTU:1500...
...
eth1      Link encap:Ethernet HWaddr 00:16:3E:00:02:03
          inet addr:192.168.1.103 Bcast:192.168.1.255 Mask:...
          inet6 addr: ...
          UP BROADCAST RUNNING MULTICAST  MTU:1500...
...
```

- Notice that both `eth0` and `eth1` now have IP addresses.

8. Use the `netstat -r` command (or `route`) to display the route table.

```
# netstat -r
Destination  Gateway      Genmask        Flags ...  Iface
default      example.com  0.0.0.0        UG         eth0
192.0.2.0    *            255.255.255.0  U          eth0
192.168.1.0  *            255.255.255.0  U          eth1
```

- Notice that all packets destined for the `192.168.1` subnet use the `eth1` interface.
- Notice that all packets destined for the `192.0.2` subnet use the `eth0` interface.
- Notice that all other packets are routed through the `192.0.2.1` default gateway (`example.com`), via `eth0`.

9. Use the `ping` command to verify that you can communicate to **dom0** and the other VM guests.
   - Press **Ctrl + C** to kill the `ping` command.

```
# ping dom0
64 bytes from example.com (192.0.2.1)...
CTRL-C
# ping host01
64 bytes from host01.example.com (192.0.2.101)...
CTRL-C
# ping host02
64 bytes from host02.example.com (192.0.2.102)...
CTRL-C
```

Practice 13-2: Using NetworkManager

Overview

In this practice, you:

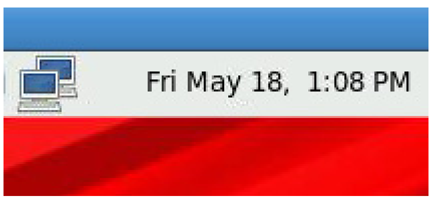
- Use the NetworkManager utility to disable and enable network connections
- View the configuration capabilities of NetworkManager
- Run the `nm-tool` utility that is included with NetworkManager

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Install NetworkManager if necessary.
   - NetworkManager is installed and running if you see this computer screen icon on the GNOME panel as follows:



   a. Use the `rpm` command to verify that the NetworkManager package is installed.

```
# rpm -qa | grep -i networkmanager
NetworkManager-...
NetworkManager-glib-...
NetworkManager-gnome-...
```

   b. If NetworkManager is not installed, use the `yum` command to install the package.

```
# yum install NetworkManager
...
```

   c. Use the `service` command to verify that NetworkManager is running.

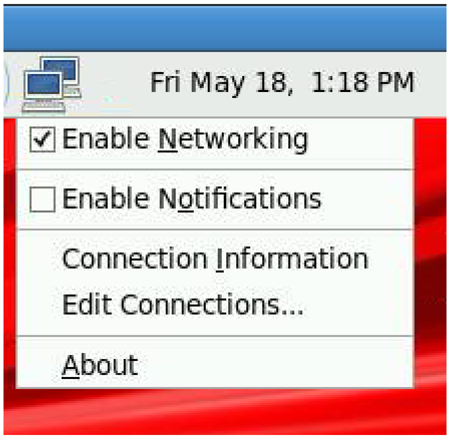
```
# service NetworkManager status
NetworkManager (pid ...) is running...
```

   d. If NetworkManager is not running, use the `service` command to start it.

```
# service NetworkManager start
...
```

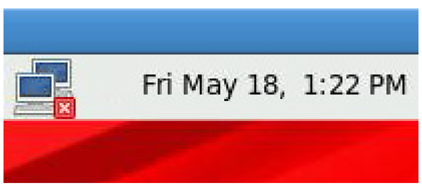
2. Use NetworkManager to enable/disable networking.

   a. Right-click the computer screens icon to display the following drop-down menu:



   b. Deselect the **Enable Networking** check box to disable networking.

      - Notice that an X in red appears on the icon, as shown:



   c. From a command window, use the `ifconfig` command to display the network status.

```
# ifconfig
lo        Link encap: Local Loopback
...
```

      - Notice that `eth0` and `eth1` interfaces do not appear in the output.

   d. Right-click the computer screens icon to display the drop-down menu.

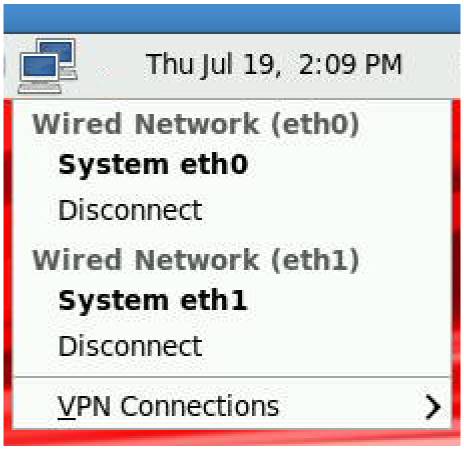
   e. Click **Enable Networking**. Ensure that the check box is selected and the X in red disappears.

   f. Run the `ifconfig` command again to ensure that the `eth0` and `eth1` interfaces appear.

```
# ifconfig
eth0      Link encap:Ethernet HWaddr 00:16:3E:00:01:03
...
eth1      Link encap:Ethernet HWaddr 00:16:3E:00:02:03
...
lo        Link encap:Local Loopback
...
```

      - Notice that the `eth0` and `eth1` interfaces now appear.

g. Click the computer screens icon to display the following drop-down menu:



- Notice that **System eth0** is in bold font, meaning it is connected.
- Notice that **System eth1** is in bold font, meaning it is also connected.

h. Click **Disconnect** under **System eth0** to disconnect `eth0`. The menu now appears as follows:

- Notice that **Wired Network (eth0)** is disconnected, but **System eth0** (not in bold) is listed under the **Available** heading.

i. From a command window, use the `ifconfig` command to display the network status.

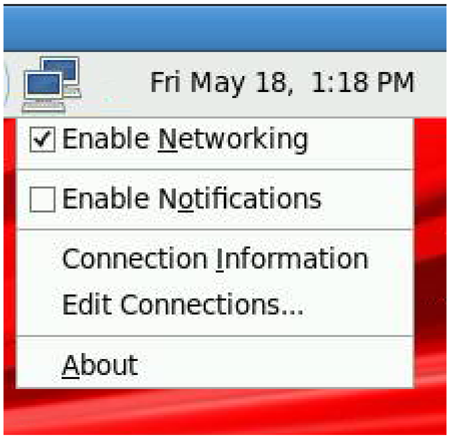
```
# ifconfig
...
```

- Notice that both `eth0` and `eth1` are listed, but `eth0` does not include an IPv4 address.

j. Click the computer screens icon and click **System eth0** to re-connect the `eth0` interface.

3. Use NetworkManager to view network connections.

   a. Right-click the computer screens icon to display the following drop-down menu:



   b. Click **Edit Connections** to display the following screen:

- Do not run the following command but note that you can also access this Network Connections window by entering `nm-connection-editor&` from a command line.

```
# nm-connection-editor&
```

c. Select **System eth0**, then click **Edit**, and then select the **IPv4 Settings** tab to display the following screen:

- Notice that this information displayed is included in the configuration file, `/etc/sysconfig/network-scripts/ifcfg-eth0`.
- Any updates made from this screen are written to the configuration file.

d. Do not make any changes; click **Cancel** after viewing the different tabs and settings.

e. Click **Close** to close the Network Connections window.

4. Run the `nm-tool` utility that is included with NetworkManager.

```
# nm-tool
NetworkManager Tool
State: connected
- Device: eth0 [System eth0] --
...
```

Practice 13-3: Using the `system-config-network` Utility

Overview

In this practice, you use the `system-config-network` text-based utility to configure the network interfaces.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Install `system-config-network` if necessary.
   a. Use the `rpm` command to verify that the `system-config-network` package is installed.

```
# rpm -qa | grep system-config-network
system-config-network-tui-...
```

   b. If the utility is not installed, use the `yum` command to install the package.

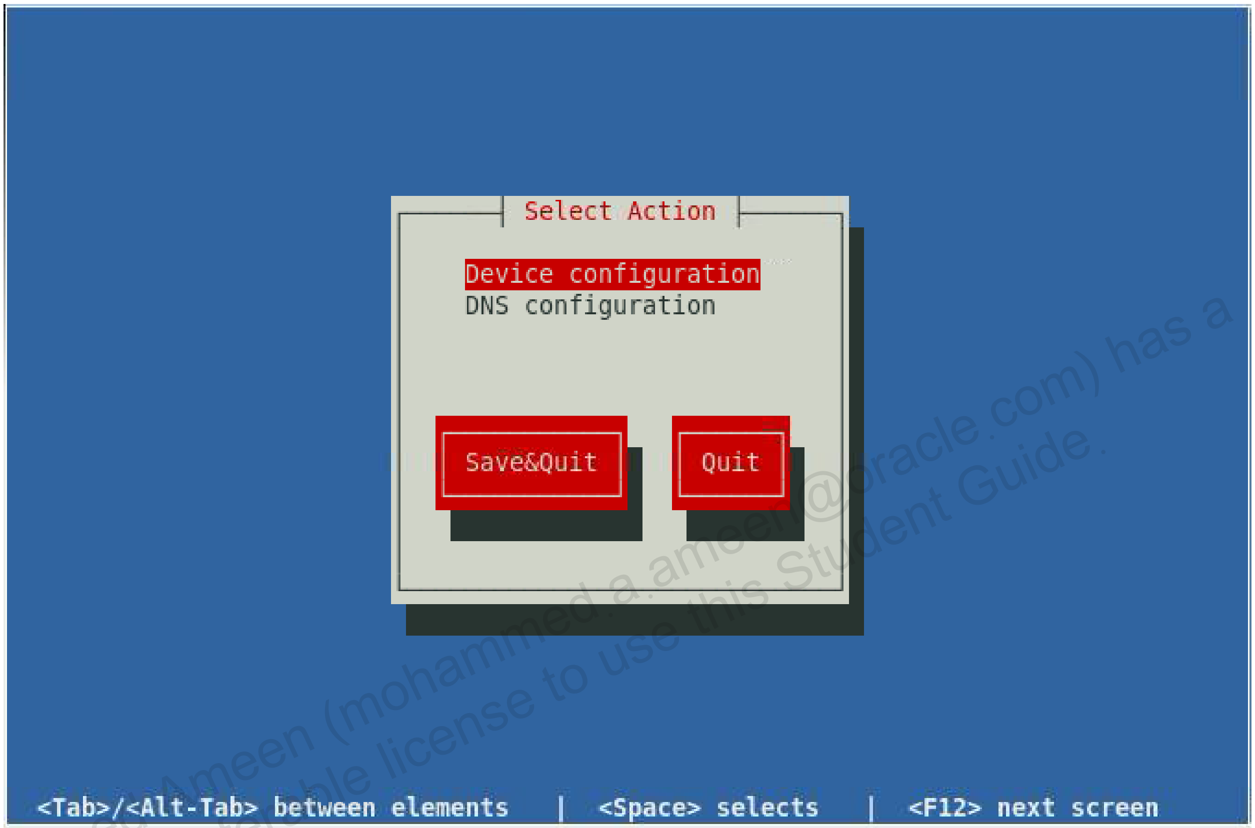
```
# yum install system-config-network-tui
...
```

2. Use `system-config-network` to configure network interfaces.

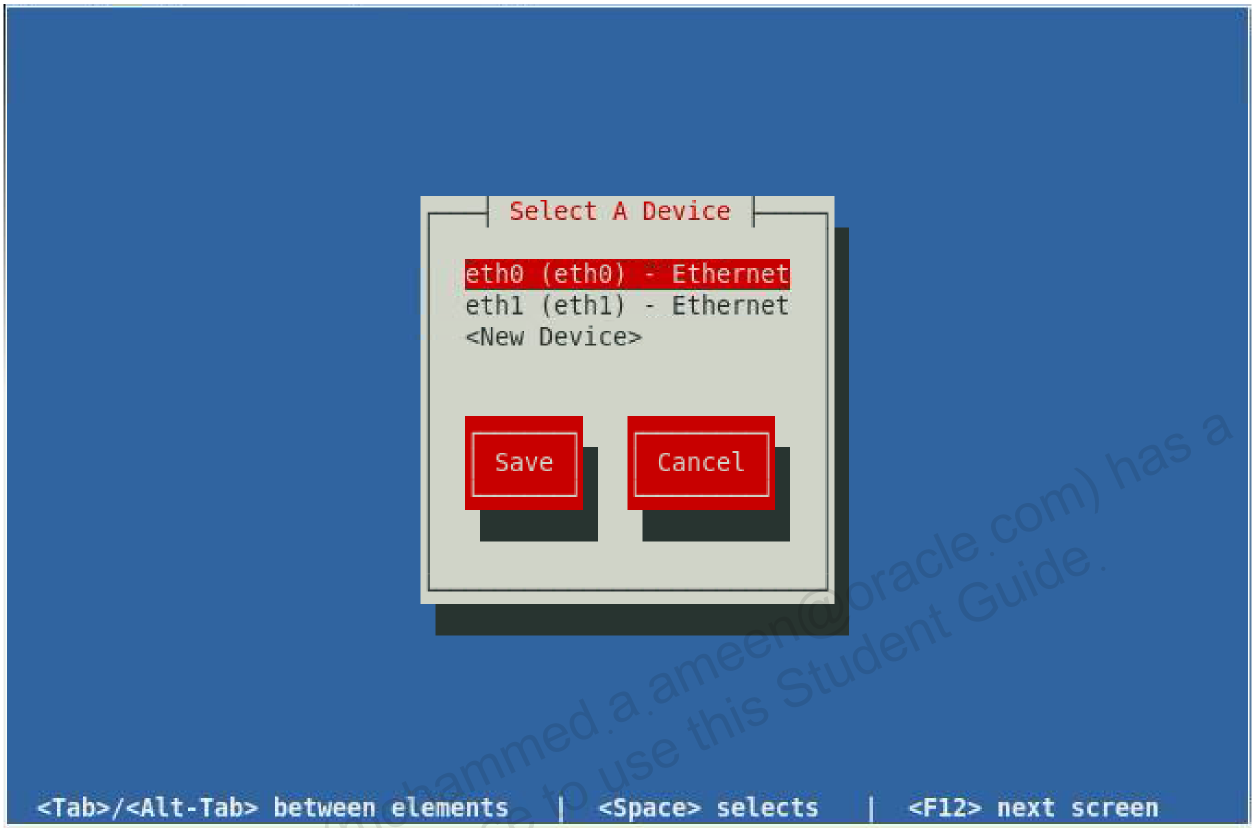
   a. Enter `system-config-network` from the command line.

```
# system-config-network
```

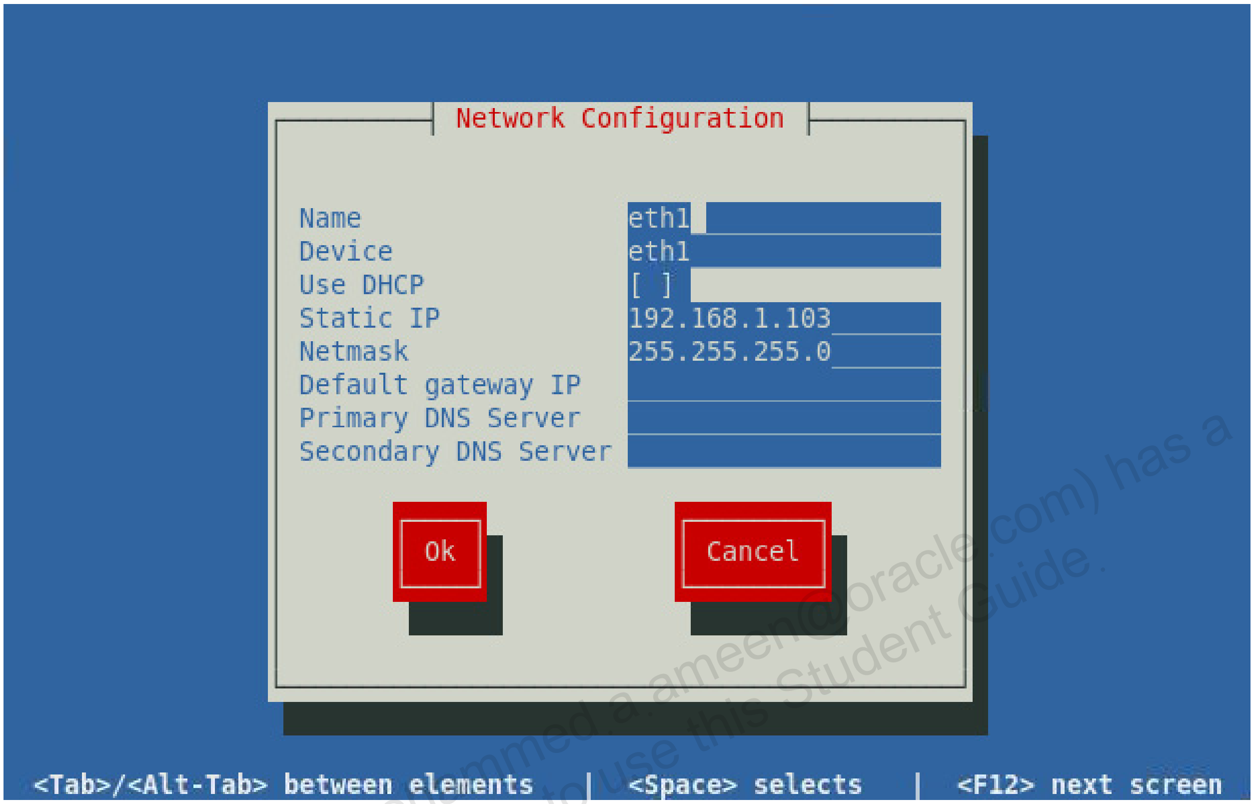
- The screen appears as follows:



b. Use the up/down arrows to select **Device configuration** and press Enter. The screen appears as follows:



c. Use the up/down arrows to select **eth1 (eth1) - Ethernet** and press Enter. The screen appears as follows:



- Notice that this information displayed is included in the configuration file, `/etc/sysconfig/network-scripts/ifcfg-eth1`.
- Any updates made from this screen are written to the configuration file.

d. Do not make any changes; use the arrow keys to highlight **Cancel** and then press Enter.

e. Highlight **Cancel** on the previous screen and press Enter.

f. Highlight **Quit** on the initial screen and press Enter to exit the utility.

Practice 13-4: Accessing the Public Yum Repository

Overview

In this practice, you configure your system to access the Public Yum Repository from **host03** VM. You then simulate upgrading your system (you *do not* actually perform the upgrade).

Assumptions

- You are the `root` user on **host03** VM.
- You are the `root` user on **dom0**.

Tasks

1. Configure the Domain Name Service (DNS) resolver configuration file.
   - This file, `/etc/resolv.conf`, provides access to DNS.

   a. From **dom0**, use the `cat` command to view the `/etc/resolv.conf` file.

```
[dom0]# cat /etc/resolv.conf
search us.oracle.com example.com
nameserver 192.0.2.1
nameserver 152.68.154.3
nameserver 10.216.106.3
nameserver 193.32.3.252
```

   - This is sample output only; your system may be different.

   b. On **host03**, use the `cat` command to view the `/etc/resolv.conf` file.

```
[host03]# cat /etc/resolv.conf
# Generated by NetworkManager
...
```

   - The first line in `/etc/resolv.conf` states, "Generated by NetworkManager".
   - If NetworkManager is running, which it is, any time you restart the network service, the contents of `/etc/resolv.conf` is updated from information in the `ifcfg-eth*` file.
   - Rather than update `/etc/resolv.conf`, update the `ifcfg-eth*` file to include DNS information.

   c. On **host03**, add the `DOMAIN` directive and add the `DNS1`, `DNS2`, `DNS3`, and `DNS4` entries to the `/etc/sysconfig/network-scripts/ifcfg-eth0` file as shown.
   - These entries were obtained from the `/etc/resolv.conf` file on **dom0**.

```
[host03]# vi /etc/sysconfig/network-scripts/ifcfg-eth0
...
DOMAIN="example.com us.oracle.com"
DNS1=192.0.2.1
DNS2=152.68.154.3
DNS3=10.216.106.3
DNS4=193.32.3.252
```

d. On **host03**, use the `service` command to restart the `network` service.

```
[host03]# service network restart
...
```

e. On **host03**, use the `cat` command to display the contents of the `/etc/resolv.conf` file.

```
[host03]# cat /etc/resolv.conf
# Generated by NetworkManager
search example.com us.oracle.com
nameserver 192.0.2.1
nameserver 152.68.154.3
nameserver 10.216.106.3
...
nameserver 193.32.3.252
```

- Notice that NetworkManager generated the `/etc/resolv.conf` file.

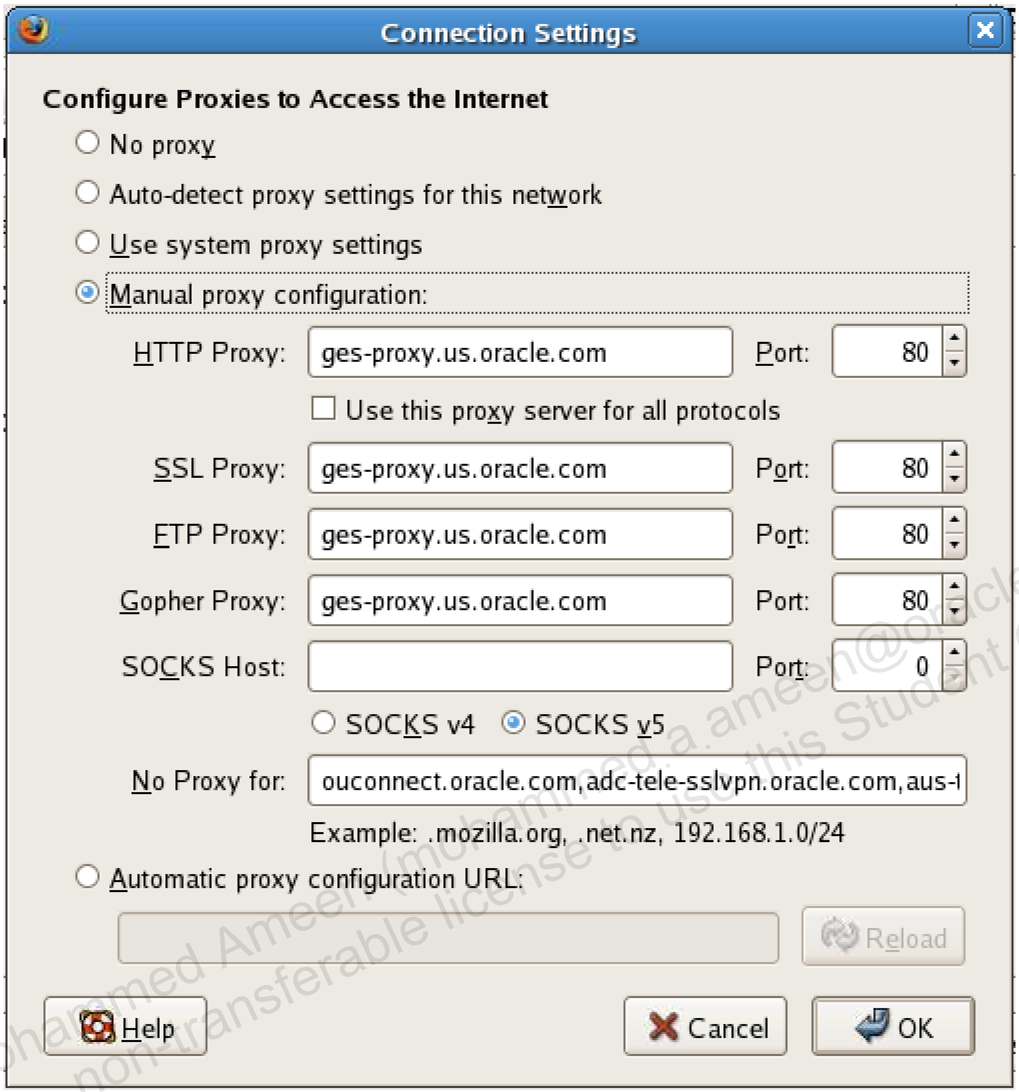
2. Determine the **HTTP Proxy** server on **dom0**.

a. On **dom0**, open Firefox by double-clicking the Firefox icon on the desktop.

b. From the browser menu bar, select **Edit > Preferences** to display the Firefox Preferences window.

c. Click the **Advanced** menu option on the Firefox Preferences window, and then select the **Network** tab.

d. Click the **Settings** button. The Connection Settings window appears.



e. Note the **HTTP Proxy** setting.

- In this example, the setting is `ges-proxy.us.oracle.com` on Port `80`.

f. Click **Cancel** to close the **Connection Settings** window.

g. Click **Close** to close the **Firefox Preferences** window.

h. Click **File > Quit** to close the browser on **dom0**.

All remaining commands are issued from **host03** VM.

3. Set the **HTTP Proxy** in Firefox to the setting configured on **dom0**.

- In this example, the **HTTP Proxy** is `ges-proxy.us.oracle.com`; yours may be different.

a. On the GNOME menu bar, click the Firefox Web Browser icon to the right of the System menu option to start the Firefox web browser.

- The browser appears.

b. From the browser menu bar, select **Edit > Preferences** to display the Firefox Preferences window.

c. Click the **Advanced** menu option and then select the **Network** tab.

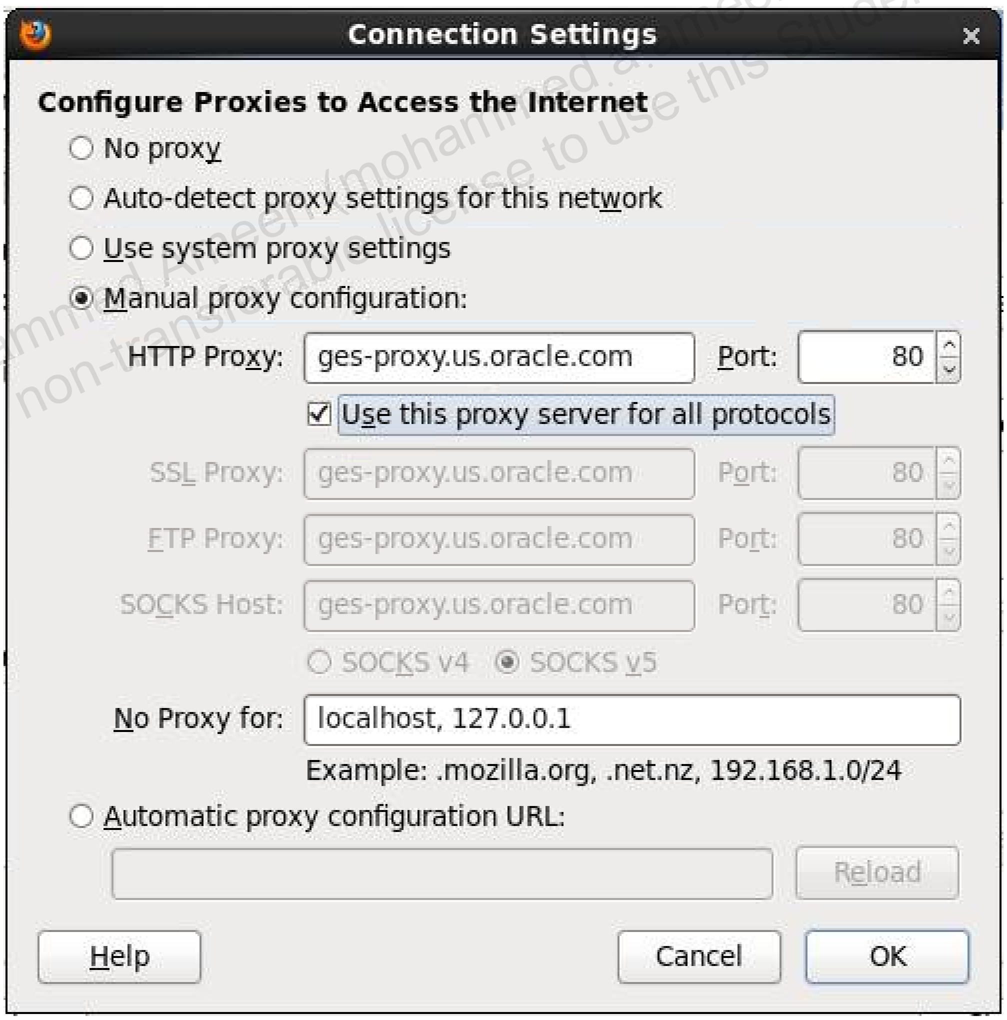
d. Click the **Settings** tab and enter the following settings:

   1) Click **Manual proxy configuration**.

   2) Set **HTTP Proxy** to `ges-proxy.us.oracle.com` on Port `80` (or whatever was set on **dom0**).

   3) Select **Use this proxy server for all protocols**.

      - The Connection Settings window should look like the following (substituting the **HTTP Proxy** settings from **dom0**):



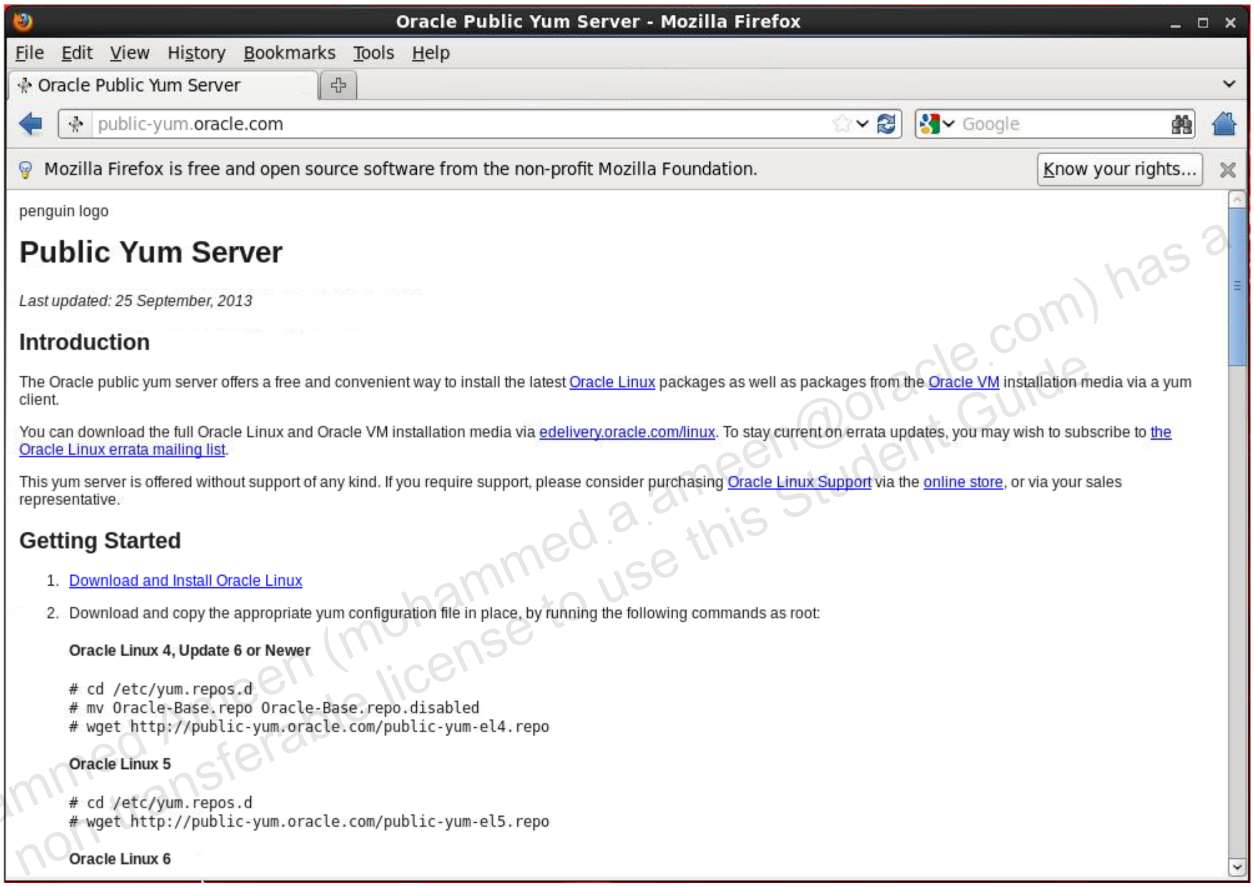
e. Click **OK**.

f. Click **Close** in the Firefox Preferences window.

4. Access the Oracle Public Yum Server URL.

   a. From a browser, enter the URL for the Public Yum Server, http://public-yum.oracle.com.

   - The page looks like the following:



   b. Scroll down to see the instructions for **Oracle Linux 6**, which are:

```
cd /etc/yum.repos.d
wget http://public-yum.oracle.com/public-yum-ol6.repo
```

   - This is information only; you do not need to run these commands.
   - The `public-yum-ol6.repo` file is already present on **host03**.

   c. Close the browser by selecting **File > Quit**.

5. Configure **host03** to access the Public Yum Server repositories.

   - You are configuring your system to access the latest Oracle Linux 6 repository on the Public Yum Server, `[public_ol6_latest]` and the latest Oracle Linux 6 UEK repositories, `[public_ol6_UEK_latest]` and [`public_ol6_UEKR3_latest`].

a. Use the `ls` command to display the `/etc/yum.repos.d` directory.

```
# ls /etc/yum.repos.d
iso.repo       packagekit-mdeia.repo      public-yum-ol6.repo
```

- Notice the `public-yum-ol6.repo` file already exists in this directory.

b. Use the `vi` editor to edit the `public-yum-ol6.repo` file.

- Change `enabled=0` to `enabled=1` for `[public_ol6_latest]`.
- Change `enabled=0` to `enabled=$uek` for `[public_ol6_UEK_latest]`.
- Change `enabled=0` to `enabled=$uekr3` for `[public_ol6_UEKR3_latest]`.

```
# cd /etc/yum.repos.d
# vi public-yum-ol6.repo
[public_ol6_latest]
...
enabled=0                                    (old value)
enabled=1                                    (new value)
...
[public_ol6_UEK_latest]
...
enabled=0                                    (old value)
enabled=$uek                                 (new value)
...
[public_ol6_UEKR3_latest]
...
enabled=0                                    (old value)
enabled=$uekr3                               (new value)
...
```

- Leave all other repositories in this file disabled (`enabled=0`).

c. Run the `yum repolist` command to list the configured repositories.

```
# yum repolist
Loaded plugins:...
... [Errno 14] PYCURL ERROR 6 – "Couldn't resolve host 'public-
yum.oracle.com'"
...
repo id                  Repo name                  status
Myrepo ...
ol6_UEK_latest ...
ol6_latest ...
```

- Notice two things:
  - You received an error because you are unable to resolve http://public-yum.oracle.com.
  - You still have your local repository – `Myrepo` – enabled.

d. To resolve the Public Yum Server, use the `export` command to set the `http_proxy` from the command line.

- Recall the proxy setting was obtained from **dom0**.

```
# export http_proxy=http://ges-proxy.us.oracle.com:80
```

e. To disable the local yum repository, use the `vi` editor to edit the `iso.repo` file and change `enabled=1` to `enabled=0` for `[Myrepo]`.

```
# vi /etc/yum.repos.d/iso.repo
[Myrepo]
...
enabled=0
...
```

f. Run the `yum clean all` command to clean up various things that accumulate in the yum cache directory.

```
# yum clean all
...
Cleaning up Everything
```

g. Run the `yum repolist` command to list the configured repositories.

- This command takes a few minutes to complete.

```
# yum repolist
...
repo id                 Repo name                    status
public_ol6_UEKR3_latest ...
public_ol6_latest ...
...
```

- Notice that you now have `public_ol6_UEKR3_latest` and `public_ol6_latest` included in the output.

6. Run the `yum update` command to display available updates. *Do not apply updates!*

- Answer **n** when asked, "Is this ok".

```
# yum update
...
Transaction Summary
========================================================
Install ...
Upgrade ...
Total download size: ...
Is this ok [y/N]: n
Exiting on user Command
```

- You should see that a number of new packages are available to be installed.
- You should also see a number of packages with updates available.
- *Do not* update at this time. It takes a couple of hours to update.

7. Enable your local yum repository and disable all repositories in the `public-yum-ol6.repo` file.

   a. Use the `vi` editor to edit the `public-yum-ol6.repo` file and change `enabled=0` for ALL repositories.

```
# vi /etc/yum.repos.d/public-yum-ol6.repo
[public_ol6_latest]
...
enabled=0                              (new value)
...
[public_ol6_UEK_latest]
...
enabled=0                              (new value)
...
[public_ol6_UEKR3_latest]
...
enabled=0                              (new value)
...
```

   b. To ensure all repositories are disabled, run the following `grep` command.

```
# grep enabled /etc/yum.repos.d/public-yum-ol6.repo
enabled=0
enabled=0
enabled=0
...
```

   - Ensure all "`enabled`" settings are `0`.

   c. To enable the local yum repository, use the `vi` editor to edit the `iso.repo` file and change `enabled=0` to `enabled=1` for `[Myrepo]`.

```
# vi /etc/yum.repos.d/iso.repo
[Myrepo]
...
enabled=1
...
```

   d. Run the command `yum clean all`, to clean up various things which accumulate in the yum cache directory.

```
# yum clean all
...
Cleaning up Everything
```

e. Run the `yum repolist` command to list the configured repositories.

```
# yum repolist
...
repo id                    Repo name                  status
Myrepo
...
```

- Notice that you now have only the `Myrepo` enabled.

-

Practices for Lesson 14: File Sharing

Chapter 14

Practices for Lesson 14: File Sharing

Practices Overview

In these practices, you:

- Configure an NFS server and client and mount an exported file system
- Use automounter to mount the virtual CD drive
- Configure and use an FTP server

Practice 14-1: Configuring an NFS Server and an NFS Client

Overview

In this practice, you:

- Export a file system from **host03** VM and mount it on **host01** VM
- Ensure that the required package is installed and that services are running
- Use various NFS-related commands and files to share file systems using NFS

Assumptions

- This practice is performed on **host01** and on **host03** VMs.
- Open a terminal window on each system.
- Log in as the `root` user on each system.
- The prompts in the solution section include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. Use the `rpm` command to verify that the `nfs-utils` package is installed on **host03**.

```
[host03]# rpm –qa | grep nfs-utils
nfs-utils-lib-...
nfs-utils-...
```

   - In this example, the package is already installed along with the `nfs-utils-lib` package.

2. Use the `vi` editor and edit `/etc/exports` on **host03** to add an entry to export `/Dev` to all client systems.

```
[host03]# vi /etc/exports
/Dev *
```

3. Use the `service` command to verify that the required services are started on **host03**.

   a. Verify that the `rpcbind` service is started.

```
[host03]# service rpcbind status
rpcbind (pid ...) is running...
```

   - In this example, the `rpcbind` service is running.

   b. Verify that the `nfs` service is started.

```
[host03]# service nfs status
rpc.svcgssd is stopped
rpc.mountd is stopped
nfsd is stopped
rpc.rquotad is stopped
```

   - In this example, the services are not running and must be started.

c. Start the `nfs` service and associated services.

```
[host03]# service nfs start
Starting NFS services: exportfs: No options for /Dev *:...
                                           [  OK  ]
Starting NFS quotas:                       [  OK  ]
Starting NFS mountd:                       [  OK  ]
Starting NFS daemon:                       [  OK  ]
Starting RPC idmapd:                       [  OK  ]
```

d. Verify that the `nfslock` service is started.

```
[host03]# service nfslock status
rpc.statd (pid ...) is running...
```

- In this example, the service is running.

4. Use the `chkconfig` command to configure the services to start at boot time.

```
[host03]# chkconfig rpcbind on
[host03]# chkconfig nfs on
[host03]# chkconfig nfslock on
```

5. Use the `showmount` command on **host03** to display exported file systems.

```
[host03]# showmount -e
Export list for host03.example.com:
/Dev *
```

- If the exported file system is not listed, restart the `nfs` service.
- Whenever a new entry is made to `/etc/exports`, restart the `nfs` service.

6. Use the `exportfs` command on **host03** to export `/Test` to all clients and allow read-write permission. Include the option to ignore `/etc/exports` entries.

```
[host03]# exportfs -i -o rw *:/Test
```

7. Use the `showmount` command on **host03** to display exported file systems.

```
[host03]# showmount -e
Export list for host03.example.com:
/Dev *
/Test *
```

- Notice that both exported file systems are listed.
- You do not need to restart the `nfs` service when using `exportfs`.

8. Use the `cat` command to view the contents of `/var/lib/nfs/etab` on **host03**.

```
[host03]# cat /var/lib/nfs/etab
/Dev  *(ro,sync,wdelay,hide,nocrossmnt,secure,root_squash,...
/Test *(rw,sync,wdelay,hide,nocrossmnt,secure,root_squash,...
```

- Notice that both exported file systems are listed in this master export table.

- The rpc.mountd process reads this file when a client attempts to mount an NFS file system.

9. Use the vi editor to edit /etc/exports on **host03** and change the entry exporting /Dev to the following:

```
[host03]# vi /etc/exports
/Dev *(rw,no_root_squash)
```

- The rw option allows client systems to make changes to the file system.
- The no_root_squash option allows root users on client systems to retain root privileges on the file system.

10. Run the exportfs -r command on **host03**.

```
[host03]# exportfs -r
```

- This command re-exports the entries in /etc/exports and synchronizes /var/lib/nfs/etab with /etc/exports.

11. Use the cat command to view the contents of /var/lib/nfs/etab on **host03**.

```
[host03]# cat /var/lib/nfs/etab
/Dev  *(rw,sync,wdelay,hide,nocrossmnt,secure,no_root_squash,
...
```

- Notice the new options on the /Dev NFS share.
- Also notice that /Test is no longer listed.

12. Use the showmount command on **host03** to display exported file systems.

```
[host03]# showmount -e
Export list for host03.example.com:
/Dev *
```

- This confirms that the /Test file system is no longer exported.

13. Mount the exported /Dev file system on **host01**.
    a. Use the mkdir command to create a mountpoint named /remote_dev on **host01**.

    ```
    [host01]# mkdir /remote_dev
    ```

    b. Use the mount command to mount the exported file system from **host03**, /Dev, with rw and nosuid options on the local mountpoint, /remote_dev.
      - The rw option mounts the file system with read-write permissions.
      - The nosuid option does not allow setuid or setgid bits to take effect.

    ```
    [host01]# mount -t nfs -o rw,nosuid host03:/Dev /remote_dev
    mount.nfs: mount system call failed
    ```

      - In this example, the mount command failed.
      - It takes several seconds for the mount command to fail. Rather than wait for the command to fail, you can press Ctrl + C to abort the mount command.

c. If the mount command failed, disable the firewall on **host03** by running the following command.

```
[host03]# service iptables stop
iptables: Flushing firewall rules:                      [  OK  ]
iptables: Setting chains to policy ACCEPT: filter       [  OK  ]
iptables: Unloading modules:                            [  OK  ]
```

- Firewalls, the iptables command, and other system security–related issues are covered in the Lesson 17 "Security Administration."
- You can allow NFS connectivity without disabling the entire firewall, which is covered in Lesson 17.
- For the purposes of this practice, stop the iptables service on **host03** to allow **host01** to mount the exported file system from **host03**.

d. Re-issue the mount command from **host01**, if it is necessary to mount the NFS share.

```
[host01]# mount -t nfs -o rw,nosuid host03:/Dev /remote_dev
```

- In this example, the mount command was successful.

e. Run the mount command to view the mount information for the NFS share.

```
[host01]# mount | grep nfs
...
host03:/Dev on /remote_dev type nfs
(rw,nosuid,vers=4,addr=192.0.2.103,clientaddr=192.0.2.101)
```

- Notice the mount options (rw,nosuid), the NFS version (4), and the server and client IP addresses.

14. Verify that NFS file system is mounted with rw (read-write) permissions.

a. On **host01**, use the df command to display the mounted file systems.

```
[host01]# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
/dev/xvda2     5.8G  1.9G   3.7G   34%  /
...
host03:/Dev  1008M   34M   924M    4%  /remote_dev
```

- Notice that the host03:/Dev file system is mounted on the local file system /remote_dev.

b. Use the ls command to list the contents of /remote_dev on **host01**.

```
[host01]# ls /remote_dev
lost+found
```

c. Use the ls command to list the contents of /Dev on **host03**.

```
[host03]# ls /Dev
lost+found
```

- Notice the contents of /Dev on **host03** are the same as /remote_dev on **host01** because they are the same directories.

d. From **host03**, use the `vi` command to create a file on `/Dev`. Enter some content in the file.

```
[host03]# vi /Dev/test
<enter some content>
```

e. From **host01**, use the `vi` command to edit the file created from **host03**. Make some changes to the content.

```
[host01]# vi /remote_dev/test
<change the content>
```

f. From **host03**, use the `cat` command to view the contents of the file in `/Dev`.

```
[host03]# cat /Dev/test
<content has changed>
```

- This again confirms `/Dev` on **host03** is the same as `/remote_dev` on **host01** and that the file system has read-write permissions.

Practice 14-2: Using Automounter

Overview

In this practice, you:

- Use automounter to mount the virtual CD drive
- Use the `-hosts` map to automount all exports from **host03**

Assumptions

- This practice is performed on **host01** VM, but you are asked to run a single command from **dom0**.
- Open a terminal window on each system.
- Log in as the `root` user on each system.
- The prompts in the solution section include either **dom0** or **host01** to indicate which system to enter the command from.

Tasks

1. From **dom0**, use the `grep` command to search for "cd" in the `vm.cfg` file for **host01**.

```
[dom0]# grep cd /OVS/running_pool/host01/vm.cfg
'file:/OVS/seed_pool/OracleLinux-R6-U5-Server-x86_64-
dvd.iso,hdc:cdrom,r']
```

- Notice that the `dvd.iso` image of Oracle Linux 6.5 is on the virtual `cdrom` drive.

All remaining commands are run from the **host01** VM.

2. Use the `rpm` command to verify that the `autofs` package is installed.

```
# rpm -q autofs
autofs-5.0.5-88.0.1.el6.x86_64
```

- In this example, the package is already installed.

3. Use the `service` command to verify that the `autofs` service is running.

```
# service autofs status
automount (pid ...) is running...
```

- In this example, the service is already running.

4. Use the `grep` command to search for "misc" in the `/etc/auto.master` master map-file.

```
# grep misc /etc/auto.master
/misc     /etc/auto.misc
...
```

- Notice that the `/misc` mount point is associated with the `/etc/auto.misc` map-file.

5. Use the `grep` command to search for "cd" in the `/etc/auto.misc` file.

```
# grep cd /etc/auto.misc
cd        -fstype=iso9600,ro,nosuid,nodev       :/dev/cdrom
```

- Notice that the `cd` mount point is associated with the `/dev/cdrom` device.

6. Use the `df` command to display the mounted file systems.

```
# df -h
Filesystem    Size  Used  Avail  Use%  Mounted on
/dev/xvda2    5.7G  2.4G   3.1G   44%  /
...
```

- Notice that the `cdrom` is not mounted.

7. Use the `ls /misc/cd` command, combining the `/misc` mountpoint from `/etc/auto.master` with the `cd` mountpoint from `/etc/auto.misc`.

```
# ls /misc/cd
ls
EFI           Packages           ResilientStorage
EULA          README-en          RPM-GPG-KEY
eula.en_US    README-en.html     RPM-GPG-KEY-oracle
...
```

- Notice that this is the contents of the Oracle Linux 6.5 `dvd-iso` image.

8. Use the `df` command to display the mounted file systems.

```
# df -h
Filesystem    Size  Used  Avail  Use%  Mounted on
/dev/xvda2    5.7G  2.4G   3.1G   44%  /
...
/dev/sr0      3.7G  3.7G      0  100%  /misc/cd
```

- Notice that the `cdrom` was "automounted" simply by accessing it with the `ls` command.

9. Use the `grep` command to search for "net" in the `/etc/auto.master` master map file.

```
# grep net /etc/auto.master
/net     -hosts
...
```

- Notice that the `/net` mount point is associated with the `-hosts` map.

10. Use the `cd` command to change to the `/net/host03` directory to automount all exports from **host03**. Use the `ls` command to list the contents of the directory.

```
# cd /net/host03
# ls
Dev
```

- Notice that the `/net/host03` directory contains the exported file system from **host03**, `/Dev`.

11. Use the `ls` command to list the contents of `/net/host03/Dev` on **host01**.

```
# ls /net/host03/Dev
lost+found     test
```

Practice 14-3: Configuring an FTP Server

Overview

In this practice, you:

- Install the `vsftpd` server package on **host03** and start the service
- Install the `ftp` (client) package and test the setup

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Use the `yum` command to install the `vsftpd` package on **host03**.

```
# yum install vsftpd
...
Transaction Summary
============================================================
Install       1 Package(s)
Total download size: 150 k
Installed size: 331 k
Is this ok [y/N]: y
...
Complete!
```

2. Use the `service` command to start the `vsftpd` service.

```
# service vsftpd start
Starting vsftpd for vsftpd:                [  OK  ]
```

3. Use the `chkconfig` command to configure `vsftpd` to start at boot time.

```
# chkconfig vsftpd on
```

4. Use the `yum` command to install the `ftp` (client) package on **host03**.

```
# yum install ftp
...
Transaction Summary
============================================================
Install       1 Package(s)
Total download size: 57 k
Installed size: 95 k
Is this ok [y/N]: y
...
Complete!
```

5. Test the setup.
   a. Use the `ftp` command to connect to `localhost` and log in as `anonymous`.
      - Use any password.
   b. After connecting, run the `ls` command to display the contents of `/var/ftp`.
   c. Conclude the test by running the `quit` command to exit.

```
# ftp localhost
Connected to localhost (127.0.0.1).
220 (vsFTPd 2.2.2)
Name (localhost:root): anonymous
331 Please specify the password.
Password: your_email_address
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls
227 Entering Passive Mode (127,0,0,1,54,166).
150 Here comes the directory listing.
drwxr-xr-x  2 0    0   4096  Apr 29  2012 pub
226 Directory send OK.
ftp> quit
221 Goodbye.
```

   - Notice that the contents of `/var/ftp` is a `pub` directory for anonymous users.

6. Use the `grep` command to display the `ftp` user information in `/etc/passwd`.

```
# grep ftp /etc/passwd
ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin
```

   - In this example, the home directory of the `ftp` user is `/var/ftp`.

7. Ensure that the permissions on the home directory are set to 755. Change the settings if necessary.

```
# ls -ld /var/ftp
drwxr-xr-x 3 root root 4096 Mar 10 11:01 /var/ftp
```

   - In this example, the permissions are correct.

8. Copy the `/root/install.log` file to `/var/ftp/pub` and rename it as `test_file`.

```
# cp /root/install.log /var/ftp/pub/test_file
```

- This file is used in the next practice.

Practice 14-4: Downloading a File from an FTP Server

Overview

In this practice, you install the `ftp` package on **host01** VM and download a file from the FTP server on **host03**.

Assumptions

You are the `root` user on **host01** VM.

Tasks

1. From **host01**, use the `cd /misc/cd` command to automount the virtual `cdrom`, which contains Oracle Linux `dvd.iso` image.

```
# cd /misc/cd
```

2. Use the `cd` command to change to the `Packages` directory.

```
# cd Packages
```

3. Use the `ls` command to display the `ftp` package name.

```
# ls ftp*
ftp-0.17-54.el6.x86_64.rpm
```

- This is an example only; your version may be different.

4. Use the `rpm` command to install the `ftp` (client) package name displayed in step 3.

```
# rpm –Uvh ftp-0.17-54.el6.x86_64.rpm
...
```

- The `ftp` package may already be installed on your system.

5. Use the `cd` command to change back to your home directory.

```
# cd
# pwd
/root
```

6. From **host01**, use the `ftp` utility to connect to the FTP server, **host03**, as `anonymous` user.

```
[host01]# ftp host03
Connected to localhost (192.0.2.103).
220 (vsFTPd 2.2.2)
Name (host03:root): anonymous
331 Please specify the password.
Password: your_email_address
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

7. Enter `help` or `?` to display a list of available commands.

```
ftp> help
...
```

8. Get `help` on some of the available commands, for example:

```
ftp> help get
get             receive file
ftp> help put
put             send one file
ftp> help mget
mget            get multiple files
ftp> help cd
cd              change remote working directory
ftp> help ls
ls              list contents of remote directory
ftp> help !
!               escape to the shell
```

9. Use the `cd` command to change to the `pub` directory.

```
ftp> cd pub
Directory successfully changed.
```

10. Use the `ls` command to list the contents of the `/var/ftp/pub` directory on the FTP server.

```
ftp> ls
227 Entering Passive Mode (192,0,2,103,26,76).
150 Here comes the directory list.
-rw-r--r--      1  0       0       50204 May 22 22:51 test_file
226 Directory send OK.
```

- Notice that the `test_file` exists in the `pub` directory on the FTP server.

11. Use the `!ls` command to list the contents of the local directory on **host01**.

```
ftp> !ls
anaconda-ks.cfg   install.log   install.log.syslog
...
```

12. Use the `get` command to download the `test_file` file from the FTP server.

```
ftp> get test_file
local: test_file remote: test_file
227 Entering Passive Mode (192,0,2,103,75,78).
150 Opening BINARY mode data connection for test_file ...
226 Transfer complete.
51355 bytes received in ...
```

13. Use the `!ls` command to list the contents of the local directory on **host01**.

```
ftp> !ls
anaconda-ks.cfg   install.log   install.log.syslog   test_file
...
```

- Notice that the `test_file` exists in the local directory on **host01**.

14. Use the `quit` command to exit `ftp`.

```
ftp> quit
221 Goodbye.
```

Practice 14-5: Configuring `xinetd`

Overview

In this practice, you configure `vsftpd` to be controlled by `xinetd`. Stated another way, you configure the `xinetd` "superserver" to start `vsftpd` on demand.

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. On **host03**, configure `vsftpd` to run in normal mode.
   - The `vsftpd` server can run in normal mode or it can run in stand-alone mode.
     - In stand-alone mode, `vsftpd` runs as a daemon and handles connections directly.
     - In normal mode, the `xinetd` daemon calls `vsftpd` each time a client tries to make a connection.

   a. Use the `grep` command to display the "`listen`" directive in the `/etc/vsftpd/vsftpd.conf` file.

   ```
   # grep listen /etc/vsftpd/vsftpd.conf
   # When "listen" directive is enabled, vsftpd runs in standalone
   mode ...
   listen=YES
   ...
   ```

   - Notice that "`listen`" is enabled (set to `YES`), which causes `vsftpd` to run in stand-alone mode.

   b. Use the `vi` editor to edit `/etc/vsftpd/vsftpd.conf` and set the "`listen`" directive to **`NO`**.

   ```
   # vi /etc/vsftpd/vsftpd.conf
   listen=NO
   ```

2. Install the `xinetd` package (if necessary).

   a. Use the `rpm -qa` command to determine if the `xinetd` package is installed.

   ```
   # rpm -qa | grep xinetd
   ```

   - No output indicates the package is not installed.

b. Use the `yum` command to install the `xinetd` package on **host03**.

```
# yum install xinetd
...
Transaction Summary
============================================================
Install       1 Package(s)
Total download size: 121 k
Installed size: 259 k
Is this ok [y/N]: y
...
Complete!
```

3. Install and configure a `xinetd` control file for `vsftpd`.
   - You must install a `xinetd` control file to run `vsftpd` in normal mode.

a. Use the `cp` command to copy the sample control file (in `/usr/share/doc/vsftpd*/vsftpd.xinetd`) to the `/etc/xinetd.d/` directory and rename the sample control file `vsftpd`.

```
# cp /usr/share/doc/vsftpd*/vsftpd.xinetd /etc/xinetd.d/vsftpd
```

b. Use the `vi` editor to edit `/etc/xinetd.d/vsftpd` and set the "`disable`" directive to **no**.

```
# vi /etc/xinetd.d/vsftpd
disable          = no
```

4. Stop the `vsftpd` service and start the `xinetd` service.

a. Use the `service` command to stop the `vsftpd` service.

```
# service vsftpd stop
Shutting down vsftpd:                              [  OK  ]
```

b. Use the `service` command to start the `xinetd` service.

```
# service xinetd start
Starting xinetd:                                   [  OK  ]
```

c. Use the `chkconfig` command to display the `xinetd`-based services.

```
# chkconfig --list
NetworkManager  0:off  1:off  2:on   3:on  4:on   5:on  6:off
abrt-ccpp       0:off  1:off  2:off  3:on  4:off  5:on  6:off
...
xinetd based services:
        chargen-dgram:  off
        chargen-stream: off
...
        time-stream:    off
        vsftpd:         on
```

- Notice that `vsftpd` is the only enabled `xinetd`-based service.
- Also notice that each `xinetd`-based service has a configuration file in the `/etc/xinetd.d` directory

d. Use the `ls` command to list the contents of the `/etc/xinetd.d` directory

```
# ls -l /etc/xinetd.d
-rw------- ... chargen-dgram
-rw------- ... chargen-stream
...
-rw-r--r-- ... vsftpd
```

5. Test the `xinetd`-based service (`vsftpd`).

a. From **host01**, use the `ftp` utility to connect to the FTP server, **host03**, as the `anonymous` user.

```
[host01]# ftp host03
Connected to localhost (192.0.2.103).
220 (vsFTPd 2.2.2)
Name (host03:root): anonymous
331 Please specify the password.
Password: your_email_address
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

b. Use the `cd` command to change to the `pub` directory.

```
ftp> cd pub
Directory successfully changed.
```

c. Use the `ls` command to list the contents of the `/var/ftp/pub` directory on the FTP server.

```
ftp> ls
227 Entering Passive Mode (192,0,2,103,26,76).
150 Here comes the directory list.
-rw-r--r--      1  0       0      50204 May 22 22:51 test_file
226 Directory send OK.
```

- Notice that the `test_file` exists in the `pub` directory on the FTP server.

d. Use the `!ls` command to list the contents of the local directory on **host01**.

```
ftp> !ls
anaconda-ks.cfg   install.log   install.log.syslog   test_file
...
```

- Notice that the `test_file` exists in the local directory on **host01**.

e. Use the `get` command to download the `test_file` file from the FTP server. Because one copy of `test_file` already exists, include a second argument—`tf2`—to rename the file.

```
ftp> get test_file tf2
local: tf2 remote: test_file
227 Entering Passive Mode (192,0,2,103,149,8).
150 Opening BINARY mode data connection for test_file ...
226 Transfer complete.
50080 bytes received in ...
```

f. Use the `!ls` command to list the contents of the local directory on **host01**.

```
ftp> !ls
anaconda-ks.cfg  install.log  install.log.syslog  test_file  tf2
...
```

- Notice that the `tf2` file exists in the local directory on **host01**.

g. Use the `quit` command to exit `ftp`.

```
ftp> quit
221 Goodbye.
```

Practices for Lesson 15: OpenSSH

Chapter 15

Practices for Lesson 15: OpenSSH

Practices Overview

In these practices, you do the following:

- You verify that the OpenSSH packages are installed and that the `sshd` service is running.
- You use the `ssh` and `scp` utilities.
- You use the `ssh-keygen` utility to generate keys enabling connectivity without supplying a password.

Practice 15-1: Connecting to a Remote Server by Using `ssh`

Overview

In this practice, you verify that the OpenSSH packages are installed, verify that the `sshd` service is started on the server, and use the `ssh` utility to establish a connection and execute a command on a remote system.

Assumptions

- This practice is performed on **host01** and **host03** VMs.
- Open a terminal window on each system.
- Log in as the `root` user on each system.
- The prompts in the solution section include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. On **host03**, use the `rpm` command to verify that the `openssh` packages are installed.

```
[host03]# rpm -qa | grep openssh
openssh-clients-...
openssh-server-...
openssh-...
```

   - In this example, the packages are already installed.

2. Use the `service` command to verify that the `sshd` service is started on **host03**.

```
[host03]# service sshd status
openssh-daemon (pid ...) is running...
```

   - In this example, the service is running.

3. Use the `chkconfig` command to verify that the `sshd` service is configured to start at boot time on **host03**.

```
[host03]# chkconfig sshd --list
sshd  0:off  1:off  2:on  3:on  4:on  5:on  6:off
```

   - In this example, the service is configured to start at boot time.

4. On **host01**, use the `rpm` command to verify that the `openssh` packages are installed.

```
[host01]# rpm -qa | grep openssh
openssh-clients-...
openssh-server-...
openssh-...
```

   - In this example, the packages are already installed.

5. On **host01**, use the `useradd` and `passwd` commands to create the `oracle` user. Assign a password of `oracle`. Ignore the "BAD PASSWORD" warning.

```
[host01]# useradd oracle
[host01]# passwd oracle
```

```
Changing password for user oracle.
New password: oracle
...
Retype new password: oracle
passwd: all authentication tokens updated successfully.
```

6. Use the `exit` command to log off as `root` on **host01**. Log back on as user `oracle`.

```
[host01]# exit
host01 login: oracle
Password: oracle
```

7. Use the `ls` command to display a long listing of all files in the home directory of user `oracle`.

```
[oracle@host01 ~]$ ls -la
...
```

   - Notice that there is no `~/.ssh` directory.

8. Perform a remote login to **host03** by using the `ssh` command.
   - Answer "`yes`" when asked, "Are you sure."
   - The `oracle` user password on **host03** is `oracle`.

```
[oracle@host01 ~]$ ssh host03
The authenticity of host 'host03 (192.0.2.103)' can't be
established. RSA key fingerprint is ...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'host03,192.0,2,103' (RSA) to the
list of known hosts.
oracle@host03's password: oracle
[oracle@host03 ~]$
```

9. Use the `hostname` command to display the host name to confirm that you successfully logged on to **host03**.

```
[oracle@host03 ~]$ hostname
host03.example.com
```

10. Use the `logout` command to close the `ssh` connection to **host03**. Use the `hostname` command to confirm that you are back to **host01**.

```
[oracle@host03 ~]$ logout
Connection to host03 closed.
[oracle@host01 ~]$ hostname
host01.example.com
```

11. Use the `ls` command to display a long listing of all files in the home directory of user `oracle`.

```
[oracle@host01 ~]$ ls -la
...
drwx------ 2 oracle oracle 4096 Mar  8 08:23 .ssh
```

- Notice that there is now a `~/.ssh` directory.

12. Use the `cd` command to change to the `~/.ssh` directory, and then use `ls` to view the contents of the directory.

```
[oracle@host01 ~]$ cd .ssh
[oracle@host01 .ssh]$ ls
known_hosts
```

- Notice that the `known_hosts` file was created.

13. Use the `cat` command to view the contents of the `known_hosts` file.

```
[oracle@host01 .ssh]$ cat known_hosts
host03,192.0.2.103 ssh-rsa ...
```

- Notice that **host03** is now a "known host".

14. Perform a remote login to **host03** using the `ssh` command.

```
[oracle@host01 .ssh]$ ssh host03
oracle@host03's password: oracle
Last login...
[oracle@host03 ~]$
```

- Notice that you are not asked to confirm this time, because of the existence of the `known_hosts` file.

15. Use the `logout` command to close the `ssh` connection to **host03**. Use the `hostname` command to confirm that you are back to **host01**. Use the `cd` command to change back to the user's home directory.

```
[oracle@host03 ~]$ logout
Connection to host03 closed.
[oracle@host01 .ssh]$ hostname
host01.example.com
[oracle@host01 .ssh]$ cd
[oracle@host01 ~]$
```

16. Log on to **host03** as user `root` and run the `ls` command with a single `ssh` command.

- The `root` user password on **host03** is `oracle`.

```
[oracle@host01 ~]$ ssh root@host03 ls
root@host03's password: oracle
...
[oracle@host01 ~]$
```

- Notice that the `ls` command ran on the remote system displaying the contents of the remote directory, and then the remote connection closed.

17. Use the `hostname` command to confirm that you are back to **host01**.

```
[oracle@host01 ~]$ hostname
host01.example.com
```

Practice 15-2: Configuring OpenSSH to Connect Without a Password

Overview

In this practice, you use the `ssh-keygen` command to generate an RSA key pair and configure OpenSSH to connect to a remote system without supplying a password. You also use the `scp` command in this practice.

Assumptions

- This practice is performed on **host01** and on **host03** VMs.
- Open a terminal window on each system.
- Log in as the `root` user on **host03**.
- Log in as the `oracle` user on **host01**.
- The prompts in the solution section include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. From **host03**, use the `su` command to become the `oracle` user.

```
[root@host03 ~]# su - oracle
[oracle@host03 ~]$ whoami
oracle
[oracle@host03 ~]$ pwd
/home/oracle
```

2. Use the `ls` command to view the contents of the `~/.ssh` directory.

```
[oracle@host03 ~]$ ls ~/.ssh
```

- Notice that the directory is empty.

3. Use the `ssh-keygen` command to create the public and private parts of an RSA key. Accept all the defaults.

```
[oracle@host03 ~]$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/oracle/.ssh/id_rsa):
ENTER
Enter passphrase (empty for no passphrase): ENTER
Enter same passphrase again: ENTER
Your identification has been saved in /home/oracle/.ssh/id_rsa.
Your public key has been saved in /home/oracle/.ssh/id_rsa.pub.
The key fingerprint is:...
The key's randomart image is:...
```

4. Use the `ls` command to view the contents of the `~/.ssh` directory.

```
[oracle@host03 ~]$ ls ~/.ssh
id_rsa     id_rsa.pub
```

- Notice that the `ssh-keygen` command generated two keys.

5. Use the `scp` command to copy `~/.ssh/id_rsa.pub` on the local system (**host03**) to `~/.ssh/authorized_keys` on the remote system (**host01**). Password is `oracle`.

```
[oracle@host03 ~]$ scp .ssh/id_rsa.pub
host01:~/.ssh/authorized_keys
The authenticity of host 'host01 (192.0.2.101)' can't be
established. RSA key fingerprint is ...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'host01,192.0,2,101' (RSA) to the
list of known hosts.
oracle@host01's password: oracle
id_rsa.pub                      100%  407   0.4KB/s   00:00
[oracle@host03 ~]$
```

- Because you are connecting to this OpenSSH server for the first time, you are asked to confirm the connection.
- Notice that a password is required to make the connection.
- Notice that the file is copied but you are still connected to the local system (**host03**).

6. Perform a remote login to **host01** using the `ssh` command.

```
[oracle@host03 ~]$ ssh host01
Last login:...
```

- Notice that you no longer need to enter a password.

7. Use the `hostname` command to confirm that you successfully logged on to **host01**.

```
[oracle@host01 ~]$ hostname
host01.example.com
```

8. Use the `ls` command to view the contents of the `~/.ssh` directory.

```
[oracle@host01 ~]$ ls ~/.ssh
authorized_keys   known_hosts
```

- Notice the existence of the `authorized_keys` file, which allowed you to connect without supplying a password.

9. Use the `logout` command to close the connection to **host01**. Use the `hostname` command to confirm that you are back to **host03**.

```
[oracle@host01 ~]$ logout
Connection to host01 closed.
[oracle@host03 ~]$ hostname
host03.example.com
```

10. Use the `exit` command to log out as `oracle` user and return to the `root` logon. Use the `whoami` command to confirm that you are logged on as `root`.

```
[oracle@host03 ~]$ exit
logout
[root@host03 ~]$ whoami
root
```

Practices for Lesson 16: Pluggable Authentication Modules (PAM)

Chapter 16

Practices for Lesson 16: Pluggable Authentication Modules (PAM)

Practices Overview

In these practices, you configure PAM authentication modules first to allow a single login only, and then to disable all non-`root` logins.

Practice 16-1: Configuring PAM for a Single Login Session

Overview

In this practice, you configure a PAM authentication module on **host03** to allow only a single login session for a user.

Assumptions

- This practice is performed on **host01** and **host03** VMs.
- You open a terminal window on each system.
- You log in as the `root` user on **host03**.
- The prompts in the solution section include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. On **host03**, view PAM configuration files and directories.
   a. Use the `ls` command to view the PAM configuration directory, `/etc/pam.d`.

```
[host03]# ls /etc/pam.d
atd           login      sshd
authconfig    newrole    ssh-keycat
...
```

   - This directory contains files that describe the authentication procedure for an application.

   b. Use the `cat` command to view the `sshd` configuration file in `/etc/pam.d`.

```
[host03]# cat /etc/pam.d/sshd
#%PAM-1.0
auth       required    pam_sepermit.so
auth       include     password-auth
...
```

   - This file contains a group of directives that define the authentication modules as well as any controls or arguments.
   - The authentication modules are listed in the third column.

   c. Use the `find` command to locate the `pam_sepermit.so` authentication module.

```
[host03]# find / -name pam_sepermit.so
/lib64/security/pam_sepermit.so
```

   - In this example, the authentication module is located in `/lib64/security`.

   d. Use the `ls` command to view the authentication modules directory.

```
[host03]# ls /lib64/security
pam_access.so    pam_krb5afs.so    pam_sepermit.so
pam_cap.so       pam_krb5.so       pam_shells.so
...
```

- Notice that all authentication modules are located in this directory.

2. On **host03**, view the man pages for the `pam_sepermit` authentication module and the associated configuration file.

   a. Most of the authentication modules have a man page describing their purpose and usage. Use the `man pam_sepermit` command to view the man page for the `pam_sepermit` authentication module.

```
[host03]# man pam_sepermit
...
    pam_sepermit - PAM module to allow/deny login depending
    On SELinux enforcement state
...
    When the user which is logging in matches an entry in the
    config file he is allowed access only when the SELinux
    is in enforcing mode. Otherwise he is denied access...


...
    See sepermit.conf(5) for details.
...
```

   - Notice that this module uses a config file, `sepermit.conf`, which controls access when SELinux is in enforcing mode.
   - SELinux stands for "Security-Enhanced Linux" and is covered in a later course.

   b. Use the `man sepermit.conf` command to view the man page for the `sepermit.conf` file.

```
[host03]# man sepermit.conf
...
    sepermit.conf - configuration file for the pam_sepermit
module
...
    The lines of the configuration file have the following
syntax:
...
```

3. SELinux is covered in a later course but for the purposes of this practice, use the `sestatus` command to display information about SELinux.

```
[host03]# sestatus
SELinux status:    enabled
...
Current mode:      enforcing
...
```

- Output shown is a sample showing that SELinux is enabled and is in enforcing mode.
- With SELinux in enforcing mode, the `pam_sepermit` authentication module allows or denies login.

4. From **host01**, confirm you can remotely log in to host03.

   a. From the login prompt, log in as `oracle` user. Password is `oracle`.

```
host01 login: oracle
Password: oracle
Last login...
[oracle@host01 ~]$
```

   b. Use the `ssh` command to connect to **host03**. Password is `oracle`.

```
[oracle@host01 ~]$ ssh host03
oracle@host03's password: oracle
Last login:...
[oracle@host03 ~]$
```

   c. Use the `hostname` command to confirm that you successfully logged in to **host03**.

```
[oracle@host03 ~]$ hostname
host03.example.com
```

   - Notice that you are successfully able to log in to **host03**.

   d. Use the `logout` command to close the connection to **host03**.

```
[oracle@host03 ~]$ logout
Connect to host03 closed.
[oracle@host01 ~]$ hostname
host01.example.com
```

   - Notice that you are now logged off of **host03** and back to **host01**.

5. On **host03**, configure the `pam_sepermit` authentication module to deny login.

   a. Use the `find` command to locate the `sepermit.conf` file.

```
[host03]# find / -name sepermit.conf
/etc/security/sepermit.conf
```

   - Notice that the `sepermit.conf` file is located in the `/etc/security` directory.

   b. Use the `vi` editor to add the following entry to `/etc/security/sepermit.conf`.

```
[host03]# vi /etc/security/sepermit.conf
oracle:exclusive
```

   - This entry, when read by the PAM module `pam_sepermit.so`, allows only a single login session for the `oracle` user.

6. From **host01**, attempt to log in to **host03**.

   Use the `ssh` command to connect to **host03**. Password is `oracle`.

```
[oracle@host01 ~]$ ssh host03
oracle@host03's password: oracle
Permission denied, please try again.
oracle@host03's password: CTRL-C
[oracle@host01 ~]$
```

- Notice that the connection is denied by the PAM authentication module.

To permit the `oracle` user login from **host01**, you can either:

- Remove the entry in the `/etc/pam.d/sshd` file to use the `pam_sepermit.so` module.
- Remove the entry in the `/etc/security/sepermit.conf` file to allow only a single login session.

7. From **host03**, permit user `oracle` to log in from **host01** by using the `vi` editor to comment out the entry to use the `pam_sepermit.so` module from the `/etc/pam.d/sshd` file.
   - Comment out this line by inserting a `#` sign at the beginning of the line as follows:

```
[host03]# vi /etc/pam.d/sshd
auth   required   pam_sepermit.so          (current entry)
#auth   required   pam_sepermit.so          (insert # sign)
```

8. From **host01**, attempt to log in to **host03**.
   a. Use the `ssh` command to connect to **host03**. Password is `oracle`.

```
[oracle@host01 ~]$ ssh host03
oracle@host03's password: oracle
Last login…
[oracle@host03 ~]$ hostname
host03.example.com
```

   - Notice that the connection is allowed, and no longer denied by the PAM authentication module.

   b. Use the `logout` command to close the connection to **host03**.

```
[oracle@host03 ~]$ logout
Connect to host03 closed.
[oracle@host01 ~]$ hostname
host01.example.com
```

   - Notice that you are now logged off of **host03** and back to **host01**.

   c. From **host01**, log out as `oracle` user.

```
[oracle@host01 ~]$ logout
...
host01 login:
```

9. Return **host03** back to the original state.
   a. From **host03**, use the `vi` editor to edit `/etc/pam.d/sshd` and uncomment the entry to use the `pam_sepermit.so` module (remove the `#` sign).
   b. From **host03**, use the `vi` editor to edit `/etc/security/sepermit.conf` and remove the entry to allow only a single login for user `oracle`.

Practice 16-2: Configuring PAM to Prevent Non-root Login

Overview

In this practice, you configure a PAM authentication module on **host01** to prevent all non-root user logins.

Assumptions

- This practice is performed on **host01** and **host03** VMs.
- Open a terminal window on each system.
- Log in as the root user on **host01**.
- The prompts in the solution section include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. On **host01**, configure a PAM authentication module on **host01** to prevent all non-root user logins.
   a. Log in to **host01** as root. Password is oracle.

```
host01 login: root
Password: oracle
Last login: ...
[root@host01]#
```

   b. Use the cat command to view the login configuration file in /etc/pam.d.

```
# cat /etc/pam.d/login
#%PAM-1.0
auth [user_unknown=ignore success=ok ignore=ignore ...
auth        include     system-auth
account     required    pam_nologin.so
...
```

   - The login utility uses the pam_nologin.so authentication module as well as several other modules.

   c. Use the man pam_nologin command to view the man page for the pam_nologin authentication module.

```
# man pam_nologin
...
    pam_nologin - Prevent non-root users from login
...
    pam_nologin is a PAM module that prevents users from
    logging into the system when /etc/nologin exists. The
    contents of the /etc/nologin file are displayed to the
    user...no effect on the root user's ability to log in.
...
```

- Notice that this module uses a configuration file – `/etc/nologin` – which, if it exists, disables non-`root` logins.

d. Use the `vi` editor and create the `/etc/nologin` file with the following contents:

```
# vi /etc/nologin
No logins allowed at this time.
```

2. From **host03**, attempt to log in to **host01**.

a. Use the `ssh` command to connect to **host01** as user `oracle`.

```
[host03]# ssh oracle@host01
oracle@host01's password: oracle
No logins allowed at this time.
Connection closed by 192.0.2.101
```

- Notice that the connection is denied by the PAM authentication module.

To permit the non-`root` user logins, you can do either of the following:

- Delete the `/etc/nologin` file from **host01**.
- Remove the entry in the `/etc/pam.d/login` file to use the `pam_nologin.so` module.

3. From **host01**, permit non-`root` user logins from **host03** by using the `vi` editor to comment out the entry to use the `pam_nologin.so` module from the `/etc/pam.d/login` file.

- Comment out this line by inserting a # sign at the beginning of the line as follows:

```
# vi /etc/pam.d/login
...
account   required   pam_nologin.so        (current entry)
#account   required   pam_nologin.so       (insert # sign)
```

4. From **host03**, attempt to log in to **host01**.

Use the `ssh` command to connect to **host01** as user `oracle`.

```
[host03]# ssh oracle@host01
oracle@host01's password: oracle
No logins allowed at this time.
Connection closed by 192.0.2.101
```

- Notice that the connection is still denied by the PAM authentication module.

5. From **host01**, use the `grep` command to search for the string "`pam_nologin`" in all the files in the `/etc/pam.d` directory.

```
[host01]# grep pam_nologin /etc/pam.d/*
/etc/pam.d/login:#account   required   pam_nologin.so
/etc/pam.d/remote:account   required   pam_nologin.so
/etc/pam.d/sshd:account   required   pam_nologin.so
```

- Notice that this module also is called from the `remote` file and from the `sshd` file.
- You would need to comment out these lines in these two files as well.
- Alternatively, simply remove the `/etc/nologin` file to allow non-`root` logins.

6. Return **host01** back to the original state.

   a. Use the `rm` command to remove the `/etc/nologin` file.

```
# rm /etc/nologin
rm: remove regular file '/etc/nologin'? y
```

   b. Use the `vi` editor to edit `/etc/pam.d/login` and uncomment the entry to use the `pam_nologin.so` module (remove the `#` sign).

```
# vi /etc/pam.d/login
...
#account   required   pam_nologin.so       (current entry)
account   required   pam_nologin.so        (remove # sign)
```

Practices for Lesson 17: Security Administration

Chapter 17

Practices for Lesson 17: Security Administration

Practices Overview

In these practices, you:

- Create `chroot` jails
- Configure `iptables` from the GUI and the command line
- Configure a TCP wrapper

Practice 17-1: Configuring a `chroot` Jail

Overview

In this practice, you configure a `chroot` jail and copy all files required to run the `/bin/bash` shell in the `chroot` jail.

Assumptions

You are the `oracle` user on **host03** VM.

Tasks

1. Use the `whoami` command to confirm that you are logged in as the `oracle` user.

```
$ whoami
oracle
```

   - If you are the `root` user, enter `exit` or `logout` to become the `oracle` user.

2. Use the `cd` command to change to your home directory.

```
$ cd
$ pwd
/home/oracle
```

3. Use the `mkdir` command to make a `jail` directory in your home directory.

```
$ mkdir jail
```

4. Use the `su` command to become the `root` user. Password is `oracle`.

```
$ su –
Password: oracle
```

5. Use the `echo` command to display the value of the `SHELL` variable.

```
# echo $SHELL
/bin/bash
```

   - In this example, `SHELL=/bin/bash`.

6. As the `root` user, use the `chroot` command to create a `chroot` jail in the `/home/oracle/jail` directory.

```
# chroot /home/oracle/jail
chroot: failed to run command '/bin/bash': No such file or
directory
```

   - Notice that if you do not specify a command as an argument, `chroot` attempts to run the value of the `SHELL` variable, `/bin/bash`, in the chroot jail directory, `/home/oracle/jail`.
   - The command failed because `/bin/bash` was not found in `/home/oracle/jail`.

7. Use the exit command to log off as root and return to oracle user login.

```
# exit
logout
$ whoami
oracle
```

8. As the oracle user, create a new root directory structure and copy all necessary files into the new root directory.

   a. Within the jail directory in your home directory, use the mkdir command to make a bin directory.

```
$ cd ~/jail
$ mkdir bin
```

   b. Use the cp command to copy /bin/bash into ~/jail/bin.

```
$ cp /bin/bash ~/jail/bin
```

   c. Use the ldd command to determine which shared libraries are required by /bin/bash.

```
$ ldd /bin/bash
linux-vdso.so.1 => (0x0000...)
libtinfo.so.5 => /lib64/libtinfo.so.5 (0x0000...)
libdl.so.2 => /lib64/libdl.so.2 (0x0000...)
libc.so.6 => /lib64/libc.so.6 (0x0000...)
/lib64/ld-linux-x86-64.so.2 (0x0000...)
```

   - In this example, there are four shared library files in /lib64 used by /bin/bash.

   d. Within the jail directory in your home directory, use the mkdir command to make a lib64 directory.

```
$ cd ~/jail
$ mkdir lib64
```

   e. Use the cp command to copy the four shared library files required for /bin/bash from /lib64 to ~/jail/lib64.

```
$ cp /lib64/libtinfo.so.5 ~/jail/lib64
$ cp /lib64/libdl.so.2 ~/jail/lib64
$ cp /lib64/libc.so.6 ~/jail/lib64
$ cp /lib64/ld-linux-x86-64.so.2 ~/jail/lib64
```

9. Use the su command to become the root user. Password is oracle.

```
$ su -
Password: oracle
```

10. As the root user, use the chroot command to create a chroot jail in the /home/oracle/jail directory.

```
# chroot /home/oracle/jail
```

- Notice that the chroot command was successful—no errors occurred and the /bin/bash program executed.

11. Use the pwd command to display the current directory.

```
# pwd
/
```

- Notice that the output indicates that the current directory is the root-level directory even though the actual directory is /home/oracle/jail.

12. Use the exit command to exit the chroot jail.

```
# exit
```

Practice 17-2: Configuring a `chroot` Jail for `ftp` Users

Overview

In this practice, you:

- Confirm that `anonymous` ftp users are placed in a chroot jail on a `vsftpd` server by default
- Configure an SELinux Boolean to allow local ftp users
- Configure local users to be placed in a chroot jail

Assumptions

- This practice is performed on **host01** and **host03** VMs.
- Open a terminal window on each system.
- Log in as the `root` user on each system.
- The prompts in the solution section include either **host01** or **host03** to indicate which system to enter the command from.
- SELinux is covered in a later course but some functionality is used in this practice.

Tasks

1. Confirm that `anonymous` ftp users are placed in a `chroot` jail by default.

   a. From **host03**, use the `ls` command to list the contents of the `/var/ftp` directory.

```
[host03]# ls -l /var/ftp
drwxr-xr-x...  pub
```

   - Notice that `/var/ftp` contains a single directory, `pub`.

   b. From **host03**, use the `ls` command to list the contents of `/var/ftp/pub`.

```
[host03]# ls /var/ftp/pub
test_file
```

   - Notice that the `/var/ftp/pub` directory contains a single file, `test_file`.

   c. From **host01**, `ftp` to **host03** as `anonymous` user. Press Enter when prompted for a password.

```
[host01]# ftp host03
Connected to host03: (192.0.2.103).
220 (vsFTPd 2.2.2)
Name...: anonymous
331 Please specify the password.
Password: ENTER
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

d. Use the `ls` command to list the contents of the current directory.

```
ftp> ls
...
drwxr-xr-x ... pub
226 Directory send OK.
ftp>
```

- Notice that the current directory contains a single directory, `pub`.

e. Use the `ls` command to list the contents of the `pub` directory.

```
ftp> ls pub
...
-rw-r--r-- ... test_file
226 Directory send OK.
ftp>
```

- Notice that the `pub` directory contains a single file, `test_file`.
- This confirms that the current location of the `anonymous` FTP user is `/var/ftp` on **host03**.

f. Use the `pwd` command to display the current directory.

```
ftp> pwd
257 "/"
ftp>
```

- Notice that the output indicates that the current directory is the root-level directory even though the actual directory is `/var/ftp`.
- This confirms that `anonymous` users are placed in a `chroot` jail by default.

g. Use the `quit` command to exit `ftp`.

```
ftp> quit
221 Goodbye.
```

2. From **host01**, attempt to `ftp` to **host03** as the `oracle` user. Password is `oracle`.

a. Use the `ftp host03` command to connect as the `oracle` user. Password is `oracle`.

```
[host01]# ftp host03
Connected to host03: (192.0.2.103).
220 (vsFTPd 2.2.2)
Name...: oracle
331 Please specify the password.
Password: oracle
500 OOPS: cannot change directory:/home/oracle
Login failed.
ftp>
```

- Notice that login failed for the `oracle` user.

b. Use the `quit` command to exit `ftp`.

```
ftp> quit
221 Goodbye.
```

3. On **host03**, use the `setenforce` command to change SELinux to permissive mode. Use the `getenforce` command before and after to confirm the change.

```
[host03]# getenforce
Enforcing
[host03]# setenforce 0
[host03]# getenforce
Permissive
```

4. From **host01**, attempt to `ftp` to **host03** as the `oracle` user. Password is `oracle`.

a. From **host01**, `ftp` to **host03** as the `oracle` user. Password is `oracle`.

```
[host01]# ftp host03
Connected to host03: (192.0.2.103).
220 (vsFTPd 2.2.2)
Name...: oracle
331 Please specify the password.
Password: oracle
230 Login successful.
...
ftp>
```

- Notice that login was successful this time.
- Setting SELinux to permissive mode allows the `oracle` user to connect to the FTP server.

b. Use the `quit` command to exit `ftp`.

```
ftp> quit
221 Goodbye.
```

5. From **host03**, determine which SELinux Boolean(s) denied the `oracle` user connection to the FTP server.

a. Run the `yum install` command to install the `policycoreutils-python` package.

- This package provides the `semanage` command.

```
# yum install policycoreutils-python
...
Transaction Summary
=============================================================
Install       6 Package(s)
Total download size: 1.2 M
Installed size: 4.4 M
Is this ok [y/N]: y
```

```
...
Complete!
```

b. Use the `semanage` command to list FTP-related Booleans.

```
# semanage boolean -l | grep ftp
ftp_home_dir           -> off    Allow ftp to read and write
files in the user home directories
...
```

- Notice that the `ftp_home_dir` Boolean is set to `off`.
- This Boolean seems to be the cause of the `oracle` user connection denial.

c. Use the `setenforce` command to change SELinux to enforcing mode. Use the `getenforce` command to confirm the change.

```
# setenforce 1
# getenforce
Enforcing
```

d. Use the `setsebool` command to set the `ftp_home_dir` Boolean to `on`.

```
# setsebool ftp_home_dir on
```

e. Use the `getsebool` command to confirm that the current status of the `ftp_home_dir` Boolean is `on`.

```
# getsebool ftp_home_dir
ftp_home_dir --> on
```

6. From **host01**, attempt to `ftp` to **host03** as the `oracle` user. Password is `oracle`.

a. From **host01**, `ftp` to **host03** as the `oracle` user. Password is `oracle`.

```
[host01]# ftp host03
Connected to host03: (192.0.2.103).
220 (vsFTPd 2.2.2)
Name...: oracle
331 Please specify the password.
Password: oracle
230 Login successful.
...
ftp>
```

- Notice that login was successful this time.
- Setting the `ftp_home_dir` SELinux Boolean to `on` allows the `oracle` user to connect to the FTP server even when SELinux policy is enforced.

b. Use the `pwd` command to display the current directory. Use the `cd` command to move up one level in the directory structure.

```
ftp> pwd
257 "/home/oracle"
ftp> cd ..
250 Directory successfully changed.
ftp> pwd
257 "/home"
ftp>
```

- The output indicates that the `oracle` user was placed in its home directory.
- This is the case for all local users that access a `vsftpd` server; they are placed in their home directory by default, and not in a `chroot` jail.

c. Use the `quit` command to exit `ftp`.

```
ftp> quit
221 Goodbye.
```

7. On **host03**, enable options in the `/etc/vsftpd/vsftpd.conf` file to put local users in a `chroot` jail.

a. Use the `vi` editor to make the following changes:

- Set the `chroot_local_user` directive to `YES`.
- Ensure the following `chroot` directives are commented out (preceded with a `#` sign, as shown):
  - `#chroot_list_enable=YES`
  - `#chroot_list_file=/etc/vsftpd/chroot_list`

```
[host03]# vi /etc/vsftpd/vsftpd.conf
chroot_local_user=YES
#chroot_list_enable=YES
#chroot_list_file=/etc/vsftpd/chroot_list
```

- Normally, after making changes to the `vsftpd.conf` file, you would need to restart the `vsftpd` service.
- However, because you configured `vsftpd` to run under `xinetd` in Practice 14-5, you do not need to restart the `vsftpd` service.

8. From **host01**, attempt to `ftp` to **host03** as the `oracle` user. Password is `oracle`.

   a. From **host01**, `ftp` to **host03** as the `oracle` user. Password is `oracle`.

```
[host01]# ftp host03
Connected to host03: (192.0.2.103).
220 (vsFTPd 2.2.2)
Name...: oracle
331 Please specify the password.
Password: oracle
230 Login successful.
...
ftp>
```

   b. Use the `pwd` command to display the current directory. Use the `ls` command to list the contents of the current directory.

```
ftp> pwd
257 "/"
ftp> ls
...
drwxr-xr-x ...  Desktop
drwxr-xr-x ...  Documents
drwxr-xr-x ...  Downloads
drwxr-xr-x ...  Music
...
```

   - The output indicates that the current directory is the root-level directory even though the actual directory is `/home/oracle` on the `vsftpd` server, **host03**.

   c. Use the `quit` command to exit `ftp`.

```
ftp> quit
221 Goodbye.
```

Practice 17-3: Configuring `iptables`

Overview

In this practice, you use the Firewall Configuration GUI as well as the `iptables` command to allow a client system to mount an NFS file system.

Assumptions

- This practice is performed on **host01** and **host03** VMs.
- Open a terminal window on each system.
- Log in as the `root` user on each system.
- The prompts in the solution section include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. From **host01**, unmount NFS file systems.
   a. Use the `umount` command to unmount `/remote_dev`.

```
[host01]# umount /remote_dev
```

   b. Use the `df` command to display the mounted file systems.

```
[host01]# df -h
Filesystem    Size  Used  Avail  Use%  Mounted on
/dev/xvda2    5.7G  2.4G   3.1G   44%  /
...
```

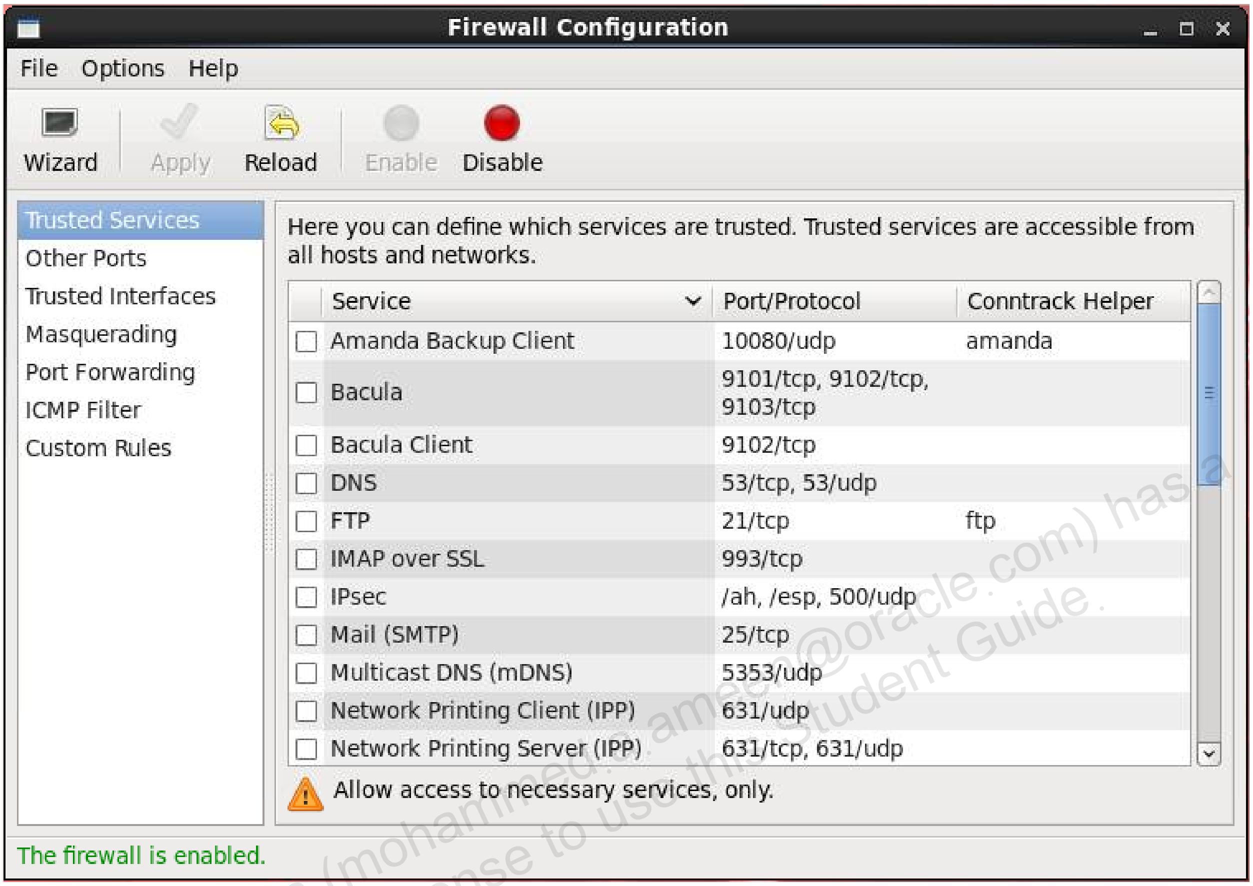
   - Notice that no NFS mounted file systems are currently mounted.

2. From **host03**, use the `system-config-firewall` command to display the Firewall Configuration GUI.

```
[host03]# system-config-firewall
```

   - Click **Close** if the Firewall Configuration Startup message appears.

- The GUI appears.



- In this example, the GUI shows the firewall is enabled. The firewall on your system may be disabled.

3. Enable the firewall.
   a. If the **Disable** button is active, click **Disable**.
   b. Click the **Enable** button.
   c. Click the **Apply** button.
   d. Click **Yes** to override any existing firewall configuration.
      - The firewall is now enabled.
   e. Select **File > Quit** to close the GUI.
4. From **host03**, use the `service` command to view the status of the `iptables` service.

```
[host03]# service iptables status
Table: filter
Chain INPUT (policy ACCEPT)
num  target   prot  opt  source      destination
1    ACCEPT   all   --   0.0.0.0/0   0.0.0.0/0      state ...
...
Chain FORWARD (policy ACCEPT)
num  target   prot  opt  source      destination
```

```
1    REJECT   all   --   0.0.0.0/0  0.0.0.0/0     reject...
...
Chain OUTPUT (policy ACCEPT)
num  target   prot  opt  source     destination
```

- This shows that the `iptables` service is running.

5. From **host03**, use the `showmount` command to list the NFS exported file systems.

```
[host03]# showmount -e
/Dev           *
```

- In this example, one NFS file system is exported.

6. From **host01**, attempt to mount an NFS file system.

   a. Use the `mount` command to mount the exported `/Dev` NFS file system from **host03** with `rw` and `nosuid` options on the local mountpoint, `/remote_dev`.

```
[host01]# mount -t nfs -o rw,nosuid host03:/Dev /remote_dev
```

   - With the firewall enabled, the `mount` command hangs.
   - Press **Ctrl + C** to abort the `mount` command.

7. From **host03**, run the `iptables -L` command.

```
[host03]# iptables -L
Chain INPUT (policy ACCEPT)
target   prot  opt  source     destination
ACCEPT   all   --   0.0.0.0/0  0.0.0.0/0     state RELATED...
...
```

- Notice that the output is similar to output from the `service iptables status` command.

8. Run the `iptables -L` command again, but this time pipe the output to `grep` and search for "nfs".

```
[host03]# iptables -L | grep nfs
```

- There are currently no rules containing the string "`nfs`".

9. From **host03**, use the Firewall Configuration GUI to configure NFS4 as a trusted service.

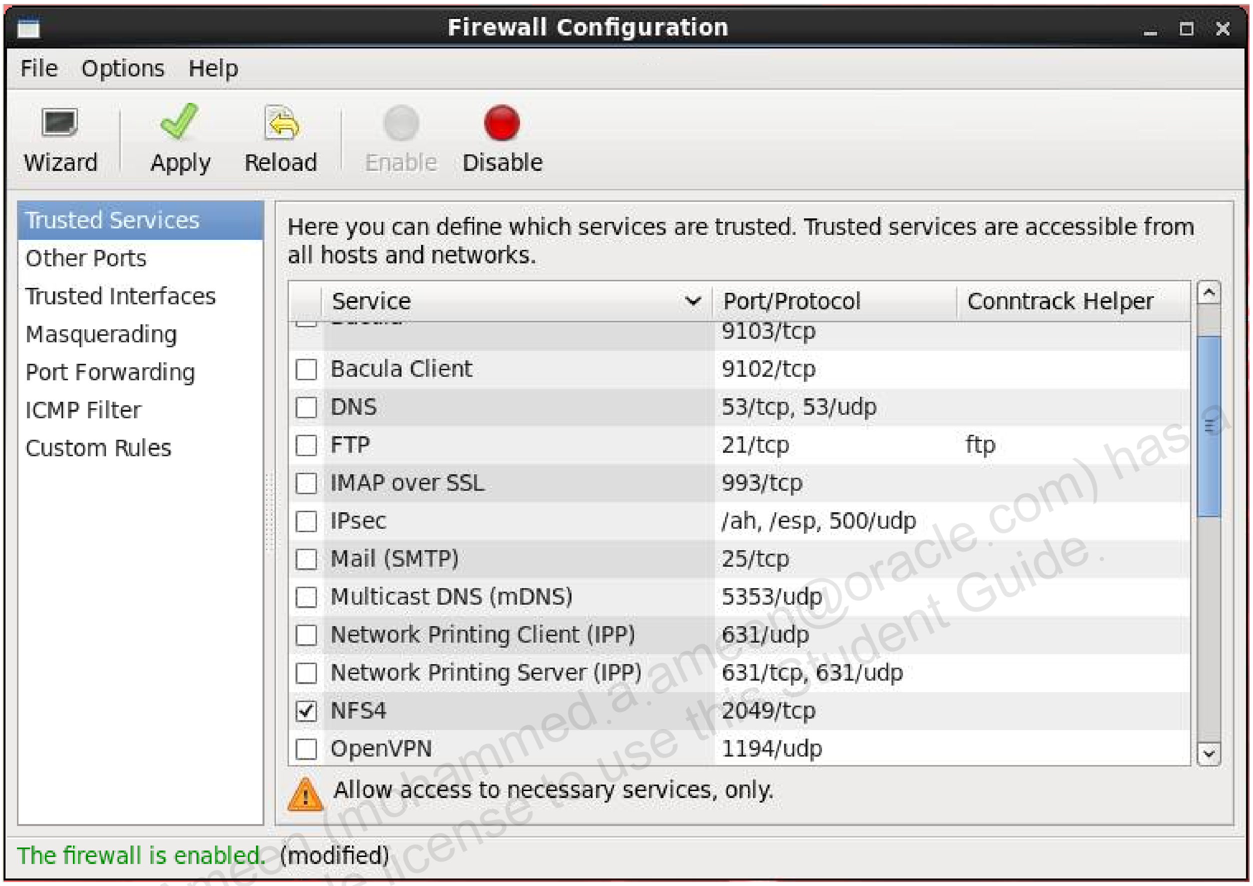
   a. Use the `system-config-firewall` command to start the Firewall Configuration GUI.

```
[host03]# system-config-firewall
```

- The GUI appears.

b. From the GUI, select the check box next to the NFS4 service, as shown.

- Scroll down if necessary to display the NFS4 check box.



- This allows the NFS services to be trusted, and not blocked by the firewall.

c. Save the changes made from the GUI.

1) Click the **Apply** button.

2) Click **Yes** to override any existing firewall configuration.

- The change is now active.

d. Select **File > Quit** to close the GUI.

10. From **host01**, attempt to mount `host03:/Dev` on `/remote_dev`.

a. Use the `mount` command to mount `host03:/Dev` on `/remote_dev` using the options as shown:

```
[host01]# mount -t nfs -o rw,nosuid host03:/Dev /remote_dev
```

- The `mount` command is successful this time.

b. Use the `df` command to display the mounted file systems.

```
[host01]# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
/dev/xvda2     5.7G  2.4G   3.1G   44%  /
...
host03:/Dev  1008M   34M   924M    4%  /remote_dev
```

- Notice that the `host03:/Dev` file system is mounted on local file system `/remote_dev`.

c. Use the `umount` command to unmount `/remote_dev`.

```
[host01]# umount /remote_dev
```

11. From **host03**, view and modify the `iptable` rules.

a. Run the `iptables -L` command, pipe the output to `grep`, and search for "`nfs`".

```
[host03]# iptables -L | grep nfs
ACCEPT   tcp  --  anywhere  anywhere  state NEW tcp dpt:nfs
```

- Now there is an "`nfs`" rule, which states to accept all incoming TCP traffic from anywhere when the destination service is `nfs`.
- This rule was created by checking **NFS4** in the Firewall Configuration GUI.

b. Run `iptables -h` to display all options.

```
[host03]# iptables -h
iptables v1.4.7
Usage: iptables -[AD] chain rule-specification [options]
       iptables -I chain [rulenum] rule-specification ...
...
```

- From the help, notice that the `--line-numbers` option displays line numbers.
- Also notice that the `-D` option deletes a rule.

c. Run the `iptables` command to list only those rules in the `INPUT` chain and include line numbers.

```
[host03]# iptables -L INPUT --line-numbers
Chain INPUT (policy ACCEPT)
num  target   prot  opt  source     destination
1    ACCEPT   all   --   0.0.0.0/0  0.0.0.0/0    state ...
...
4    ACCEPT   tcp   --   0.0.0.0/0  0.0.0.0/0    state ...nfs
...
```

- Notice that the rule to accept `nfs` traffic is line number 4.

d. Use the `iptables` command to delete line number 4 from the `INPUT` chain. Run the previous command to confirm that the rule is deleted.

```
[host03]# iptables –D INPUT 4
[host03]# iptables –L INPUT --line-numbers
Chain INPUT (policy ACCEPT)
num  target   prot  opt  source     destination
1    ACCEPT   all   --   0.0.0.0/0  0.0.0.0/0    state ...
...
```

- Notice that the `nfs` rule in line number 4 has been deleted.
- Any rules below line number 4 have been renumbered.

e. Use the `service` command to save the `iptables` rules.

- Always save after making any rule changes.

```
[host03]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables...
```

f. Use the `service` command to restart the `iptables` service.

- After making any rule changes, restart the `iptables` service.

```
[host03]# service iptables restart
iptables: Setting chains to policy ACCEPT: filter   [  OK  ]
iptables: Flushing firewall rules:                  [  OK  ]
iptables: Unloading modules:                        [  OK  ]
iptables: Applying firewall rules:                  [  OK  ]
```

12. From **host01**, use the `mount` command to mount `host03:/Dev` on `/remote_dev` using the options as shown:

```
[host01]# mount –t nfs –o rw,nosuid host03:/Dev /remote_dev
```

- As expected, the `mount` command hangs on **host01**, when NFS4 is not trusted.
- Press **Ctrl + C** to abort.

13. From **host03**, create an "`nfs`" rule to trust NFS4.

a. Use the `iptables` command to insert the "`nfs`" rule after line 4 with the following characteristics from the command line:

- Chain = `INPUT`
- Protocol = `tcp`
- State = `NEW`
- Destination port = `nfs`
- Target = `ACCEPT`

```
[host03]# iptables –I INPUT 4 –p tcp –m state --state NEW --
dport nfs –j ACCEPT
```

- This rule accepts incoming `tcp` traffic for `nfs`.

b. Use the service command to save the iptables rules.

```
[host03]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables...
```

c. Use the cat command to view the /etc/sysconfig/iptables file.

```
[host03]# cat /etc/sysconfig/iptables
...
-A INPUT -p tcp -m state --state NEW -m tcp --dport 2049 -j
ACCEPT
...
```

- Notice that the new rule has the actual port number, 2049, for nfs.

d. Use the grep command to search for 2049 in the /etc/services file.

```
[host03]# grep 2049 /etc/services
nfs  2049/tcp ...
```

e. Use the service command to restart the iptables service.

```
[host03]# service iptables restart
iptables: Flushing firewall rules:                         [  OK  ]
iptables: Setting chains to policy ACCEPT: filter          [  OK  ]
iptables: Unloading modules:                               [  OK  ]
iptables: Applying firewall rules:                         [  OK  ]
```

14. From **host01**, attempt to mount the NFS file system.

a. Use the mount command to mount host03:/Dev on /remote_dev.

```
[host01]# mount -t nfs -o rw,nosuid host03:/Dev /remote_dev
```

- The mount command is successful this time.

b. From **host01**, use the df command to display the mounted file systems.

```
[host01]# df -h
Filesystem    Size  Used  Avail  Use%  Mounted on
/dev/xvda2    5.7G  2.4G   3.1G   44%  /
...
host03:/Dev  1008M   34M   924M    4%  /remote_dev
```

- Notice that the host03:/Dev file system is mounted on local file system /remote_dev.

Practice 17-4: Configuring a TCP Wrapper

Overview

In this practice, you configure a TCP wrapper to deny one system from using OpenSSH utilities to connect to another system. You also create a custom log file to capture connection attempts that are denied.

Assumptions

- This practice is performed on **host01** and **host03** VMs.
- Open a terminal window on each system.
- Log in as the `root` user on each system.
- The prompts in the solution section include either **host01** or **host03** to indicate which system to enter the command from.

Tasks

1. From **host01**, confirm that you can use the `ssh` command to connect to **host03**. Password is `oracle`. Use the `logout` command to log off after confirming that you can connect.

```
[host01]# ssh host03
root@host03's password: oracle
Last login: ...
[root@host03 ~]# logout
Connection to host03 closed.
[host01]#
```

2. On **host03**, use the `vi` editor to edit `/etc/hosts.deny` and add the following entry.

```
[host03]# vi /etc/hosts.deny
sshd : 192.0.2.101
```

- This entry denies **host01** (`192.0.2.101`) from using the OpenSSH utilities to connect to **host03**.

3. From **host01**, attempt to use the `ssh` command to connect to **host03**.

```
[host01]# ssh host03
ssh_exchange_identification: Connection closed by remote host
```

- This time you are denied connecting using `ssh`.

4. On **host03**, use the `vi` editor to edit `/etc/hosts.deny` and modify the entry as follows.

```
[host03]# vi /etc/hosts.deny
sshd : 192.0.2.101 : spawn /bin/echo "%c tried to connect to %d
and was blocked." >> /var/log/tcpwrappers.log
```

5. From **host01**, attempt to use the `ssh` command to connect to **host03**.

```
[host01]# ssh host03
ssh_exchange_identification: Connection closed by remote host
```

- You are still denied connecting and a message was written to a log file.

6. On **host03**, use the `cat` command to view the `/var/log/tcpwrappers.log` file.

```
[host03]# cat /var/log/tcpwrappers.log
192.0.2.101 tried to connect to sshd and was blocked.
```

7. Restore the system.

   a. On **host03**, use the `vi` editor to edit `/etc/hosts.deny` and delete the entry created earlier in this practice.

   - Delete the strike-through line as follows:

```
[host03]# vi /etc/hosts.deny
sshd : 192.0.2.101 : spawn /bin/echo "%c tried to connect to %d and was blocked." >> /var/log/tcpwrappers.log
```

   b. From **host01**, confirm that you can use the `ssh` command to connect to **host03**. Password is `oracle`. Use the `logout` command to log off after confirming that you can connect.

```
[host01]# ssh host03
root@host03's password: oracle
Last login: ...
[root@host03 ~]# hostname
host03.example.com
[root@host03 ~]# logout
Connection to host03 closed.
[host01]#
```

Practices for Lesson 18: Oracle on Oracle

Chapter 18

Practices for Lesson 18: Oracle on Oracle

Practices Overview

In these practices, you install and run Oracle RDBMS Pre-Install RPM for Oracle Linux 6. You also configure ASMLib.

Practice 18-1: Using `sftp` to Upload `oracle-rdbms-server` Package

Overview

In this practice, you use `sftp` to upload the `oracle-rdbms-server-12cR1-preinstall` package from the **dom0** to the **host03** VM. Normally, you obtain these packages from ULN or from the Public Yum Server.

Assumptions

- You are the `root` user on the **host03** VM.
- You are the `root` user on **dom0**.

Tasks

1. Use the `sftp` to transfer the `oracle-rdbms-server-12cR1-preinstall` package from **dom0** to **host03**.
   a. From **dom0**, use the `cd` command to change to the `/OVS/seed_pool/sfws` directory.

   ```
   [dom0]# cd /OVS/seed_pool/sfws
   ```

   b. Use the `ls` command to view the directory for the `oracle*` package.

   ```
   [dom0]# ls oracle*
   oracle-rdbms-server-12cR1-preinstall-1.0-8.el6.x86_64.rpm
   ```

   c. Use the `sftp` command to connect to **host03** as `root`. Password is `oracle`.

   ```
   [dom0]# sftp host03
   root@host03's password: oracle
   sftp>
   ```

   d. Use the `mput` command to copy the `oracle*` packages to **host03**.

   ```
   sftp> put oracle*
   Uploading oracle-rdbms-server-12cR1-preinstall-1.0-
   8.el6.x86_64.rpm to /root/...
   ...
   ```

   e. Use the `quit` command to exit `sftp`.

   ```
   sftp> quit
   ```

Practice 18-2: Installing and Running Oracle RDBMS Pre-Install

Overview

In this practice, you install the `oracle-rdbms-server-12cR1-preinstall` package, run the "`verify`" program, and view the results.

Assumptions

- You are the `root` user on **host03** VM.
- All commands are issued from **host03** VM.

Tasks

1. Install the `oracle-rdbms-server-12cR1-preinstall` package and view the results of the installation.
   a. Use the `cd` command to change to the `root` user's home directory, and then use the `yum` command to install the `oracle-rdbms-server-12cR1-preinstall` package.

```
# cd
# yum install oracle-rdbms-server-12cR1-preinstall-1.0-
8.el6.x86_64.rpm
...
Setting up Install Process
...
Resolving Dependencies
...
Transaction Summary
=========================================================
Install       7 Package(s)
Total size: 7.3 M
Total download size: 7.3 M
Installed size: 23 M
Is this ok [y/N] y
Downloading Packages:
...
Installed:
  oracle-rdbms-server-12cR1-preinstall.x86_64 0:1.0-8.el6
Dependency Installed:
...
Complete!
```

b. Use the `find` command to locate all `oracle-rdbms*` files.

```
# find / -name "*oracle-rdbms*"
/usr/bin/oracle-rdbms-server-12cR1-preinstall-verify
/etc/sysconfig/oracle-rdbms-server-12cR1-preinstall
/etc/sysconfig/oracle-rdbms-server-12cR1-preinstall/oracle-
rdbms-server-12cR1-preinstall.conf
...
/etc/rc.d/init.d/oracle-rdbms-server-12cR1-preinstall-firstboot
/etc/security/limits.d/oracle-rdbms-server-12cR1-preinstall.conf
/var/log/oracle-rdbms-server-12cR1-preinstall
...
```

c. Use the `ls -l` command to display the file type of the `/usr/bin/oracle-rdbms*` file.

```
# ls –l /usr/bin/oracle-rdbms*
lrwxrwxrwx ... /usr/bin/oracle... -> /etc/sysconfig/oracle...
```

- Notice that this file is a symbolic link to a file in the `/etc/sysconfig/oracle-rdbms*` directory.

d. Change to the `/etc/sysconfig/oracle-rdbms*` directory and view the contents of the directory.

```
# cd /etc/sysconfig/oracle-rdbms*
# ls
oracle-rdbms-server-12cR1-preinstall.conf
oracle-rdbms-server-12cR1-preinstall.param
oracle-rdbms-server-12cR1-preinstall-verify
```

e. Use the `less` command to view each of the files in the `/etc/sysconfig/oracle-rdbms*` directory.

```
# less oracle-rdbms-server-12cR1-preinstall.conf
...
# less oracle-rdbms-server-12cR1-preinstall.param
...
# less oracle-rdbms-server-12cR1-preinstall-verify
...
```

- Notice that the `*.param` file is the main configuration file.
- Notice that the `*verify` file is the Bash script that modifies settings.

2. View the modifications made by the `oracle-rdbms-server-12cR1-preinstall-verify` script.
   - Note that previous versions of the `oracle-rdbms-server-...-verify` script needed to be run manually. With this version, however, the script is executed automatically when the RPM package is installed.

a. Use the `less` command to view the Oracle RDBMS Pre-install log file, `/var/log/oracle-rdbms-server-12cR1-preinstall/results/orakernel.log` (sample output shown).

```
# cd /var/log/oracle-rdbms-server-12cR1-preinstall/results
# less orakernel.log
Adding group oinstall with gid 54321
Adding group dba
User oracle is already present
uid=500(oracle) gid=500(oracle) ...
Creating oracle user passed

Verifying kernel parameters as per Oracle recommendations...
fs.file-max          6815744
kernel.sem           250 32000 100 128
kernel.shmmni        4096
...
Setting kernel parameters as per oracle recommendations...
Altered file /etc/sysctl.conf
Original file backed up at /etc/sysctl.conf.orabackup
Verifying & setting of kernel parameters passed

Setting user limits using /etc/security/limits.d/oracle...

Verifying oracle user OS limits as per Oracle recommendations...
Adding oracle soft nofile 1024
Adding oracle hard nofile 65536
Adding oracle soft nproc 16384
Adding oracle hard nproc 16384
Adding oracle soft stack 10240
Adding oracle hard stack 32768
Setting oracle user OS limits as per oracle recommendations...
Altered file /etc/security/limits.d/oracle-rdbms...
Original file backed up at
Verifying & setting of user limits passed

Verifying kernel boot parameters as per Oracle ...
...
Altered file /boot/grub/grub.conf
Original file backed up at /boot/grub/grub.conf.orabackup
Verifying & setting of boot parameters passed

Taking a backup of old config files under /var/log/oracle...
```

- Notice that the required user and groups are created if necessary.
- Notice that kernel parameters are set and /etc/sysctl.conf is backed up beforehand.
- Notice that oracle user OS limits are set in /etc/security/limits.d/oracle-rdbms-server-12cR1-preinstall.conf.
- Notice that kernel boot parameters are set and /boot/grub.grub.conf is backed up beforehand.

b. Use the find command to list the files backed up before settings were changed, *orabackup.

```
# find / -name "*orabackup"
/etc/sysctl.conf.orabackup
/boot/grub/grub.conf.orabackup
```

c. Use the diff command to view the changes made in the grub.conf file.

```
# diff /boot/grub/grub.conf /boot/grub/grub/conf.orabackup
...
```

- Notice that the "numa=off" kernel parameter is added to the grub.conf file.

d. Use the wc -l command to display the number of lines in the /etc/sysctl.conf files.

```
# wc -l /etc/sysctl.conf /etc/sysctl.conf.orabackup
72 /etc/sysctl.conf
40 /etc/sysctl.conf.orabackup
...
```

- Notice that 32 new lines are added to the sysctl.conf file.

e. Use the diff command to view the changes made in the sysctl.conf file.

```
# diff /etc/sysctl.conf /etc/sysctl.conf.orabackup
...
```

f. Use the cat command to view the limits set in the /etc/security/limits.d/oracle-rdbms-server-12cR1-preinstall.conf file.

```
# cat /etc/security/limits.d/oracle-rdbms-server-12cR1-
preinstall.conf
...
```

Practice 18-3: Preparing Disks for ASM Use

Overview

In this practice, you:

- Remove all NFS file systems
- Unmount all file systems on `/dev/xvdb` and `/dev/xvdb`
- Delete all existing partitions to use the disks for ASM
- Create one partition using the entire disk on `/dev/xvdb`
- Create one partition using the entire disk on `/dev/xvdd`

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Use the `vi` editor to delete all entries from `/etc/exports`.

```
# vi /etc/exports
Delete all entries and save the file
```

2. Use the `service` command to restart the `nfs` service.

```
# service nfs restart
...
```

3. Use the `showmount` command to confirm that there are no exported file systems.

```
# showmount -e
Export list for host03.example.com
```

- No NFS file systems are being exported.

4. Unmount all file systems on `/dev/xvdb` and `/dev/xvdd`.

   a. Use the `df` command to list the mounted partitions.

```
# df -h
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvda2      5.7G  4.7G   760M   87%  /
...
/dev/xvdb1     1008M   34M   924M    4%  /Dev
/dev/xvdd1      992M  1.3M   940M    1%  /Test
```

- In this example, the following partitions need to be unmounted (your system may be different).
  - `/dev/xvdb1` mounted on `/Dev`
  - `/dev/xvdd1` mounted on `/Test`

b. Use the `umount` command to unmount file systems on `/dev/xvdb1` and `/dev/xvdd1`.

```
# umount /Dev
# umount /Test
```

5. Delete all partitions on `/dev/xvdb` and `/dev/xvdd`.

a. Use the `fdisk` command to display the partition table on `/dev/xvdb` and then delete all the partitions.

```
# fdisk /dev/xvdb
...
Command (m for help): p
Disk /dev/xvdb: 5368 MB, 5368709120 bytes
...
    Device Boot    Start        End       Blocks     Id    System
/dev/xvdb1           ...
/dev/xvdb2           ...

Command (m for help): d
Partition number (1-4): 1

Command (m for help): d
Selected partition 2

Command (m for help): p
...
    Device Boot    Start        End       Blocks     Id    System

Command (m for help): w
...
```

- In this example, two partitions are deleted (your system may be different).

b. Use the `fdisk` command to display the partition table on `/dev/xvdd` and then delete all the partitions.

```
# fdisk /dev/xvdd
...
Command (m for help): p
Disk /dev/xvdb: 5368 MB, 5368709120 bytes
...
    Device Boot    Start        End       Blocks     Id    System
/dev/xvdd1           ...
/dev/xvdd2           ...

Command (m for help): d
```

```
Partition number (1-4): 1

Command (m for help): d
Selected partition 2

Command (m for help): p
...
    Device Boot   Start        End      Blocks    Id    System

Command (m for help): w
...
```

- In this example, two partitions are deleted (your system may be different).

6. Use the `vi` editor to delete entries pertaining to file systems on `/dev/xvdb` and `/dev/xvdd` from the `/etc/fstab` file.

```
# vi /etc/fstab
LABEL=Dev   /Dev   ... (delete this entry)
LABEL=Test  /Test  ... (delete this entry)
```

- In this example, delete the two entries shown (your system may be different).

7. In preparation for the ASMLib lab, you must disable SELinux completely and reboot.
   - ASMlib is not supported with SElinux enabled (it is a known issue).

   Use the `vi` editor to edit `/etc/selinux/config` as follows:

```
# vi /etc/selinux/config
SELINUX=enforcing (old value)
SELINUX=disabled  (new value)
```

8. Reboot your system and log back in.

```
# reboot
...
```

- After you reboot your system, your vnc session closes.

a. Connect to **host03** by using VNC.

   1) Run the `vncviewer&` command.

```
# vncviewer&
```

   - The **VNC Viewer: Connection Details** dialog box is displayed.

   2) Enter the command, localhost:<port_number>, substituting the correct port number for the **host03** guest. For example, if the port number is `5903`, enter the following and click OK:

```
localhost:5903
```

b. Select `Oracle Student` from the GNOME login window; password is `oracle`.

c. Right-click the GNOME desktop and select **Open in Terminal** from the pop-up menu.

d. In the terminal window, become the `root` user by entering the `su -` command followed by the `root` password, `oracle`.

```
$ su –
Password: oracle
```

9. Create a partition on `/dev/xvdb`.

a. Use the `fdisk` command to partition `/dev/xvdb`.

```
# fdisk /dev/xvdb
WARNING: DOS-compatible mode is deprecated. It’s strongly
recommended to switch off the mode (command ‘c’) and change
display units to sectors (command ‘u’).
Command (m for help):
```

b. Add a new **primary** partition, giving the partition number **1**.

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
```

c. Continue adding the new partition, using the entire disk as follows:

```
First cylinder (1-652, default 1): ENTER
Using default value 1
Last cylinder, +cylinders or +size{K,M,G} (1-652, default 652):
ENTER
```

d. Save the new partition table.

```
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

10. Create a partition on `/dev/xvdd`.

a. Use the `fdisk` command to partition `/dev/xvdd`.

```
# fdisk /dev/xvdd
WARNING: DOS-compatible mode is deprecated. It’s strongly
recommended to switch off the mode (command ‘c’) and change
display units to sectors (command ‘u’).
Command (m for help):
```

b. Add a new **primary** partition, giving the partition number **1**.

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
```

c. Continue adding the new partition, using the entire disk as follows:

```
First cylinder (1-652, default 1): ENTER
Using default value 1
Last cylinder, +cylinders or +size{K,M,G} (1-652, default 652):
ENTER
```

d. Save the new partition table.

```
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

Practice 18-4: Installing and Configuring ASMLib

Overview

In this practice, you:

- Determine which package provides the `oracleasm` binary executable
- Configure ASMLib
- Load and initialize the ASMLib driver
- Mark disk partitions for ASM use
- View information about the ASM disk partitions

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Determine the package name that provides the `oracleasm` binary executable.
   a. Use the `which` command to display the location of the `oracleasm` binary.

```
# which oracleasm
/usr/sbin/oracleasm
```

   b. Use the `rpm -qf` command to display the package name that provides the oracleasm binary.

```
# rpm -qf /usr/sbin/oracleasm
oracleasm-support-2.1.8-1.el6.x86_64
```

   - Recall that you installed this optional package during the Oracle Linux installation practice in Lesson 3.

2. Use the `oracleasm configure -i` command to configure ASMLib.

```
# oracleasm configure -i
Configuring the Oracle ASM library driver.

This will configure the on-boot properties of the Oracle ASM
library driver. The following questions will determine whether
the driver is loaded on boot and what permissions it will have.
The current values will be shown in brackets ('[]'). Hitting
<ENTER> without typing an answer will keep that current value.
Ctrl-C will abort.

Default user to own the driver interface []: oracle
Default group to own the driver interface []: dba
Start Oracle ASM library driver on boot (y/n) [n]: y
Scan for Oracle ASM disks on boot (y/n) [y]: ENTER
Writing Oracle ASM library driver configuration: done
```

3. Use the `oracleasm init` command to load and initialize the ASMLib driver.

```
# oracleasm init
Creating /dev/oracleasm mount point: /dev/oracleasm
Loading module "oracleasm": oracleasm
Configuring "oracleasm" to use device physical block size
Mounting ASMlib driver filesystem: /dev/oracleasm
```

4. Use the `oracleasm configure` command without the `-i` option.

```
# oracleasm configure
ORACLEASM_ENABLED=true
ORACLEASM_UID=oracle
ORACLEASM_GID=dba
ORACLEASM_SCANBOOT=true
ORACLEASM_SCANORDER=""
ORACLEASM_SCANEXCLUDE=""
ORACLEASM_USE_LOGICAL_BLOCK_SIZE="false"
```

5. Mark disk partitions for ASM use.
   a. Use the `oracleasm createdisk` command to mark `/dev/xvdb1` for ASM use.

```
# oracleasm createdisk VOL1 /dev/xvdb1
Writing disk header: done
Instantiating disk: done
```

   b. Use the `oracleasm createdisk` command to mark `/dev/xvdd1` for ASM use.

```
# oracleasm createdisk VOL2 /dev/xvdd1
Writing disk header: done
Instantiating disk: done
```

6. Use the `ls` command to display a long list of `/dev/oracleasm/disks` directory.

```
# ls -l /dev/oracleasm/disks
brw-rw----. 1 oracle  dba  ... VOL1
brw-rw----. 1 oracle  dba  ... VOL2
```

7. Use the `oracleasm listdisks` command to list the disk names of marked ASMLib disks.

```
# oracleasm listdisks
VOL1
VOL2
```

8. Use the `oracleasm scandisks` command to enable cluster nodes to identify which shared disks have been marked as ASMLib disks on another node.

```
# oracleasm scandisks
Reloading disk partitions: done
Cleaning any stale ASM disks...
Scanning system for ASM disks...
```

9. Use the `oracleasm querydisk` command to determine whether a disk name or disk device is being used by ASMLib.

```
# oracleasm querydisk VOL1
Disk "VOL1" is a valid ASM disk
# oracleasm querydisk /dev/xvdd1
Device "/dev/xvdd1" is marked an ASM disk with the label "VOL2"
```

Practices for Lesson 19: System Monitoring

Chapter 19

Practices for Lesson 19: System Monitoring

Practices Overview

In these practices, you:

- Use the `sosreport` utility to collect system information
- Use standard Linux utilities to monitor system resource usage
- Use OSWatcher Black Box and OSWatcher Black Box Analyzer

Practice 19-1: Using `sosreport` to Collect System Information

Overview

In this practice, you:

- Use the `sosreport` utility to collect system information
- Extract the compressed TAR file and view the collected information
- View the status of the `sosreport` plug-ins

Assumptions

You are the `root` user on **host03** VM.

Tasks

1. Use the `rpm` command to verify that the `sos` packing is installed.

```
[host03]# rpm -q sos
sos-2.2-47.0.1.el6.noarch
```

- In this example, the package is installed.

2. Run the `sosreport` command.

   a. Press Enter when prompted to continue.

   b. Provide your first initial and last name when prompted.

   c. Enter number `1` as the case number for which you are generating the report.

```
[host03]# sosreport
sosreport (version 2.2)
  This command will collect diagnostic and configuration
information from this Oracle Linux system and installed
applications.
  An archive containing the collected information will be
generated in /tmp and may be provided to a Oracle USA
support representative.
  Any information provided to Oracle USA will be treated
in accordance with the published support policies at:
  https://linux.oracle.com/

  The generated archive may contain data considered
sensitive and its content should be reviewed by the
originating organization before being passed to any third
party.
  No changes will be made to system configuration.
Press ENTER to continue, or CTRL-C to quit. ENTER
Please enter your first initial and last name [host03]: cmac
Please enter the case number that you are generating... 1
```

```
Running plugins. Please wait ...
Completed...
...
Your sosreport has been generated and saved in:
  /tmp/sosreport-cmac.1-...tar.xv
...
```

3. Use the `cd` command to change to the `/tmp` directory.

```
[host03]# cd /tmp
```

4. Use the `ls` command to display a long listing of the `/tmp` directory.

```
[host03]# ls -l
...
-rw------- sosreport-cmac.1-...tar.xz
-rw------- sosreport-cmac.1-...tar.xz.md5
...
```

- Notice the two `sosreport` files, one with the `.xz` extension and one with the `.md5` extension.
- The `.xz` file is the compressed data file.
- Note the size of the `.xz` file.

5. Use the `xz -d` command to uncompress the `.xz` file.

```
[host03]# xz -d sosreport-cmac.1-...tar.xz
```

6. Use the `ls` command to display a long listing of the `/tmp` directory.

```
[host03]# ls -l
...
-rw------- sosreport-cmac.1-...tar
-rw------- sosreport-cmac.1-...tar.xz.md5
...
```

- Notice that the `sosreport` file with the `.tar.xz` extension now has a `.tar` extension.
- Note the size of the `.tar` file—it is considerably larger than the compressed (`.xz`) file.

7. Use the `tar` command to extract the `.tar` file.

```
[host03]# tar xvf sosreport-cmac.1-...tar
host03-.../
host03-.../uname -> sos_commands/kernel/uname_-a
host03-.../date -> sos_commands/general/date
host03-.../java -> sos_commands/general/alternatives_--displa...
host03-.../ifconfig -> sos_commands/networking/ifconfig_-a
...
```

- Notice that the `tar` file is extracted in a `host03-...` directory.

8. Use the cd command to change to the host03-... directory.

```
[host03]# cd host03-*
```

9. Use the ls command to display a long listing of the host03-... directory.

```
[host03]# ls -l
...
drwxr-xr-x          boot
lrwxrwxrwx          chkconfig -> sos_commands/startup/...
lrwxrwxrwx          date -> sos_commands/general/date
...
drwxr-xr-x          etc
lrwxrwxrwx          free -> sos_commands/memory/free
lrwxrwxrwx          hostname -> sos_commands/general/...
...
drwxr-xr-x          lib
...
```

- Notice that a number of directories that contain data collected from the system exist.
- Notice that a number of symbolic links that contain the output of several status-related commands exist.

10. Use the sosreport -l command to list the plug-ins.

```
[host03]# sosreport -l
sosreport (version 2.2)
The following plugins are currently enabled:
 acpid              acpid related information
 anaconda           Anaconda / Installation information
...
The following plugins are currently disabled:
 amd                Amd automounter information
 cloudforms         Cloudforms related information
...
The following plugin options are available:
 apache.log          off gathers all apache logs
 auditd.syslogsize   15 max size (MiB) to collect per syslog...
...
```

Practice 19-2: Using Standard Linux Performance Monitoring Tools

Overview

In this practice, you use standard Linux system resource monitoring utilities to observe the following:

- CPU statistics
- Memory statistics
- Disk I/O statistics
- Network statistics

Assumptions

- You are the `root` user on **dom0**.
- The **host03** VM is running.
- The output shown in the tasks is only a sample. Your output will be different.

Tasks

1. Open two terminal windows on **dom0** and use the `su -` command to become the `root` user (the `root` user's password on **dom0** is `oracle`) in both windows.
   - You have two terminal windows open:
     - One window to run the Linux system resource monitoring utilities
     - One window to run a command-line utility to generate a system load

```
[dom0]$ su –
Password: oracle
[dom0]#
```

2. Use the `ssh` command to log in to **host03**. The `root` password is `oracle`.
   - Log in to **host03** from both terminal windows.

```
[dom0]# ssh host03
root@host03's password: oracle
[root@host03 ~]#
```

3. Observe the CPU statistics.
   a. In one window, use the `top` command to display CPU usage and load averages.
      - The `top` command also monitors process statistics and memory usage.

```
# top
```

   b. In the second window, run the following command to generate a system load:

```
# dd if=/dev/zero of=/dev/null bs=1024
```

A sample output of `top` is shown as follows:

```
root@host03:~
File  Edit  View  Search  Terminal  Help
top - 04:46:26 up 2 days, 12:33,  3 users,  load average: 0.22, 0.06, 0.06
Tasks: 136 total,   2 running, 134 sleeping,   0 stopped,   0 zombie
Cpu(s): 19.7%us, 80.3%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   2050816k total,  1964924k used,    85892k free,    97400k buffers
Swap:  3876860k total,        0k used,  3876860k free,  1514156k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
15792 root      20   0  102m  512  424 R 99.7  0.0   0:15.19 dd
15747 root      20   0 15088 1124  836 R  0.3  0.1   0:00.15 top
    1 root      20   0 19408 1520 1220 S  0.0  0.1   0:00.74 init
    2 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S  0.0  0.0   0:00.15 ksoftirqd/0
    5 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kworker/0:0H
    7 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kworker/u:0H
    8 root      RT   0     0    0    0 S  0.0  0.0   0:00.00 migration/0
    9 root      20   0     0    0    0 S  0.0  0.0   0:00.00 rcu_bh
   10 root      20   0     0    0    0 S  0.0  0.0   0:03.88 rcu_sched
   11 root      RT   0     0    0    0 S  0.0  0.0   0:02.53 watchdog/0
   12 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 cpuset
   13 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 khelper
   14 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kdevtmpfs
   15 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 netns
   16 root      20   0     0    0    0 S  0.0  0.0   0:00.00 xenwatch
   17 root      20   0     0    0    0 S  0.0  0.0   0:00.01 xenbus
```

- In this example, CPU usage is high as indicated by the `0.0%id` (idle) statistic.
- The biggest consumer of the CPU is the `dd` process, which has a PID of `15792`.
- Load average is the average number of processes in the run queue or the number of processes waiting to run on the CPU.
  - The load average over the last 1 minute is `0.22`, over the last 5 minutes is `0.06`, and over the last 15 minutes is `0.06`.
  - A high load average is an indication that your system does not have sufficient CPU capacity.
- Press `q` to quit the `top` command.

c. Use the `mpstat -P ALL 1` command to display CPU statistics.

- The `1` argument provides statistics at one-second intervals.
- The `-P ALL` option provides statistics for each individual processor and globally for all processors.

```
[root@host03 ~]# mpstat -P ALL 1
Linux 3.8.13-16.2.1.el6uek.x86_64 (host03.example.com)  12/21/2013      _x86_64_        (1 CPU)

04:49:51 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest   %idle
04:49:52 AM  all   22.77    0.00   77.23    0.00    0.00    0.00    0.00    0.00    0.00
04:49:52 AM    0   22.77    0.00   77.23    0.00    0.00    0.00    0.00    0.00    0.00

04:49:52 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest   %idle
04:49:53 AM  all   22.22    0.00   77.78    0.00    0.00    0.00    0.00    0.00    0.00
04:49:53 AM    0   22.22    0.00   77.78    0.00    0.00    0.00    0.00    0.00    0.00

04:49:53 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest   %idle
04:49:54 AM  all   19.80    0.00   80.20    0.00    0.00    0.00    0.00    0.00    0.00
04:49:54 AM    0   19.80    0.00   80.20    0.00    0.00    0.00    0.00    0.00    0.00

04:49:54 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest   %idle
04:49:55 AM  all   20.20    0.00   79.80    0.00    0.00    0.00    0.00    0.00    0.00
04:49:55 AM    0   20.20    0.00   79.80    0.00    0.00    0.00    0.00    0.00    0.00

04:49:55 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest   %idle
04:49:56 AM  all   20.00    0.00   80.00    0.00    0.00    0.00    0.00    0.00    0.00
04:49:56 AM    0   20.00    0.00   80.00    0.00    0.00    0.00    0.00    0.00    0.00

04:49:56 AM  CPU    %usr   %nice    %sys %iowait    %irq   %soft  %steal  %guest   %idle
```

- In this example, CPU usage is high as indicated by the `0.00` (`%idle`) statistic.
- Press Ctrl + C after viewing a few intervals.

d. Use the `sar -u` command to display system-wide CPU usage. Only a partial output is shown as follows:

```
root@host03:~
File  Edit  View  Search  Terminal  Help
[root@host03 ~]# sar -u
Linux 3.8.13-16.2.1.el6uek.x86_64 (host03.example.com)  12/21/2013      _x86_64_        (1 CPU)

12:00:01 AM     CPU     %user     %nice   %system   %iowait    %steal     %idle
12:10:01 AM     all      0.01      0.00      0.03      0.01      0.00     99.96
12:20:01 AM     all      0.04      0.00      0.03      0.01      0.00     99.92
12:30:01 AM     all      0.01      0.00      0.02      0.01      0.00     99.97
12:40:01 AM     all      0.01      0.00      0.01      0.00      0.00     99.98
12:50:01 AM     all      0.01      0.00      0.02      0.00      0.00     99.98
01:00:01 AM     all      0.01      0.00      0.02      0.00      0.00     99.97
01:10:01 AM     all      0.01      0.00      0.02      0.00      0.00     99.96
01:20:01 AM     all      0.05      0.00      0.03      0.01      0.00     99.91
01:30:01 AM     all      0.01      0.00      0.01      0.00      0.00     99.98
01:40:01 AM     all      0.01      0.00      0.01      0.00      0.00     99.98
01:50:01 AM     all      0.01      0.00      0.01      0.01      0.00     99.97
02:00:01 AM     all      0.01      0.00      0.01      0.00      0.00     99.98
02:10:01 AM     all      0.01      0.00      0.02      0.01      0.00     99.97
02:20:01 AM     all      0.04      0.00      0.03      0.02      0.00     99.90
02:30:01 AM     all      0.01      0.00      0.02      0.00      0.00     99.98
02:40:01 AM     all      0.01      0.00      0.02      0.01      0.00     99.97
02:50:01 AM     all      0.01      0.00      0.02      0.00      0.00     99.98
03:00:01 AM     all      0.01      0.00      0.02      0.01      0.00     99.97
03:10:01 AM     all      0.01      0.00      0.02      0.00      0.00     99.97
03:20:01 AM     all      0.04      0.00      0.04      0.01      0.00     99.91
```

- In this example, CPU usage is low, with a few exceptions, as indicated by the `99.xx` (`%idle`) statistics.
- An “Average” line is not shown in this example.

e. Use the `sar -q` command to display run queue length and load averages. Only a partial output is shown as follows:

```
root@host03:~
File  Edit  View  Search  Terminal  Help
[root@host03 ~]# sar -q
Linux 3.8.13-16.2.1.el6uek.x86_64 (host03.example.com)  12/21/2013      _x86_64
_       (1 CPU)

12:00:01 AM   runq-sz  plist-sz   ldavg-1   ldavg-5  ldavg-15
12:10:01 AM         0       223      0.00      0.02      0.05
12:20:01 AM         0       223      0.00      0.01      0.05
12:30:01 AM         0       223      0.00      0.01      0.05
12:40:01 AM         0       223      0.00      0.01      0.05
12:50:01 AM         0       223      0.00      0.01      0.05
01:00:01 AM         0       223      0.00      0.01      0.05
01:10:01 AM         0       223      0.00      0.01      0.05
01:20:01 AM         0       223      0.00      0.01      0.05
01:30:01 AM         0       223      0.00      0.01      0.05
01:40:01 AM         0       223      0.00      0.01      0.05
01:50:01 AM         0       223      0.00      0.01      0.05
02:00:01 AM         0       223      0.00      0.01      0.05
02:10:01 AM         0       223      0.00      0.01      0.05
02:20:01 AM         0       223      0.00      0.01      0.05
02:30:01 AM         0       223      0.00      0.01      0.05
02:40:01 AM         0       223      0.00      0.01      0.05
02:50:01 AM         0       223      0.00      0.01      0.05
03:00:01 AM         0       223      0.00      0.01      0.05
03:10:01 AM         0       224      0.00      0.01      0.05
```

- In this example, the CPU is not saturated as indicated by the low run queue length (`runq-sz`) and low load averages (last 1 minute: `ldavg-1`, last 5 minutes: `ldavg-5`, and last 15 minutes: `ldavg-15`).
- A run queue size greater than the number of CPUs on your system is usually indicative of a CPU bottleneck.
- An "Average" line is not shown in this example.

f. Use the `vmstat 1` command to display CPU statistics.

- This command is primarily used for monitoring virtual memory statistics.

```
[root@host03 ~]# vmstat 1
procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 1  0      0  85248  97820 1514300    0    0     6     7   36   15  0  1 99  0  0
 1  0      0  85232  97820 1514304    0    0     0     0 1055   49 18 82  0  0  0
 1  0      0  85240  97820 1514304    0    0     0     0 1058   16 19 81  0  0  0
 1  0      0  85240  97820 1514304    0    0     0     0 1054   34 20 80  0  0  0
 1  0      0  85240  97820 1514304    0    0     0     0 1061   24 20 80  0  0  0
 1  0      0  85240  97820 1514304    0    0     0     0 1061   27 20 80  0  0  0
 1  0      0  85272  97824 1514304    0    0     0    16 1045   30 21 79  0  0  0
 1  0      0  85272  97824 1514304    0    0     0     0 1031   28 19 81  0  0  0
 1  0      0  85272  97824 1514304    0    0     0     0 1032   18 19 81  0  0  0
 1  0      0  85272  97824 1514304    0    0     0     0 1052   50 21 79  0  0  0
 1  0      0  85264  97824 1514304    0    0     0     0 1066   16 19 81  0  0  0
 1  0      0  85264  97824 1514304    0    0     0     0 1029   28 17 83  0  0  0
 1  0      0  85264  97832 1514304    0    0     0    16 1037   20 18 82  0  0  0
 1  0      0  85264  97832 1514304    0    0     0     0 1034   33 18 82  0  0  0
 1  0      0  85264  97832 1514304    0    0     0     0 1045   22 20 80  0  0  0
 1  0      0  85264  97832 1514304    0    0     0     0 1029   28 20 80  0  0  0
 1  0      0  85264  97832 1514304    0    0     0     0 1051   16 20 80  0  0  0
 1  0      0  85264  97832 1514304    0    0     0     0 1044   28 20 80  0  0  0
 1  0      0  85272  97840 1514304    0    0     0    16 1080   24 16 84  0  0  0
 1  0      0  85272  97840 1514304    0    0     0     0 1031   36 19 81  0  0  0
 1  0      0  85272  97840 1514304    0    0     0     0 1038   18 19 81  0  0  0
```

- The `1` argument provides statistics at one-second intervals.
- In this example, CPU usage is high as indicated by the low `id` (`%idle`) statistics.
- In this example, the high run queue length (`r`) statistics are also an indication of CPU saturation.
- Press Ctrl + C after viewing a few intervals.

g. Use the `iostat` command to display CPU usage.

- This command is primarily used for monitoring system I/O device loads.

```
[root@host03 ~]# iostat
Linux 3.8.13-16.2.1.el6uek.x86_64 (host03.example.com)  12/21/2013      _x86_64_       (
1 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.27    0.02    0.80    0.11    0.00   98.80

Device:            tps   Blk_read/s   Blk_wrtn/s   Blk_read   Blk_wrtn
xvda              0.70        12.07        13.62    2651004    2993354
xvdb              0.00         0.02         0.00       3800          0
xvdd              0.00         0.02         0.00       3800          0
scd0              0.00         0.00         0.00        532          0

[root@host03 ~]#
```

- In this example, the average `%idle` statistic for the CPU is 98.80%.

4. Observe the memory statistics.

a. Use the `top` command to display memory usage.

```
# top
```

A sample output of `top` is shown as follows:

```
root@host03:~
File  Edit  View  Search  Terminal  Help
top - 05:20:03 up 2 days, 13:07,  3 users,  load average: 1.00, 0.99, 0.89
Tasks: 136 total,   2 running, 134 sleeping,   0 stopped,   0 zombie
Cpu(s): 19.5%us, 80.5%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   2050816k total,  1964924k used,    85892k free,    97904k buffers
Swap:  3876860k total,        0k used,  3876860k free,  1514284k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
15792 root      20   0  102m  512  424 R 98.7  0.0  33:45.91 dd
10127 oracle    20   0  401m  11m 8984 S  0.3  0.6   0:01.39 metacity
    1 root      20   0 19408 1520 1220 S  0.0  0.1   0:00.75 init
    2 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S  0.0  0.0   0:00.15 ksoftirqd/0
    5 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kworker/0:0H
    7 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kworker/u:0H
    8 root      RT   0     0    0    0 S  0.0  0.0   0:00.00 migration/0
    9 root      20   0     0    0    0 S  0.0  0.0   0:00.00 rcu_bh
   10 root      20   0     0    0    0 S  0.0  0.0   0:03.94 rcu_sched
   11 root      RT   0     0    0    0 S  0.0  0.0   0:02.54 watchdog/0
   12 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 cpuset
   13 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 khelper
   14 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kdevtmpfs
   15 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 netns
   16 root      20   0     0    0    0 S  0.0  0.0   0:00.00 xenwatch
   17 root      20   0     0    0    0 S  0.0  0.0   0:00.01 xenbus
```

- The upper section of `top` displays memory statistics:
  - The `Mem:` line reflects how much physical memory your system has, how much is used, and how much is free.
  - The `Swap:` line reflects how much swap memory your system has, how much is used, and how much is free.
  - When a computer runs out of physical memory and starts using swap space, its performance deteriorates dramatically.
  - If you run out of swap, you will most likely crash your programs or the OS.
- The lower section of `top` displays a list of processes sorted by CPU usage, with the top consumer of CPU listed first.

b. Sort by memory usage, press `F` or `O` to change the sort field, and then select either of the following to sort processes by memory usage:

    - `n`: %MEM = Memory usage (RES)
    - `q`: RES = Resident size (kb)

- Press the Enter key after selecting a sort field.

```
root@host03:~
File  Edit  View  Search  Terminal  Help
top - 05:21:16 up 2 days, 13:08,  3 users,  load average: 1.00, 0.99, 0.90
Tasks: 136 total,   2 running, 134 sleeping,   0 stopped,   0 zombie
Cpu(s): 17.7%us, 82.3%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   2050816k total,  1964932k used,    85884k free,    97920k buffers
Swap:  3876860k total,        0k used,  3876860k free,  1514284k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 1769 root      20   0  158m  24m 9060 S  0.3  1.2   0:25.32 Xorg
10135 oracle    20   0  521m  20m  13m S  0.0  1.0   0:36.60 nautilus
10167 oracle    20   0  316m  18m 9236 S  0.0  0.9   0:00.16 python
10300 oracle    20   0  393m  17m  13m S  0.0  0.9   0:00.08 gnote
10164 oracle    20   0  472m  14m  10m S  0.0  0.7   0:00.54 gpk-update-icon
10132 oracle    20   0  321m  13m  10m S  0.0  0.7   0:02.65 gnome-panel
11314 oracle    20   0  287m  13m 9560 S  0.0  0.7   0:03.83 gnome-terminal
10301 oracle    20   0  452m  12m  10m S  0.0  0.6   0:01.24 clock-applet
10143 oracle    20   0  310m  11m 8348 S  0.0  0.6   0:00.16 nm-applet
10160 oracle    20   0  322m  11m 8976 S  0.0  0.6   0:02.91 wnck-applet
10127 oracle    20   0  401m  11m 8984 S  0.0  0.6   0:01.39 metacity
10299 oracle    20   0  382m  11m 8768 S  0.0  0.6   0:00.07 gdm-user-switch
10165 oracle    20   0  360m  10m 8252 S  0.0  0.5   0:00.23 gnome-volume-co
10102 oracle    20   0  488m  10m 7868 S  0.0  0.5   0:04.47 gnome-settings-
10159 oracle    20   0  308m 9304 7080 S  0.0  0.5   0:00.04 trashapplet
10213 oracle    20   0  260m 8500 4464 S  0.0  0.4   0:03.26 gnome-screensav
10163 oracle    20   0  255m 8092 6036 S  0.0  0.4   0:00.06 bluetooth-apple
```

- Press `q` to quit the `top` command.

c. Use the `vmstat` command to display memory statistics.

```
root@host03:~
File Edit View Search Terminal Help
[root@host03 ~]# vmstat 1
procs -----------memory---------- ---swap-- -----io---- --system-- -----cpu-----
 r  b   swpd   free   buff  cache   si   so    bi    bo   in   cs us sy id wa st
 1  0      0  86220  97924 1514292    0    0     6     7   38   15  0  1 99  0  0
 1  0      0  86256  97924 1514292    0    0     0     0 1056   48 19 81  0  0  0
 1  0      0  86256  97924 1514292    0    0     0     0 1046   24 20 80  0  0  0
 1  0      0  86256  97928 1514288    0    0     0    12 1065   58 18 82  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1050   34 20 80  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1083   54 20 80  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1040   22 19 81  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1060   29 19 81  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1044   23 22 78  0  0  0
 1  0      0  86256  97928 1514292    0    0     0    16 1041   34 18 82  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1076   63 19 81  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1072   68 22 78  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1045   29 18 82  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1048   28 18 82  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1039   33 18 82  0  0  0
 1  0      0  86256  97928 1514292    0    0     0    20 1031   32 19 81  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1048   23 19 81  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1046   26 19 81  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1046   22 17 83  0  0  0
 1  0      0  86256  97928 1514292    0    0     0     0 1050   59 20 80  0  0  0
^C
```

- The `1` argument provides statistics at one-second intervals.
- The important memory statistics are:
  - `swpd` – The amount of virtual memory used
  - `free` – The amount of idle memory
  - `si` – The amount of memory swapped in from disk (per second)
  - `so` – The amount of memory swapped out to disk (per second)
- In this example, the system has sufficient free memory and is not swapping.
- Press Ctrl + C after viewing a few intervals.

d. Use the `free` command to display memory statistics.

```
root@host03:~
File Edit View Search Terminal Help
[root@host03 ~]# free -m
             total       used       free     shared    buffers     cached
Mem:          2002       1923         79          0         95       1478
-/+ buffers/cache:        348       1654
Swap:         3785          0       3785
[root@host03 ~]#
```

- This example uses the `-m` option to display amounts in megabytes.
- This command displays the total amount of free and used physical and swap memory in your system.

e. Use the `sar -r` command to display memory usage statistics. Only a partial output is shown as follows:

```
[root@host03 ~]# sar -r
Linux 3.8.13-16.2.1.el6uek.x86_64 (host03.example.com)  12/21/2013      _x86_64_      (
1 CPU)

12:00:01 AM kbmemfree kbmemused  %memused kbbuffers  kbcached  kbcommit   %commit
12:10:01 AM     58468   1992348     97.15     89480   1554664    501988      8.47
12:20:01 AM     58344   1992472     97.16     89604   1554672    501988      8.47
12:30:01 AM     58352   1992464     97.15     89680   1554672    501988      8.47
12:40:01 AM     58196   1992620     97.16     89756   1554676    501988      8.47
12:50:01 AM     58064   1992752     97.17     89836   1554680    501988      8.47
01:00:01 AM     58072   1992744     97.17     89920   1554676    501988      8.47
01:10:01 AM     57940   1992876     97.17     90016   1554688    501988      8.47
01:20:01 AM     57692   1993124     97.19     90148   1554692    501988      8.47
01:30:01 AM     57724   1993092     97.19     90220   1554696    501988      8.47
01:40:01 AM     57716   1993100     97.19     90308   1554704    501988      8.47
01:50:01 AM     57608   1993208     97.19     90372   1554704    501988      8.47
02:00:01 AM     57444   1993372     97.20     90460   1554708    501988      8.47
02:10:01 AM     57320   1993496     97.21     90544   1554712    501988      8.47
02:20:01 AM     57104   1993712     97.22     90672   1554716    501988      8.47
02:30:01 AM     57196   1993620     97.21     90764   1554724    501988      8.47
02:40:01 AM     57080   1993736     97.22     90848   1554724    501988      8.47
02:50:01 AM     56948   1993868     97.22     90928   1554728    501988      8.47
03:00:01 AM     56856   1993960     97.23     91004   1554728    501988      8.47
03:10:01 AM     56460   1994356     97.25     91120   1554736    502612      8.48
03:20:01 AM     55380   1995436     97.30     91236   1554732    503944      8.50
03:30:01 AM     55252   1995564     97.31     91312   1554744    501988      8.47
03:40:01 AM     55112   1995704     97.31     91408   1554744    501988      8.47
```

- Observe the `%memused` value, which is the percentage of used memory.
- An "Average" line is not shown in this example.

f. Use the `sar -B` command to display memory paging statistics. Only a partial output is shown as follows:

```
root@host03:~
File  Edit  View  Search  Terminal  Help
[root@host03 ~]# sar -B
Linux 3.8.13-16.2.1.el6uek.x86_64 (host03.example.com)  12/21/2013      _x86_64_        (1 CPU)

12:00:01 AM  pgpgin/s pgpgout/s   fault/s  majflt/s  pgfree/s pgscank/s pgscand/s pgsteal/s    %vmeff
12:10:01 AM      0.01      1.01     11.97      0.00      6.42      0.00      0.00      0.00      0.00
12:20:01 AM      0.00      1.82     20.39      0.00     15.21      0.00      0.00      0.00      0.00
12:30:01 AM      0.00      0.74      3.02      0.00      3.73      0.00      0.00      0.00      0.00
12:40:01 AM      0.00      0.77      3.03      0.00      3.74      0.00      0.00      0.00      0.00
12:50:01 AM      0.00      0.80      3.02      0.00      3.76      0.00      0.00      0.00      0.00
01:00:01 AM      0.00      0.80      4.74      0.00      4.31      0.00      0.00      0.00      0.00
01:10:01 AM      0.00      0.86     11.95      0.00      6.43      0.00      0.00      0.00      0.00
01:20:01 AM      0.00      1.88     20.39      0.00     15.21      0.00      0.00      0.00      0.00
01:30:01 AM      0.00      0.73      3.02      0.00      3.73      0.00      0.00      0.00      0.00
01:40:01 AM      0.00      0.83      3.03      0.00      3.76      0.00      0.00      0.00      0.00
01:50:01 AM      0.00      0.74      3.03      0.00      3.75      0.00      0.00      0.00      0.00
02:00:01 AM      0.00      0.80      3.03      0.00      3.71      0.00      0.00      0.00      0.00
02:10:01 AM      0.00      0.85     11.94      0.00      6.42      0.00      0.00      0.00      0.00
02:20:01 AM      0.00      1.84     20.56      0.00     15.23      0.00      0.00      0.00      0.00
02:30:01 AM      0.00      0.83      3.03      0.00      3.73      0.00      0.00      0.00      0.00
02:40:01 AM      0.00      0.74      4.75      0.00      4.32      0.00      0.00      0.00      0.00
02:50:01 AM      0.00      0.83      3.02      0.00      3.76      0.00      0.00      0.00      0.00
03:00:01 AM      0.00      0.75      3.02      0.00      3.73      0.00      0.00      0.00      0.00
03:10:01 AM      0.00      0.91     11.86      0.00      6.29      0.00      0.00      0.00      0.00
03:20:01 AM      0.01      3.60     24.41      0.00     16.01      0.00      0.00      0.00      0.00
03:30:01 AM      0.00     10.16    147.15      0.00     58.27      0.00      0.00      0.00      0.00
03:40:01 AM      0.00      0.75      3.03      0.00      3.78      0.00      0.00      0.00      0.00
03:50:01 AM      0.00      0.80      3.03      0.00      3.77      0.00      0.00      0.00      0.00
```

- Observe `pgscank`, which is the number of pages scanned by the `kswapd` daemon per second, and `pgscand`, which is the number of pages scanned directly per second.
- An “Average” line is not shown in this example.

g. Use the `sar -W` command to display memory swapping statistics. Only a partial output is shown as follows:

```
[root@host03 ~]# sar -W
Linux 3.8.13-16.2.1.el6uek.x86_64 (host03.example.com)  12/21/2013      _x86_6

12:00:01 AM  pswpin/s pswpout/s
12:10:01 AM      0.00      0.00
12:20:01 AM      0.00      0.00
12:30:01 AM      0.00      0.00
12:40:01 AM      0.00      0.00
12:50:01 AM      0.00      0.00
01:00:01 AM      0.00      0.00
01:10:01 AM      0.00      0.00
01:20:01 AM      0.00      0.00
01:30:01 AM      0.00      0.00
01:40:01 AM      0.00      0.00
01:50:01 AM      0.00      0.00
02:00:01 AM      0.00      0.00
02:10:01 AM      0.00      0.00
02:20:01 AM      0.00      0.00
02:30:01 AM      0.00      0.00
02:40:01 AM      0.00      0.00
02:50:01 AM      0.00      0.00
03:00:01 AM      0.00      0.00
03:10:01 AM      0.00      0.00
03:20:01 AM      0.00      0.00
03:30:01 AM      0.00      0.00
03:40:01 AM      0.00      0.00
03:50:01 AM      0.00      0.00
```

- The `pswpin/s` value is the number of swap pages the system swapped in per second, and `pswpout/s` is the number of swap pages the system swapped out per second.
- An “Average” line is not shown in this example.

h. Use the `cat` command to view the contents of `/proc/meminfo`.

```
[root@host03 ~]# cat /proc/meminfo
MemTotal:        2050816 kB
MemFree:           85756 kB
Buffers:           98244 kB
Cached:          1514344 kB
SwapCached:            0 kB
Active:           621588 kB
Inactive:        1121660 kB
Active(anon):      30036 kB
Inactive(anon):   104536 kB
Active(file):     591552 kB
Inactive(file):  1017124 kB
Unevictable:           0 kB
Mlocked:               0 kB
SwapTotal:       3876860 kB
SwapFree:        3876860 kB
Dirty:                12 kB
Writeback:             0 kB
AnonPages:        130684 kB
Mapped:            60520 kB
Shmem:              3908 kB
Slab:             168484 kB
SReclaimable:     114464 kB
SUnreclaim:        54020 kB
KernelStack:        1800 kB
PageTables:        24048 kB
NFS_Unstable:          0 kB
```

5. Observe the disk I/O statistics.

a. Use the `iostat –xz` command to display I/O statistics for devices.

```
[root@host03 ~]# iostat -xz
Linux 3.8.13-16.2.1.el6uek.x86_64 (host03.example.com)  12/21/2013      _x86_64_        (1 CPU)

avg-cpu:  %user   %nice %system %iowait  %steal   %idle
           0.41    0.02    1.35    0.11    0.00   98.11

Device:         rrqm/s   wrqm/s     r/s     w/s   rsec/s   wsec/s avgrq-sz avgqu-sz   await  svctm  %util
xvda              0.00     0.22    0.39    0.31    11.98    13.56    36.46     0.03   43.50   1.76   0.12
xvdb              0.00     0.00    0.00    0.00     0.02     0.00     8.39     0.00    1.91   1.79   0.00
xvdd              0.00     0.00    0.00    0.00     0.02     0.00     8.39     0.00    2.59   2.43   0.00
scd0              0.00     0.00    0.00    0.00     0.00     0.00     6.91     0.00    4.06   4.06   0.00

[root@host03 ~]#
```

- Observe the `%util` value, which is the percentage of CPU time during which I/O requests were issued to the device.
- Device saturation occurs when this value is close to `100%`.

- Observe the `avgqu-sz` value, which is the average queue length of the requests that were issued to the device.
- If average queue length is greater than `1`, it is an indication of device I/O saturation.

b. Use the `sar -d` command to display I/O statistics for devices. Only a partial output is shown as follows:

```
root@host03:~
File  Edit  View  Search  Terminal  Help
[root@host03 ~]# sar -d
Linux 3.8.13-16.2.1.el6uek.x86_64 (host03.example.com)  12/21/2013      _x86_64_      (1 CPU)

12:00:01 AM       DEV       tps  rd_sec/s  wr_sec/s  avgrq-sz  avgqu-sz     await     svctm     %util
12:10:01 AM  dev202-0      0.15      0.03      2.03     14.00      0.00      1.11      0.50      0.01
12:10:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:10:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:10:01 AM   dev11-0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:20:01 AM  dev202-0      0.23      0.00      3.64     15.94      0.00      1.76      0.36      0.01
12:20:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:20:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:20:01 AM   dev11-0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00

12:20:01 AM       DEV       tps  rd_sec/s  wr_sec/s  avgrq-sz  avgqu-sz     await     svctm     %util
12:30:01 AM  dev202-0      0.11      0.00      1.48     13.66      0.00      3.51      1.57      0.02
12:30:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:30:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:30:01 AM   dev11-0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:40:01 AM  dev202-0      0.11      0.00      1.55     13.65      0.00      0.40      0.19      0.00
12:40:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:40:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:40:01 AM   dev11-0      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00

12:40:01 AM       DEV       tps  rd_sec/s  wr_sec/s  avgrq-sz  avgqu-sz     await     svctm     %util
12:50:01 AM  dev202-0      0.12      0.00      1.60     13.91      0.00      0.45      0.23      0.00
12:50:01 AM dev202-16      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
12:50:01 AM dev202-48      0.00      0.00      0.00      0.00      0.00      0.00      0.00      0.00
```

- This command also provides `%util` and `avgqu-sz` statistics.
- An “Average” line is not shown in this example.

6. Observe network statistics.

   a. Use the `ip -s link` command to observe network statistics.

```
root@host03:~
File  Edit  View  Search  Terminal  Help
[root@host03 ~]# ip -s link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    RX: bytes  packets  errors  dropped overrun mcast
    6556       78       0       0       0       0
    TX: bytes  packets  errors  dropped carrier collsns
    6556       78       0       0       0       0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:16:3e:00:01:03 brd ff:ff:ff:ff:ff:ff
    RX: bytes  packets  errors  dropped overrun mcast
    84986092   186190   0       6       0       0
    TX: bytes  packets  errors  dropped carrier collsns
    6793583    51719    0       0       0       0
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:16:3e:00:02:03 brd ff:ff:ff:ff:ff:ff
    RX: bytes  packets  errors  dropped overrun mcast
    4214922    110919   0       0       0       0
    TX: bytes  packets  errors  dropped carrier collsns
    258        3        0       0       0       0
[root@host03 ~]#
```

- The number of bytes received and bytes transmitted on each interface is provided by `RX: bytes` and `TX: bytes` values.
- This command also provides the number of packets transmitted and received, the number of errors, dropped packets, overruns, and collisions.
- Frames are dropped and the overrun counter is incremented when the capacity of the interface is exceeded.

b. Use the `ifconfig` command to observe network statistics. Only a partial output is shown as follows:

```
root@host03:~
File  Edit  View  Search  Terminal  Help
[root@host03 ~]# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:16:3E:00:01:03
          inet addr:192.0.2.103  Bcast:192.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::216:3eff:fe00:103/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:186242 errors:0 dropped:6 overruns:0 frame:0
          TX packets:51719 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:84988068 (81.0 MiB)  TX bytes:6793583 (6.4 MiB)

eth1      Link encap:Ethernet  HWaddr 00:16:3E:00:02:03
          inet6 addr: fe80::216:3eff:fe00:203/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:110971 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4216898 (4.0 MiB)  TX bytes:258 (258.0 b)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:78 errors:0 dropped:0 overruns:0 frame:0
          TX packets:78 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:6556 (6.4 KiB)  TX bytes:6556 (6.4 KiB)
```

- This command provides similar output as the `ip -s link` command.

c. Use the `netstat -s` command to observe network statistics. Only a partial output is shown as follows:

- If the `ip` and `ifconfig` commands show an excessive amount of errors, more information can be found by examining the `netstat -s` output.

```
[root@host03 ~]# netstat -s
Ip:
    50309 total packets received
    0 forwarded
    0 incoming packets discarded
    50309 incoming packets delivered
    51533 requests sent out
Icmp:
    31 ICMP messages received
    0 input ICMP message failed.
    ICMP input histogram:
        destination unreachable: 6
        echo requests: 25
    264 ICMP messages sent
    0 ICMP messages failed
    ICMP output histogram:
        destination unreachable: 239
        echo replies: 25
IcmpMsg:
        InType3: 6
        InType8: 25
        OutType0: 25
        OutType3: 239
Tcp:
    23 active connections openings
    10 passive connection openings
    9 failed connection attempts
```

- This command displays summary statistics for each protocol.
- Observe the number of segments retransmitted as an indicator of network interface saturation.
- Many performance problems associated with the network involve retransmission of TCP packets.

d. Use the `netstat -i` command to observe a table listing of network interfaces.

```
[root@host03 ~]# netstat -i
Kernel Interface table
Iface       MTU Met    RX-OK RX-ERR RX-DRP RX-OVR    TX-OK TX-ERR TX-DRP TX-OVR Flg
eth0       1500   0   186341      0      6      0    51719      0      0      0 BMRU
eth1       1500   0   111070      0      0      0        3      0      0      0 BMRU
lo        65536   0       78      0      0      0       78      0      0      0 LRU
[root@host03 ~]#
```

- Observe the `RX-ERR` and `TX-ERR` values for any receive and transmit errors.

7. Use the System Monitor GUI to display system resource usage.
   - This application requires that you access the GNOME desktop.

a. Use the `exit` command to close the `ssh` connection.

```
# exit
logout
Connection to host03 closed.
```

b. From **dom0**, run the `xm list -l host03 | grep location` command to determine the VNC port number for **host03**.

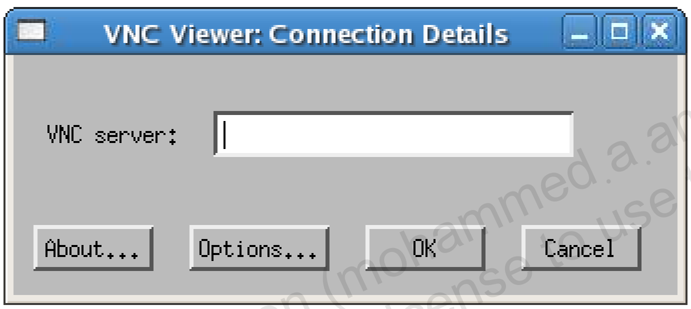
```
[dom0]# xm list -l host03 | grep location
                  (location 0.0.0.0:5900)
                  (location 2)
```

- In this example, the VNC port number is `5900`. This may not be true in your case.

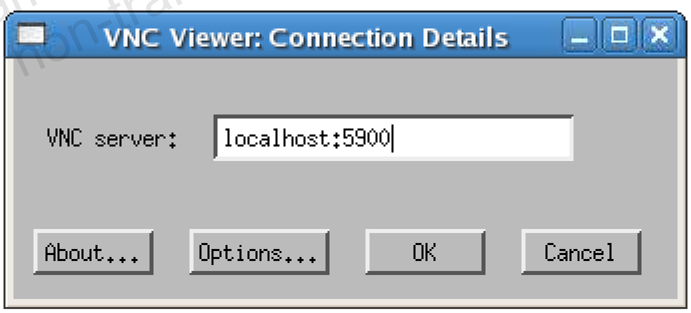
c. Run the `vncviewer&` command:

```
[dom0]# vncviewer&
```

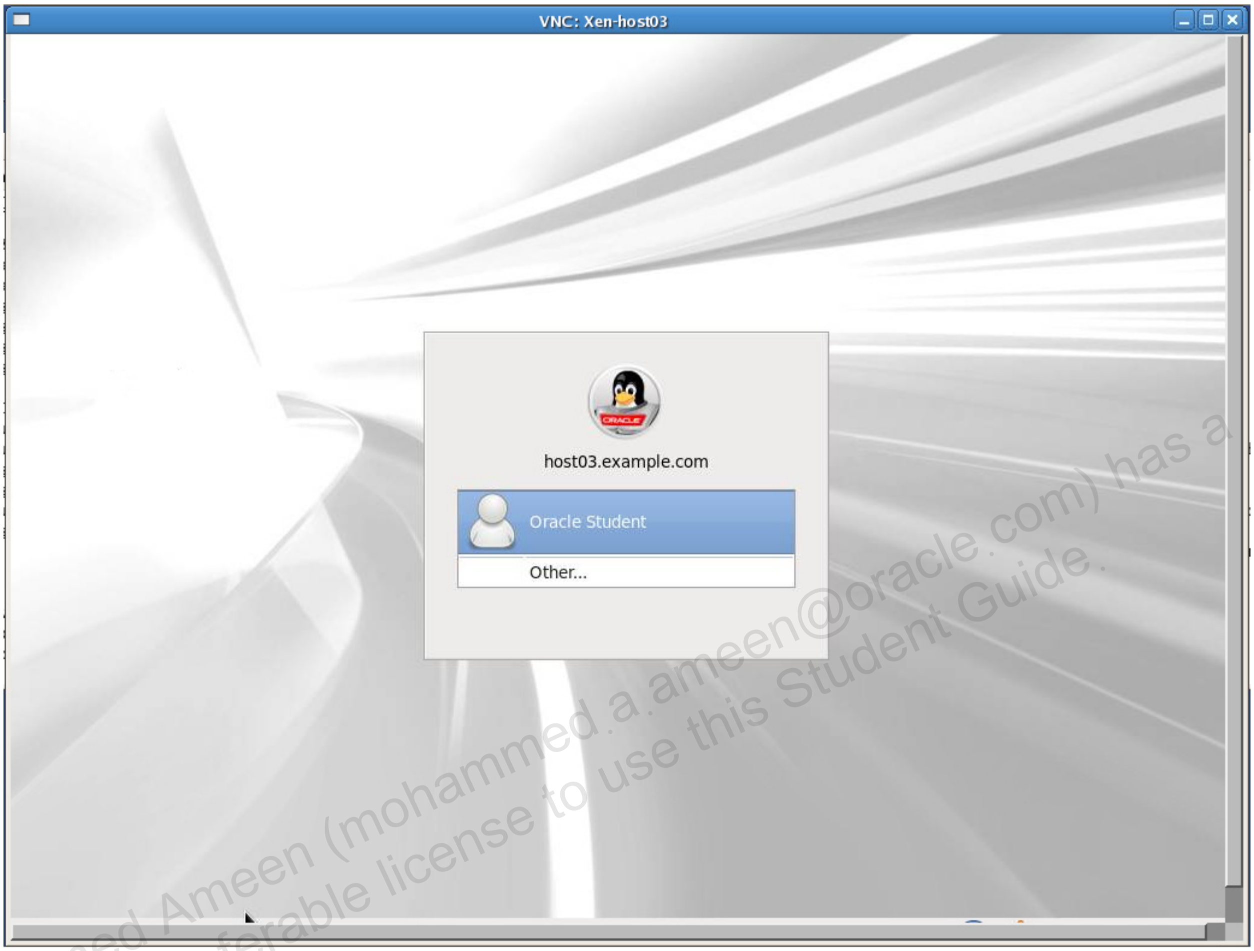
- The **VNC Viewer: Connection Details** dialog box is displayed as shown in the following screenshot:



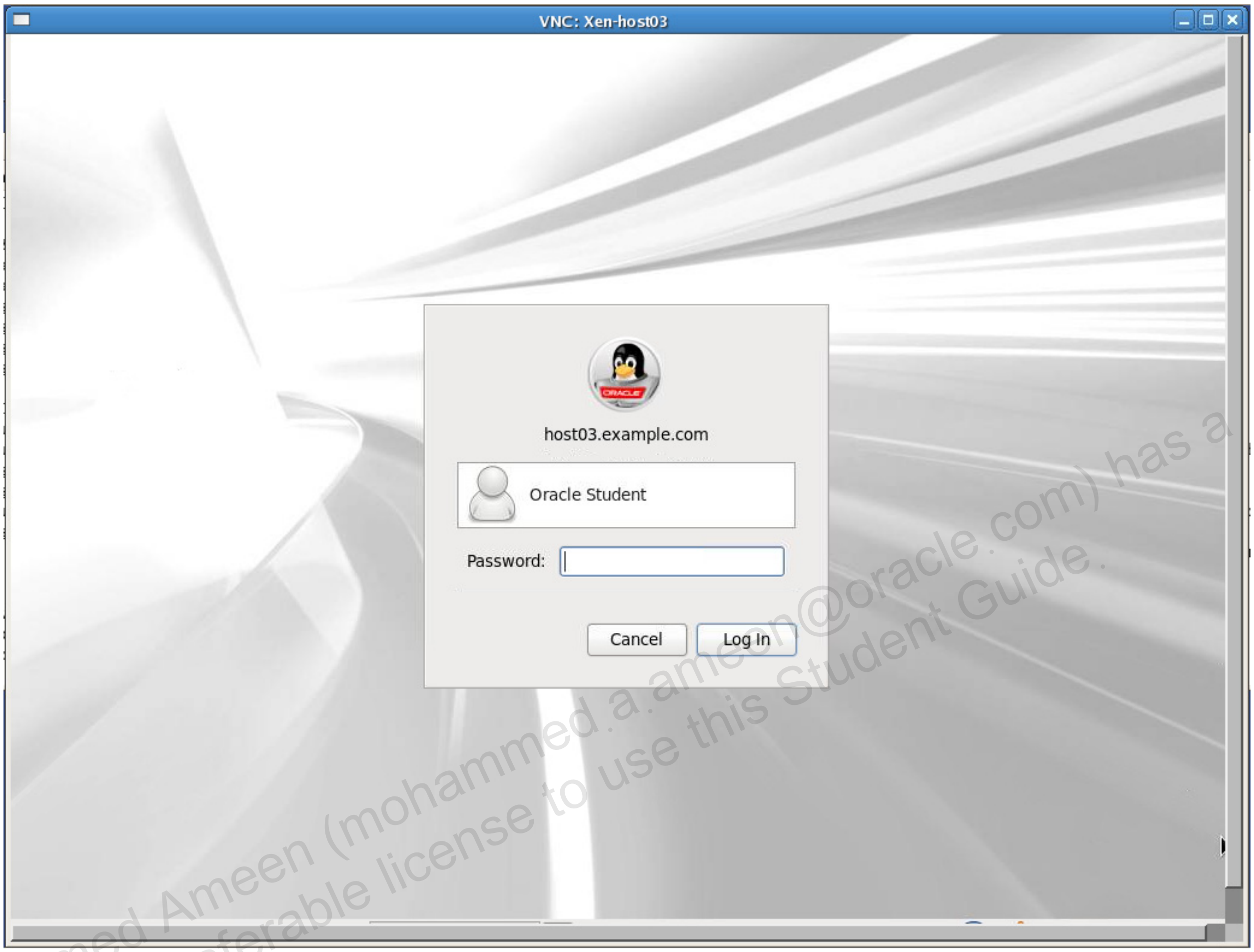
d. Enter the `localhost:<port_number>` command, substituting the VNC port number displayed from the previous `xm list` command. For example, if the port number is `5900`, enter `localhost:5900` as shown and click **OK**.



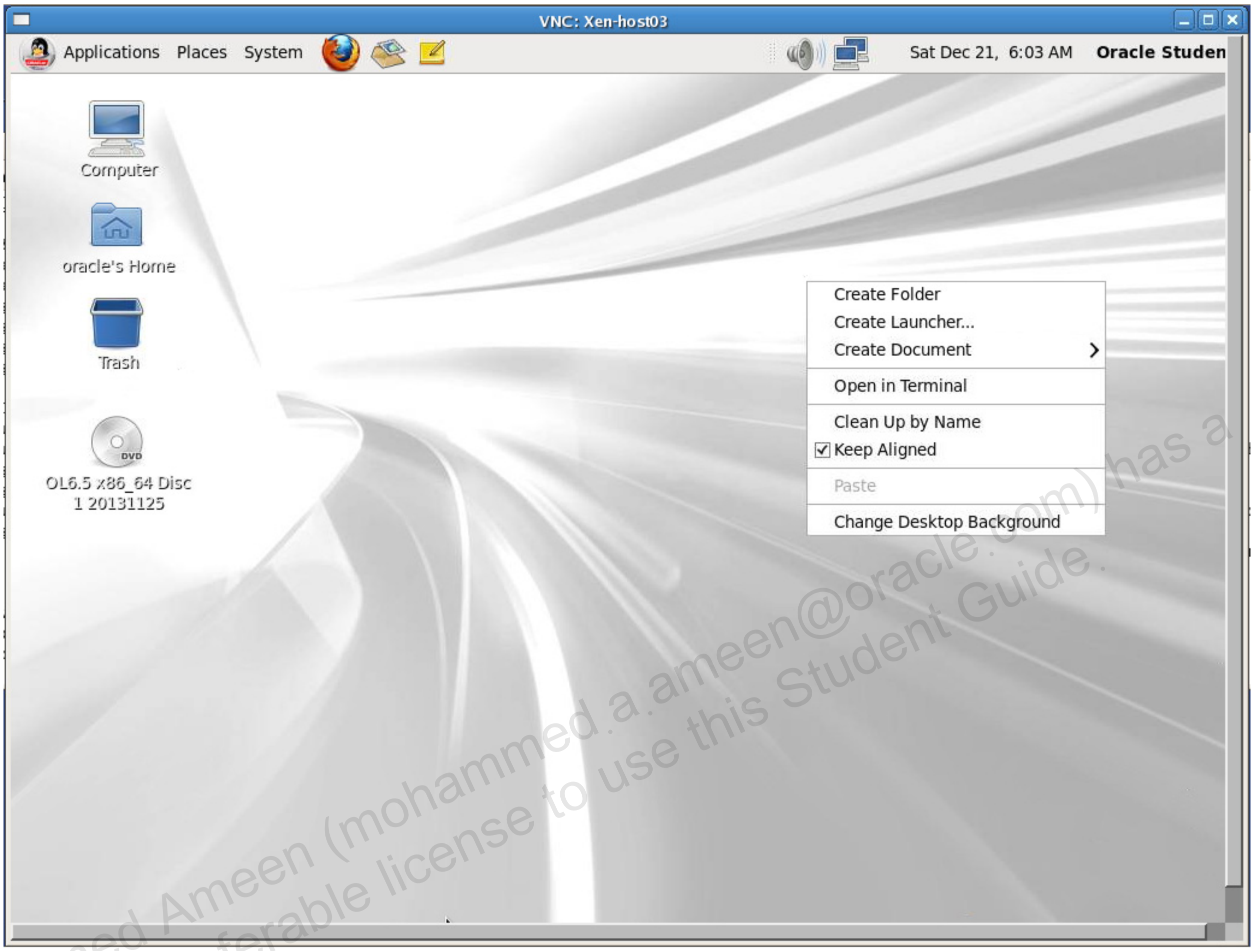
e. The GNOME desktop login window appears as shown in the following screenshot. Select **Oracle Student** from the list of users.



f. You are prompted for the **Oracle Student** password as shown in the following screenshot. Enter the password `oracle` and click **Log In**.



g. The GNOME desktop appears. Right-click the desktop to display the pop-up menu as shown in the following screenshot. Click **Open in Terminal** to open a terminal window.



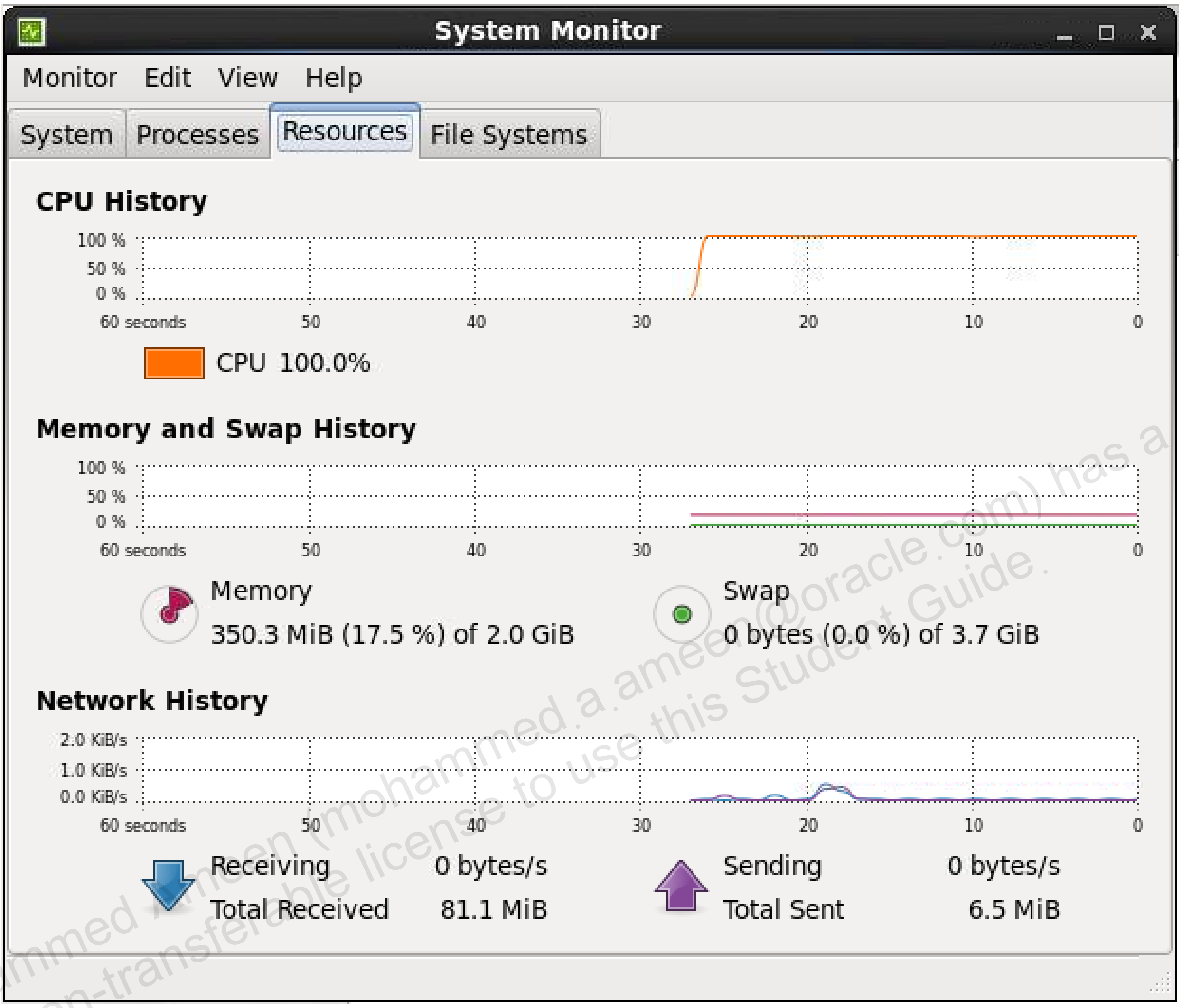
h. From the terminal window, use the `su -` command to become the `root` user. The `root` user's password is `oracle`.

```
$ su -
Password: oracle
#
```

i. From the terminal window, enter the `gnome-system-monitor` command to display the System Monitor GUI.

```
# gnome-system-monitor
```

The System Monitor is displayed as shown in the following screenshot:



j. Click the other tabs and observe the information displayed on each tab:

1) System tab
2) Processes tab
3) File Systems tab

k. Click **Monitor > Quit** to exit the System Monitor GUI.

- Do not exit the GNOME desktop.

Practice 19-3: Installing and Using OSWatcher

Overview

In this practice, you install and run the OSWatcher Black Box (OSWbb) product and view the collected data. Refer to Doc ID 580513.1 for "How To Start OSWatcher Black Box Every System Boot."

Assumptions

- You are the `root` user on the **host03** VM.
- You are the `root` user on **dom0**.

Tasks

1. Use the `sftp` to copy the `oswbb703.tar` file from **dom0** to **host03**.
   a. From **host03**, use the `service` command to verify that `sshd` is running. Start the service if necessary.

```
[host03]# service sshd status
opensh-daemon (pid ...) is running...
```

   - In this example, the service is running.

   b. From **dom0**, use the `cd` command to change to the `/OVS/seed_pool/sfws` directory.

```
[dom0]# cd /OVS/seed_pool/oswbb
```

   c. Use the `ls` command to view the directory for the OSWbb TAR file.

```
[dom0]# ls *tar
oswbb703.tar
```

   d. Use the `sftp` command to connect to **host03** as `root`. Password is `oracle`.

```
[dom0]# sftp root@host03
root@host03's password: oracle
sftp>
```

   e. Use the `put` command to copy the OSWbb TAR file to **host03**.

```
sftp> put oswbb703.tar
Uploading oswbb473.tar to /root/oswbb703.tar
oswbb703.tar ...
```

   f. Use the `quit` command to exit `sftp`.

```
sftp> quit
```

Perform all remaining steps in this practice from **host03**.

2. From **host03**, install, start, and stop OSWbb.
   a. Use the `cd` command to change to the `root` user's home directory.

```
# cd
```

b. Use the `tar` command to extract the `oswbb703.tar` file.

```
# tar xvf oswbb703.tar
oswbb/
...
```

c. Use the `cd` command to change to the `oswbb` directory, and then use `ls` to view the contents of the directory.

```
# cd oswbb
# ls
analysis/      iosub.sh      oswnet.sh         tarupfiles.sh
...
extras.txt     oswbba.jar    startOSWbb.sh
...
```

- Notice the `startOSWbb.sh` file, which is the script used to start OSWbb.

d. Use the `startOSWbb.sh` command to start OSWbb.

```
# ./startOSWbb.sh
Info...You did not enter a value for snapshotInterval.
Info...Using default value = 30
Info...You did not enter a value for archiveInterval.
Info...Using default value = 48
Setting the archive log directory to /root/oswbb/archive

Testing for discover of OS Utilities
VMSTAT found on your system
IOSTAT found on your system
MPSTAT found on your system
NETSTAT found on your system
TOP found on your system

Testing for discover of OS CPU COUNT
...
CPU COUNT = 1
Discovery completed.
Starting OSWatcher Black Box v7.0 ...
With SnapshotInterval = 30
With ArchiveInterval = 48
...
Starting Data Collection...
oswbb heartbeat:...
oswbb heartbeat:...
oswbb heartbeat:...
...
```

- OSWbb started successfully, the discovery process completed and data collection begins.
- The default intervals (`snapshotInterval = 30` and `archiveInterval = 48`) are used.

e. After a few data collection events (`oswbb heartbeat`) have completed, use the `stopOSWbb.sh` command to stop OSWbb.

```
# ./stopOSWbb.sh
```

3. View the data collection directories.

Use the `cd` command to change to the `archive` directory, and then use `ls` to view the contents of the directory.

```
# cd archive
# ls
oswiostat/   oswmpstat/   oswprvtnet/   oswslabinfo/
oswvmstat/   oswmeminfo/  oswnetstat/   oswps/       oswtop/
```

- The `archive` directory is created when OSWbb is started for the first time.
- The directory contains nine subdirectories, one for each data collector.

4. View the `oswiostat` directory.

a. Use the `cd` command to change to the `oswiostat` directory, and then use `ls` to view the contents of the directory.

```
# cd oswiostat
# ls
host03.example.com_iostat...
```

b. Use the `less` command to view the file.

```
# less host03...
Linux OSW v7.0
zzz ***...
...
```

- Notice that this file contains the output of the `iostat` command.
- The `iostat` command ran at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

5. View the `oswmpstat` directory.

a. Use the `cd` command to change to the `oswmpstat` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswmpstat
# pwd
/root/oswbb/archive/oswmpstat
# ls
host03.example.com_mpstat...
```

b. Use the less command to view the file.

```
# less host03...
Linux OSW v7.0
zzz ***...
...
```

- Notice that this file contains the output of the mpstat command.
- The mpstat command ran at 30-second intervals (the value of snapshotInterval).
- Each interval begins with zzz *** characters followed by a time stamp.

6. View the oswprvtnet directory.

   Use the cd command to change to the oswprvtnet directory, and then use ls to view the contents of the directory.

```
# cd ../oswprvtnet
# pwd
/root/oswbb/archive/oswprvtnet
# ls
```

- Notice that this directory is empty.
- This directory contains the status of RAC private networks.
- You need to manually create the private.net file to run traceroute commands.

7. View the oswslabinfo directory.

   a. Use the cd command to change to the oswslabinfo directory, and then use ls to view the contents of the directory.

```
# cd ../oswslabinfo
# pwd
/root/oswbb/archive/oswslabinfo
# ls
host03.example.com_slabinfo...
```

   b. Use the less command to view the file.

```
# less host03...
zzz ***...
slabinfo - version: 2.1
...
```

- Notice that this file contains the contents of the /proc/slabinfo file.
- The /proc/slabinfo file is read at 30-second intervals (the value of snapshotInterval).
- Each interval begins with zzz *** characters followed by a time stamp.

8. View the oswvmstat directory.

a. Use the cd command to change to the oswvmstat directory, and then use ls to view the contents of the directory.

```
# cd ../oswvmstat
# pwd
/root/oswbb/archive/oswvmstat
# ls
host03.example.com_vmstat...
```

b. Use the less command to view the file.

```
# less host03...
Linux OSW v7.0 ...
SNAP_INTERVAL 30
CPU_COUNT 1
zzz ***...
...
```

- Notice that this file contains the output of the vmstat command.
- The vmstat command ran at 30-second intervals (the value of snapshotInterval).
- Each interval begins with zzz *** characters followed by a time stamp.

9. View the oswmeminfo directory.

a. Use the cd command to change to the oswmeminfo directory, and then use ls to view the contents of the directory.

```
# cd ../oswmeminfo
# pwd
/root/oswbb/archive/oswmeminfo
# ls
host03.example.com_meminfo...
```

b. Use the less command to view the file.

```
# less host03...
zzz ***...
MemTotal: ...
MemFree: ...
...
```

- Notice that this file contains the contents of the /proc/meminfo file.
- The /proc/meminfo file is read at 30-second intervals (the value of snapshotInterval).
- Each interval begins with zzz *** characters followed by a time stamp.

10. View the oswnetstat directory.

    a. Use the cd command to change to the oswnetstat directory, and then use ls to view the contents of the directory.

```
# cd ../oswnetstat
# pwd
/root/oswbb/archive/oswnetstat
# ls
host03.example.com_netstat...
```

    b. Use the less command to view the file.

```
# less host03...
Linux OSW v7.0
zzz ***...
...
```

    - Notice that this file contains the output of the netstat command.
    - The netstat command ran at 30-second intervals (the value of snapshotInterval).
    - Each interval begins with zzz *** characters followed by a time stamp.

11. View the oswps directory.

    a. Use the cd command to change to the oswps directory, and then use ls to view the contents of the directory.

```
# cd ../oswps
# pwd
/root/oswbb/archive/oswps
# ls
host03.example.com_ps...
```

    b. Use the less command to view the file.

```
# less host03...
Linux OSW v7.0
zzz ***...
...
```

    - Notice that this file contains the output of the ps command.
    - The ps command ran at 30-second intervals (the value of snapshotInterval).
    - Each interval begins with zzz *** characters followed by a time stamp.

12. View the `ostop` directory.

   a. Use the `cd` command to change to the `ostop` directory, and then use `ls` to view the contents of the directory.

```
# cd ../oswtop
# pwd
/root/oswbb/archive/oswtop
# ls
host03.example.com_top...
```

   b. Use the `less` command to view the file.

```
# less host03...
Linux OSW v7.0
zzz ***...
...
```

- Notice that this file contains the output of the `top` command.
- The `top` command ran at 30-second intervals (the value of `snapshotInterval`).
- Each interval begins with `zzz ***` characters followed by a time stamp.

Practice 19-4: Using OSWatcher Black Box Analyzer

Overview

In this practice, you perform the following:

- Start OSWatcher Black Box Analyzer (OSWbba) on **host03**
- View CPU and Memory Graphs from OSWbba
- Use OSWbba to analyze the data
- View the analysis report

Assumptions

- You are the `root` user on **host03**.
- You completed Practice 19-3: Installing and Using OSWatcher.

Tasks

1. Start OSWatcher Black Box Analyzer (OSWbba) on **host03**.
   a. Use the `java -version` command to display the Java version number.

```
# java –version
java version "1.7.0_45"
...
```

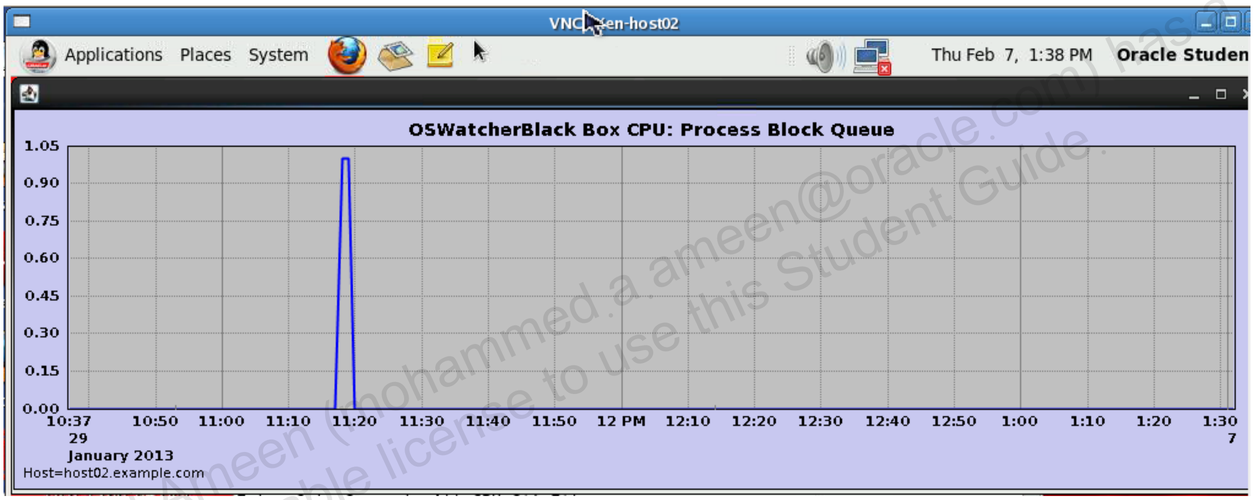
- In this example, version `1.7.0_45` is installed.
- The minimum version is `1.4.2`.

   b. Use the `java –jar` command to display the OSWbba menu. First, ensure that you are in the directory where OSWbba is installed (`~/oswbb`).

```
# cd ~/oswbb
# java –jar oswbba.jar –i ~/oswbb/archive
...
Enter 1 to Display CPU Process Queue Graphs
Enter 2 to Display CPU Utilization Graphs
Enter 3 to Display CPU Other Graphs
Enter 4 to Display Memory Graphs
Enter 5 to Display Disk IO Graphs

Enter 6 to Generate All CPU Gif Files
Enter 7 to Generate All Memory Gif Files
Enter 8 to Generate All Disk Gif Files

Enter L to Specify Alternate Location of Gif Directory
Enter T to Specify Different Time Scale
Enter D to Return to Default Time scale
Enter R to Remove Currently Displayed Graphs
```

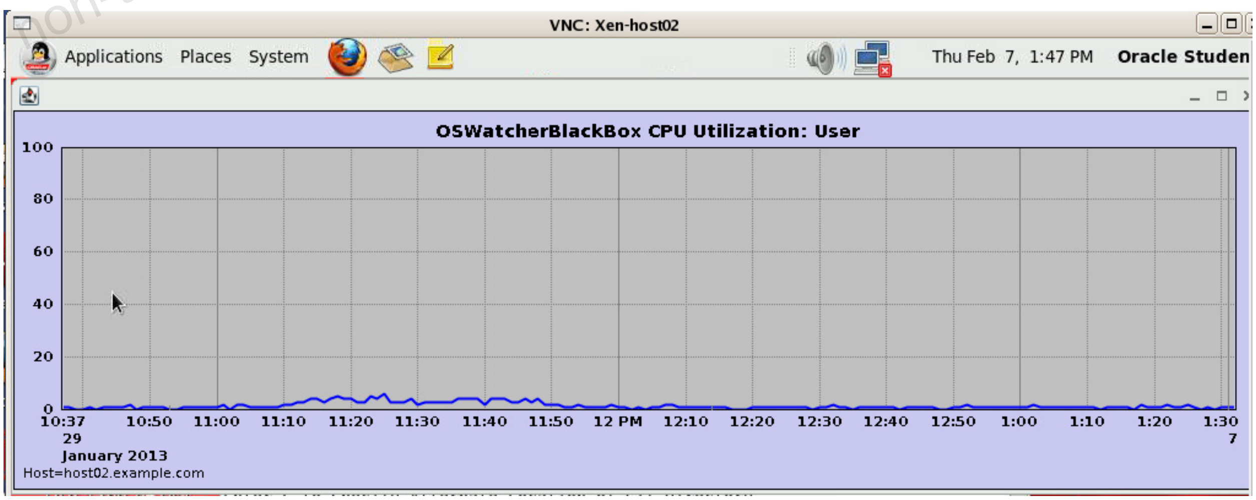
```
Enter A to Analyze Data
Enter S to Analyze Subset of Data(Changes analysis dataset
including graph time scale)

Enter P to Generate A Profile
Enter X to Export Parsed Data to File
Enter Q to Quit Program

Please Select an Option:
```

2. View CPU and Memory Graphs. A sample output is shown as follows. Your graphs will be different.
   a. From the OSWbba menu, select `1` to display the CPU Process Queue graph.



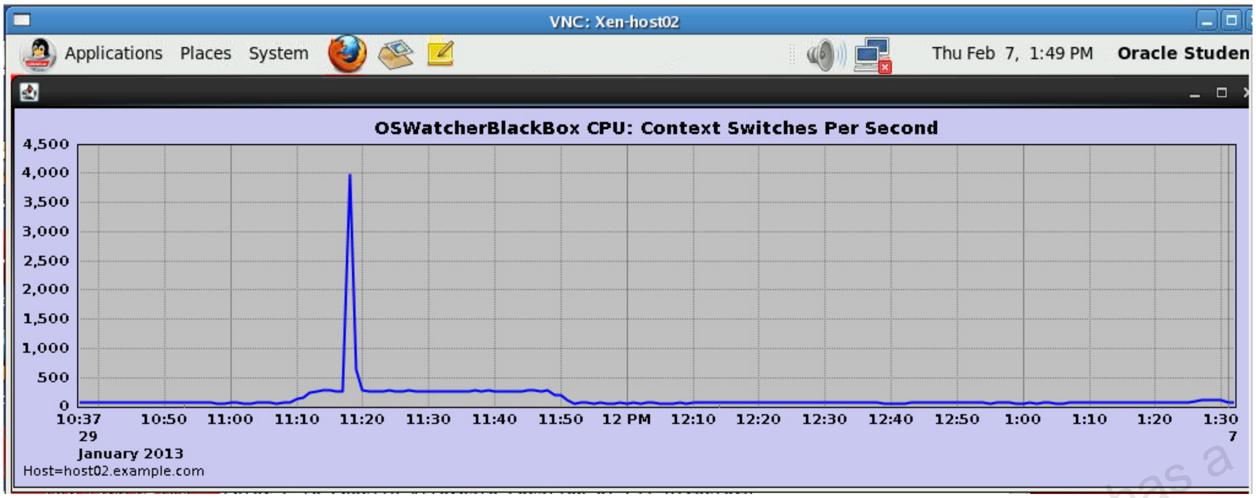
- After viewing the graph, select `R` from the menu to remove the graph.

   b. From the OSWbba menu, select `2` to display the CPU Utilization graph.



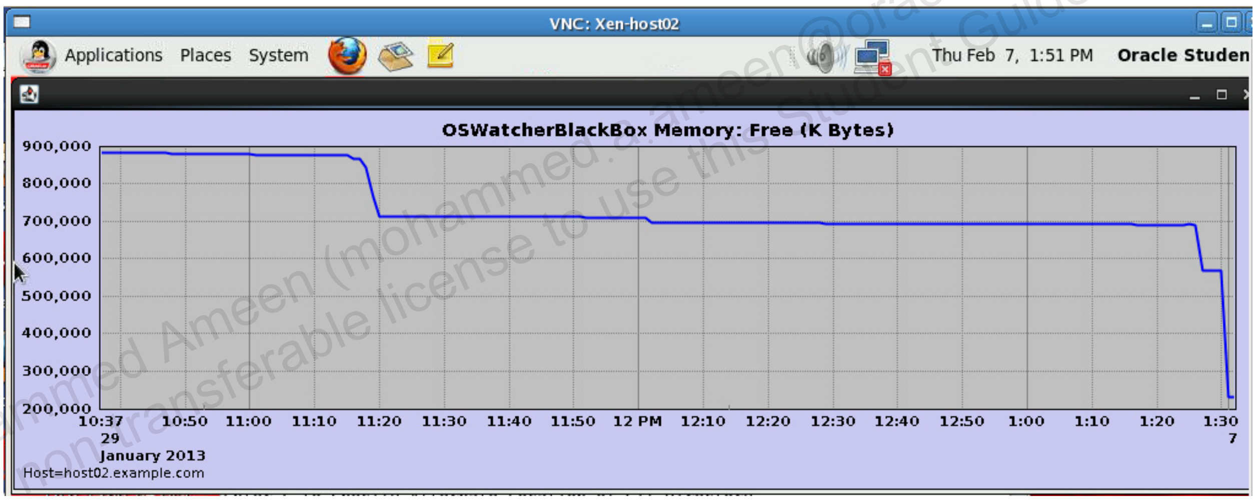
- After viewing the graph, select `R` from the menu to remove the graph.

c. From the OSWbba menu, select `3` to display the CPU Other graph.



- After viewing the graph, select `R` from the menu to remove the graph.

d. From the OSWbba menu, select `4` to display the Memory Graph.



- After viewing the graph, select `R` from the menu to remove the graph.

3. Run the OSWbba analyzer and view the report.

a. From the OSWbba menu, select `A` to analyze the collected data and produce a report. Provide a file name or press Enter to accept the default name.

```
...
Enter A to Analyze Data
...
Please Select an Option:a
Enter a fully qualified analysis file name or enter <CR> to
accept default name: ENTER
A new analysis file analysis/host03...txt has been created.
```

- This message “A new analysis file analysis/host03...txt has been created” appears and the menu is then re-displayed.

- You could alternatively run the analyzer from the command line by using the following command (do not run this command, this is information only):

```
# java -jar oswbba.jar -i ~/oswbb/archive -A
```

b. Exit the OSWbba menu by selecting option `Q` to quit.

c. From the **host03** command line, use the `ls` command to display the analysis file in the `~/oswbb/analysis` directory.

```
# ls ~/oswbb/analysis
host03.example.com...txt
```

- Your analysis file name is different from the sample file name shown.

d. Use the `less` command to view the analysis file.

- Use your analysis file name observed in the previous step.

```
# less ~/oswbb/analysis/host03*
...
```

- Notice the file has system information at the top followed by sections such as:
  - Section 1: System Status
  - Section 2: System Slowdown
    - Section 2.1: System Slowdown RCA Ordered By Impact
  - Section 3: System General Findings
  - Section 4: CPU Detailed Findings
    - Section 4.1: CPU Run Queue:
    - Section 4.2: CPU Utilization: Percent Busy
    - Section 4.3: CPU Utilization: Percent Sys
  - Section 5: Memory Detailed Findings
    - Section 5.1: Memory: Process Swap Queue
    - Section 5.2: Memory: Scan Rate
    - Section 5.3: Memory: Page In:
    - Section 5.4: Memory: Page Tables (Linux only):
    - Section 5.5: Top 5 Memory Consuming Processes Beginning
    - Section 5.6: Top 5 Memory Consuming Processes Ending
  - Section 6: Disk Detailed Findings
    - Section 6.1: Device Percent Busy Findings
    - Section 6.2: Device Service Times Findings
    - Section 6.3: Device Wait Queue Times Findings
  - Section 7: Network Detailed Findings
    - Section 7.1: Network Data Link Findings
    - Section 7.2: Network IP Findings
    - Section 7.3: Network UDP Findings
    - Section 7.4: Network TCP Findings
  - Section 8: Process Detailed Findings

- Section 8.1: PS Process Summary Ordered By Time
- Section 8.2: PS for Processes With Status = D or T Ordered By Time
- Section 8.3: PS for (Processes with CPU > 0) When System Idle CPU < 30% Ordered By Time
- Section 8.4: Top VSZ Processes Increasing Memory Per Snapshot
- Section 8.5: Top RSS Processes Increasing Memory Per Snapshot

e. After viewing the analysis file, press `q` to exit the `less` command.

4. Terminate the `dd` command started in Practice 19-2 (3b).

   In the second terminal window on **host03** in which you initiated the following `dd` command, press Ctrl + C to terminate the command.

```
# dd if=/dev/zero of=/dev/null bs=1024
CTRL-C
```

Practices for Lesson 20: System Logging

Chapter 20

Practices for Lesson 20: System Logging

Practices Overview

In these practices, you configure system logging, use `rsyslog` templates to format message logs, and install and run `logwatch`.

Practice 20-1: Viewing the System Log File Configuration

Overview

In this practice, you view the system logging configuration file, modify the file, and observe the impact of the modifications. You also configure log file rotation.

Assumptions

You are the `root` user on the **host03** VM.

Tasks

1. Explore the main configuration file for system logging, `/etc/rsyslog.conf`.
   a. Use the `less` command to view the system logging configuration file.

```
# less /etc/rsyslog.conf
# rsyslog v5 configuration file
...
#### MODULES ####
$ModLoad imuxsock.so # provides support for local system...
$ModLoad imklog.so   # provides kernel logging support...
...
#### GLOBAL DIRECTIVES ####
# use default timestamp format
$ActionFileDefaultTemplate RSYSLOG_TraditionalFileFormat
...
#### RULES ####
...
# Log all kernel messages to the console.
# Logging much else clutters up the screen
#kern.*                                       /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none     /var/log/messages

# The authpriv file has restricted access.
authpriv.*                                    /var/log/secure

# Log all the mail messages in one place.
mail.*                                       -/var/log/maillog

# Log cron stuff
cron.*                                       /var/log/cron
...
```

b. Use the up-arrow and down-arrow keys to view the various sections of the file.
   - The **MODULES** section uses the `$ModLoad` command to load the modules.
   - The **GLOBAL DIRECTIVES** section specifies configuration options.
   - The **RULES** section defines a filter (facility.priority) and an action.

c. Press the letter **q** to exit the "`less`" command.

2. Change the action for `cron` logging.

a. Use the `ls` command to list the `cron*` files in the `/var/log` directory.
   - Because this is a freshly installed system, you might not have any rotated log files (files with a date stamp).

```
# ls /var/log/cron*
cron   cron-20131222
```

b. Use the `vi` editor to modify the system logging configuration file. Change the action for `cron` logging to log to a different log file: `/var/log/cron_new`.

```
# vi /etc/rsyslog.conf
...
# Log cron stuff
cron.*              /var/log/cron        (old entry)
cron.*              /var/log/cron_new    (new entry)
```

c. Use the `service` command to restart the `rsyslog` service.

```
# service rsyslog restart
Shutting down system logger:                [  OK  ]
Starting system logger:                     [  OK  ]
```

d. Use the `crontab -e` command to create a `cron` job that runs the `ls` command every minute.

```
# crontab -e
* * * * * ls
```

e. Use the `ls` command to list the `cron*` files in the `/var/log` directory.

```
# ls /var/log/cron*
cron    cron-20131222    cron_new
```

   - Notice the new log file, `cron_new`.

f. Use the `tail` command to view the last two lines in the `cron` log file.

```
# tail -2 /var/log/cron
Dec 13 11:22:51 host03 crontab[8129]: (root) LIST (root)
Dec 13 11:23:01 host03 CROND[8129]: (root) CMD (root)
```

   - This is sample output only.

g. Use the `head` command to view the first entries in the `cron_new` log file.

```
# head /var/log/cron_new
Dec 13 11:24:01 host03 CROND[8178]: (root) CMD (ls)
Dec 13 11:25:01 host03 CROND[8196]: (root) CMD (ls)
...
```

- Notice from the date_time stamps that the new log entries are being written to `cron_new`.

h. Use the `vi` editor to modify the system logging configuration file. Change the action for `cron` logging back to the original log file.

```
# vi /etc/rsyslog.conf
...
# Log cron stuff
cron.*              /var/log/cron_new       (old entry)
cron.*              /var/log/cron           (new entry)
```

i. Use the `service` command to restart the `rsyslog` service.

```
# service rsyslog restart
Shutting down system logger:                    [  OK  ]
Starting system logger:                         [  OK  ]
```

j. Use the `tail` command to ensure that `cron` is now logging to `/var/log/cron`.

```
# tail /var/log/cron
...
Dec 13 11:44:01 host03 CROND[8404]: (root) CMD (ls)
# tail /var/log/cron_new
...
Dec 13 11:43:01 host03 CROND[8376]: (root) CMD (ls)
```

- Notice that the entry in `cron` has a later time stamp than the entry in `cron_new`.
- You might need to wait a minute for the `cron` job to run.

k. Use the `rm` command to delete the `cron_new` log file.

```
# rm /var/log/cron_new
rm: remove regular file 'cron_new'? y
```

l. Use `crontab -r` to remove the `crontab`.

```
# crontab -r
```

3. Configure `rsyslog` to log debug messages.

a. Use the `vi` editor to modify the system logging configuration file. Add an entry at the bottom of the file to log all debug messages to `/var/log/debug`.

```
# vi /etc/rsyslog.conf
...
*.debug                                 /var/log/debug
```

b. Use the `service` command to restart the `rsyslog` service.

```
# service rsyslog restart
Shutting down system logger:                [  OK  ]
Starting system logger:                     [  OK  ]
```

c. Use the `logger` command to generate an **informational** log message.

```
# logger –p info "This is an info-priority message"
```

d. Use the `tail` command to view the log files.

```
# tail /var/log/messages
...
Dec 13 13:36:12 host03 oracle: This is an info-priority message
# tail /var/log/debug
...
Dec 13 13:36:12 host03 oracle: This is an info-priority message
```

- Notice that the message was written to both log files.

e. Use the `logger` command to generate a **debug-level** log message.

```
# logger –p debug "This is a debug-priority message"
```

f. Use the `tail` command to view the log files.

```
# tail /var/log/messages
...
Dec 13 13:36:12 host03 oracle: This is an info-priority message
# tail /var/log/debug
...
Dec 13 13:42:16 host03 oracle: This is a debug-priority message
```

- Notice that the debug-level message was written only to `/var/log/debug`.

g. Use the `vi` editor to modify `/etc/rsyslog.conf` and remove the entry at the bottom of the file to log all debug messages.

- Remove this entry: `*.debug      /var/log/debug`.

h. Use the `service` command to restart the `rsyslog` service.

```
# service rsyslog restart
Shutting down system logger:                [  OK  ]
Starting system logger:                     [  OK  ]
```

4. Configure log file rotation.

a. Use the `ls` command to view the contents of the `/var/log` directory.

```
# ls /var/log/messages*
/var/log/messages  /var/log/messages-20131222
# ls /var/log/maillog*
/var/log/maillog   /var/log/maillog-20111211
# ls /var/log/cron*
/var/log/cron      /var/log/cron-20111211
```

- Notice that some files in `/var/log` have numbers at the end of the file name.
- These numbers represent a rotated log with the time stamp added to the log file name.
- You may not have log files with a time stamp appended to the file name. It depends on how long your system has been running.

b. Use the `vi` editor to modify the `/etc/logrotate.conf` configuration file. Change the frequency of the default log file rotation from **weekly** to **daily**.

```
# vi /etc/logrotate.conf
...
# rotate log files weekly
weekly                  (old entry)
daily                   (new entry)
```

- Your log files now rotate daily after making this change.
- Because this is a freshly installed system, there will be no rotated log files.

Practice 20-2: Using rsyslog Templates

Overview

In this practice, you use `rsyslog` templates to format `rsyslog` output.

Assumptions

You are the `root` user on the **host03** VM.

Tasks

1. Use the `vi` editor to modify `/etc/rsyslog.conf` and define a template.

   Add a new line at the bottom of the file as follows:

   ```
   $template class,"Message: %msg%\n"
   ```

   - This entry creates a template named `class`.

2. Continuing editing `/etc/rsyslog.conf` and create a log file that uses the template.

   Add a new line at the bottom of the file as follows:

   ```
   *.*    /var/log/class.log;class
   ```

   - This entry must be after the entry that defined the template.
   - This entry writes all messages to the `/var/log/class.log` file and formats the entries using the template `class`.

3. After saving the changes to `/etc/rsyslog.conf`, use the `service` command to restart the `rsyslog` service.

   ```
   # service rsyslog restart
   Shutting down system logger:              [  OK  ]
   Starting system logger:                   [  OK  ]
   ```

4. Use the `cat` command to view the `/var/log/class.log` file.

   ```
   # cat /var/log/class.log
   Message: ...
   ...
   ```

   - Notice that all entries are preceded by the text "`Message:`" followed by the actual message, as defined in the template `class`.

5. Use the `vi` editor to modify `/etc/rsyslog.conf` and modify the `class` template.

   Change the template definition as follows:

   ```
   $template class,"Time: %timestamp%, Facility: %syslogfacility-
   text%, Priority: %syslogpriority-text%, Hostname: %hostname%,
   Message: %msg%\n"
   ```

6. After saving the changes to `/etc/rsyslog.conf`, use the `service` command to restart the `rsyslog` service.

```
# service rsyslog restart
Shutting down system logger:            [  OK  ]
Starting system logger:                 [  OK  ]
```

7. Use the `cat` command to view the `/var/log/class.log` file.

```
# cat /var/log/class.log
Message: ...
...
Time: ...
...
```

- Notice that the newest entries now include the `Time`, `Facility`, `Priority`, `Hostname`, and `Message` properties, as defined in the template `class`.

8. Use the `vi` editor to modify `/etc/rsyslog.conf` and delete the following entries you added in this practice.
   - `$template class,”Time: %timestamp%, Facility: %syslogfacility-text%, Priority: %syslogpriority-text%, Hostname: %hostname%, Message: %msg%\n”`
   - `*.*   /var/log/class.log;class`

9. After saving the changes to `/etc/rsyslog.conf`, use the `service` command to restart the `rsyslog` service.

```
# service rsyslog restart
Shutting down system logger:            [  OK  ]
Starting system logger:                 [  OK  ]
```

10. Use the `rm` command to remove the `/var/log/class.log` file.

```
# rm /var/log/class.log
rm: remove regular file ‘/var/log/class.log’? y
```

Practice 20-3: Using `logwatch`

Overview

In this practice, you install the `logwatch` package, view the main configuration file (the `cron` file), and run the `logwatch` utility from the command line.

Assumptions

You are the `root` user on the **host03** VM.

Tasks

1. Install the `logwatch` package.
   a. Use the `rpm` command to determine whether the `logwatch` package is already installed.

```
# rpm -q logwatch
```

   - In this example, the `logwatch` package is not installed.

   b. Use the `yum` command to install the `logwatch` package.

```
# yum install logwatch
...
Setting up Install Process
Resolving Dependencies
--> Running transaction check
...
--> Finished Dependency Resolution
Dependencies Resolved
===================================================
Package
===================================================
Installing:
 logwatch
Installing for dependencies:
 perl-Date-Manip
 perl-YAML-Syck
Transaction Summary
===================================================
Install    3 Package(s)
Total download size: 1.7 M
Installed size: 11 M
Is this ok [y/N]: y
Downloading packages
...
Running rpm_check debug
```

```
Running Transaction Test
...
  Installing : perl-YAML-Syck-1.07-4.el6.x86_64
  Installing : perl-Date-Manip-6.24-1.el6.noarch
  Installing : logwatch-7.3.6-49.el6.noarch
...
Installed:
 logwatch.noarch 0:7.3.6-52.el6
Dependency Installed:
 perl-Date-Manip.noarch 0:6.24-1.el6   perl-YAML-Syck.x86_64...

Complete!
```

c. Use the `rpm` command to verify that `logwatch` has successfully installed.

```
# rpm –q logwatch
logwatch-7.3.6-52.el6.noarch
```

2. View `logwatch` files.

a. Use the `find` command to list all `logwatch` files.

```
# find / -name "*logwatch*"
/var/cache/logwatch
/var/lib/yum/yumdb...logwatch-7.3.6-52.el6.noarch
/usr/sbin/logwatch
...
/usr/share/logwatch/default.conf/logwatch.conf
/usr/share/logwatch/scripts/logwatch/pl
...
/etc/cron.daily/0logwatch
/etc/logwatch
/etc/logwatch/conf/logwatch.conf
...
```

b. Use the `less` command to view the main `logwatch` configuration file.

```
# less /usr/share/logwatch/default.conf/logwatch.conf
...
```

- Notice various configurable items such as the following:
  - `LogDir`
  - `TmpDir`
  - `MailTo`
  - `Print`
  - `Save`
  - `Range`
  - `Detail`

- Service
- DailyReport

c. Use the `less` command to view the `logwatch cron` file.

```
# less /etc/cron.daily/0logwatch
...
```

3. Run `logwatch` from the command line.

a. Run the `logwatch --help` command to view the `logwatch` help.

```
# logwatch --help
Usage: /usr/sbin/logwatch [--detail <level] [--logfile <name>]
   [--print] [--mailto <addr>] ...
```

b. Run `logwatch` and display output to your screen.

```
# logwatch --print
################# logwatch 7.3.6 (05/19/07) #################
       Proccessing Initiated: Fri Jan...
       Date Range Processed: yesterday
...
```

Practices for Lesson 21: Troubleshooting

Chapter 21

Practices for Lesson 21: Troubleshooting

Practices Overview

In these practices, you perform a variety of troubleshooting exercises and fix some common problems. Specific problems you encounter include the following:

- System boots into single-user mode by default.
- Status utilities are not producing expected output.
- A `cron` job fails to run.
- A user cannot log in.
- File system does not mount.
- Logical volume space is exhausted.
- You cannot `ping` remote hosts.
- There are NFS permission problems.
- You cannot log in to remote hosts using `ssh`.
- Log file is not getting updated.

Each practice:

- Tells you what problem the activity is designed to simulate
- Tells you how to set up the problem
- Gives you some hints or things to check
- Has an associated "Solution" that provides steps to diagnose and resolve the problem

Each solution:

- Includes only the steps to resolve the problem
- Does not include steps to set up the problem

Appendix:

- An appendix exists, which provides the source code for the executables that cause the problems that you need to troubleshoot and fix.

Practice 21-1: Transferring Utilities from dom0

Overview

In this practice, you transfer files hosted on **dom0** to **host01**. In subsequent practices, you are instructed to run one of these executables that causes the problem that you need to troubleshoot and fix.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Use the `sftp` command to transfer files from **dom0** to **host01**.
   a. From the root user's home directory on **host01**, use the `sftp` command to connect to **dom0**.
      - Answer "`yes`" when prompted, "Are you sure…"
      - Provide `root` password of `oracle` when prompted.

```
# cd
# pwd
/root
# sftp dom0
The authenticity of host 'dom0 (192.0.2.1)' can't be
established. RSA key fingerprint is ...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.0.2.1' (RSA) to the list of
known hosts.
root@dom0's password: oracle
sftp>
```

   b. From the `sftp` prompt, use the `cd` command to change to the `/OVS/seed_pool/new_stuff/ts_scripts` directory on **dom0**. Use the `ls` command to list the directory contents.

```
sftp> cd /OVS/seed_pool/ts_scripts
sftp> ls
two.x
three.x
four.x
five.x
six.x
eight.x
nine.x
ten.x
eleven.x
sftp>
```

c. From the `sftp` prompt, use the `mget *` command to upload all the files in the `/OVS/seed_pool/ts_scripts` directory from **dom0** to **host01**.

```
sftp> mget *
Fetching .../two.x to two.x
...
sftp>
```

d. Use the `quit` command to exit `sftp`.

```
sftp> quit
```

Practice 21-2: System Boots into Single-User Mode

Overview

In this practice, your system boots into single-user mode by default, which is not what you want.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Display the current run level.

   Use the `who -r` command to view the current run level.

```
# who -r
run-level 3 <date_time>
```

   - Notice the run level is `3`.

2. Execute the `two.x` program from the `root` user's home directory.

```
# cd
# pwd
/root
# ./two.x
```

3. Reboot the system.

   a. Use the `reboot` command to reboot your system:

```
# reboot
...
```

   - After you reboot your system, your VNC session closes.

   b. Connect to **host01** by using VNC.

      1) Run the `vncviewer&` command.

```
# vncviewer&
```

      - The VNC Viewer: Connection Details dialog box is displayed.

      2) Enter the command `localhost:<port_number>`, substituting the correct port number for the **host01** guest. For example, if the port number is 5900, enter the following and click **OK**:

```
localhost:5900
```

   - Notice you did not need to log in. You were automatically logged in as `root`.

   c. Use the `who -r` command to view the current run level.

```
# who -r
run-level S <date_time>
```

   - Notice the run level is `S` or single-user mode.

4. Diagnose and fix the problem so that the system does not boot into single-user mode.

- Review Lesson 4: Linux Boot Process
  - Which file specifies the default run level? Is the default run level set to `S`?
  - Which file allows you to view kernel boot parameters? Is `single` a kernel boot parameter?
  - Which file allows you to specify kernel boot parameters? Is `single` specified as a kernel boot parameter?
  - How do you change kernel boot parameters at boot time?
  - How do you permanently change kernel boot parameters?

5. Reboot the system after problem is fixed to confirm it does not boot into single-user mode.

Solution 21-2: System Boots into Single-User Mode

Steps

1. Determine why the system boots into single-user mode.

   a. Use the `cat` command to view the default run level set in the `/etc/inittab` file.

```
# cat /etc/inittab
...
id:3:initdefault:
```

   - Notice the default run level is `3`, not `S`. Therefore, this is not causing the problem.

   b. Use the `cat` command to view the kernel boot parameters in the `/proc/cmdline` file.

```
# cat /proc/cmdline
ro root=UUID=... rd_NO_LUKS rd_NO_LVM LANG-en_US.UTF-8 rd_NO_MD
SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM
rhgb quiet single
```

   - Notice the word `single` is a kernel boot parameter that is causing your system to boot into single-user mode.

   c. Use the `grep` command and search for the word "`single`" in the `/boot/grub/grub.conf` file.

```
# grep single /boot/grub/grub.conf
kernel /vmlinuz-3.8.13-16... single
```

   - Notice the word `single` appears on the kernel line for the `3.8.13-16` kernel.
   - This would cause the `3.8.13-16` kernel to boot into single-user mode by default.

   d. Use the `uname -r` command to determine which kernel is running.

```
# uname -r
3.8.13-16.2.1.el6uek.x86_64
```

   - The `3.8.13-16` kernel is running. The word `single` needs to be removed from the associated kernel line in the `/boot/grub/grub.conf` file.

2. Fix the system so it does not boot into single-user mode.

   a. Use the `vi` editor to modify `/boot/grub/grub.conf` and remove the word `single` at the end of the `3.8.13-16` kernel line.

```
# vi /boot/grub/grub.conf
...
default=0
...
title Oracle Linux Server Unbreakable Kernel (3.8.13-16...
        root (hd0,0)
        kernel /vmlinuz-3.8.13-16... single
...
```

   b. Press Ctrl + D to continue the boot process.

- The login prompt appears after the boot process completes.

```
# CTRL-d
...
host01 login:
```

c. Log in as `root`. Password is `oracle`.

```
host01 login: root
Password: oracle
```

d. Use the `who -r` command to view the current run level.

```
# who -r
run-level 3 <date_time>              last=S
```

- Notice the run level is `3`. The last run level was `S`.

3. Confirm the system does not boot into single-user mode.

a. Use the `reboot` command to reboot your system:

```
# reboot
...
```

- After you reboot your system, your VNC session closes.

b. Connect to **host01** by using VNC.

1) Run the `vncviewer&` command.

```
# vncviewer&
```

- The VNC Viewer: Connection Details dialog box is displayed.

2) Enter the command `localhost:<port_number>`, substituting the correct port number for the **host01** guest. For example, if the port number is 5900, enter the following and click **OK**:

```
localhost:5900
```

- Notice you do need to log in this time.

c. Log in as `root`. The password is `oracle`.

```
host01 login: root
Password: oracle
```

d. Use the `who -r` command to view the current run level.

```
# who -r
run-level 3 <date_time>
```

- Notice the run level is now `3` and not single-user.

Practice 21-3: Status Commands Fail

Overview

In this practice, you notice that some of the "status" utilities are not producing the expected output. You diagnose and fix this problem.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Execute the `three.x` program from the `root` user's home directory.

```
# cd
# pwd
/root
# ./three.x
```

2. Run some "status" tools and notice the errors.

   a. Run the `mpstat` command.

```
# mpstat
Linux 3.8.13-16...
Cannot open /proc/stat: No such file or directory
```

   b. Run the `iostat` command.

```
# iostat
Linux 3.8.13-16...
Cannot open /proc/stat: No such file or directory
```

   c. Run the `netstat` command.

```
# netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address Foreign Address State
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags    Type    State   I-Node Path
```

   d. Run the `route` command.

```
# route
/proc/net/route: No such file or directory
INET (IPv4) not configured in this system.
```

   e. Run the `ifconfig` command.

```
# ifconfig
Warning: cannot open /proc/net/dev (No such file or directory).
Limited output.
...
```

- Notice that none of these tools produce the expected output.

3. Diagnose and fix the problem with the "status" tools.
   - Review Lesson 5: System Configuration
     - Commands that are failing are commands that provide information about the current state of the kernel.
     - Which file system contains a hierarchy of special files that represent the current state of the kernel?
     - How do you check if this file system is mounted?
     - If this file system is not mounted, how do you mount it?
4. Re-run some of the previous commands that failed to ensure that the problem was fixed.
   a. Run the `mpstat` command.
   b. Run the `iostat` command.
   c. Run the `netstat` command.
   d. Run the `route` command.
   e. Run the `ifconfig` command.

Solution 21-3: Status Commands Fail

Steps

1. Some of the output indicates a problem with the `proc` file system. Diagnose the problem.
   a. Use the `ls` command to display the contents of `/proc`.

```
# ls /proc
```

   - No output is not normal and suggests the `proc` file system is not mounted.

   b. Run the `mount` command to display mounted file systems.

```
# mount
/dev/xvda2 on / type ext4 (rw)
sysfs on /sys type sysfs (rw)
...
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
```

   - Notice that `proc` is not mounted.

   c. Run the `df -h` command to display mounted file systems.

```
# df -h
df: `/proc/sys/fs/binfmt_misc': No such file or directory
Filesystem        Size  Used  Avail  Use%  Mounted on
/dev/xvda2        5.8G  2.2G   3.3G   40%  /
...
```

   - Notice this output reports that `/proc/sys/fs/binfmt_misc` does not exist.

2. Fix the problem.
   a. View the `/etc/fstab` file to ensure the entry to mount `proc` exists.

```
# cat /etc/fstab
...
proc                  /proc       proc     defaults 0   0
```

   - Notice there is an entry to mount the `proc` file system in the `/etc/fstab` file.

   b. Run the `mount -a` command to re-mount entries in the `/etc/fstab` file.

```
# mount -a
```

3. Re-run some of the previous “status” commands to ensure that the problem is fixed.
   a. Run the `mpstat` command.
   b. Run the `iostat` command.
   c. Run the `netstat` command.
   d. Run the `route` command.
   e. Run the `ifconfig` command.

Practice 21-4: `cron` Job Fails to Run

Overview

In this practice, you diagnose and fix a problem that is preventing a `cron` job from running.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Create a `crontab` for the `root` user.
   a. Use the `crontab -e` command to create a `crontab` job that runs the `vmstat` command every minute.

```
# crontab -e
* * * * * vmstat
```

   - The `crontab -e` command uses the `vi` editor. Save your changes and exit `vi`.

   b. After one minute, use the `tail` command to view the last few lines in the `/var/log/cron` log file.

```
# tail /var/log/cron
...
<date_time> host01 CROND[...] (root) CMD (vmstat)
```

   - Notice there is an entry in this file that states the date and time that the `cron` daemon ran the `vmstat` command.
   - You also have mail, because the output from `cron` jobs is sent to the user's mailbox.

   c. Use the `mail` command to view the results of your job.

```
# mail
...
N # Cron Daemon  <date_time> ... "Cron <root@host01> vm"
&
```

   - Notice you have mail that contains the output of the `vmstat` command.

   d. View the details of mailbox entries by entering the associated number and then press **Enter**.
   - To re-display the header, press **h** and then press **Enter**.
   - To quit the mail program, press **q** and then press **Enter**.

   e. Press **q** and then **Enter** to quit the mail program.

```
& q
```

2. Execute the `four.x` program from the `root` user's home directory.

```
# cd
# pwd
/root
# ./four.x
```

3. Edit the `crontab` for the `root` user.

   a. Use the `crontab -e` command to edit the `crontab` job and replace `vmstat` with `iostat`.

```
# crontab -e
* * * * * vmstat                                  (old entry)
* * * * * iostat                                  (new entry)
```

   b. After one minute, use the `tail` command to view the last few lines in the `/var/log/cron` log file.

```
# tail /var/log/cron
...
```

   - Notice there is NO entry in this file that states that the `iostat` command ran.

   c. Use the `mail` command to view your incoming mailbox.

```
# mail
...
&
```

   - Notice there is NO mail entry containing the output of the `iostat` command.

   d. Press **q** and then **Enter** to quit the mail program.

```
& q
```

4. Diagnose and fix the problem of the `cron` job failing to run.
   - Review Lesson 8: Automating Tasks
     - Does the `root` user have a `crontab` entry?
     - Is the `root` user's `crontab` entry valid?
     - Are the permissions correct on the `root` user's `crontab` file?
     - Is the `cron` daemon running?

5. Verify that the `cron` job is running every minute.

   a. Run the `mail` command to view the output of the `iostat` command.

   b. Use the `tail` command to view `/var/log/cron` and ensure the `cron` job is running every minute.

6. Remove the `root` user's `crontab` before continuing to the next practice.

Solution 21-4: `cron` Job Fails to Run

Steps

1. Diagnose the problem of the `cron` job failing to run.
   a. Use the `crontab -l` command to list the `crontab` entry.

```
# crontab -l
* * * * * iostat
```

   - Notice the `crontab` entry exists and the format is valid.

   b. Use the `ls -l` command to view the root user's `crontab` entry in the `/var/spool/cron` directory.

```
# ls -l /var/spool/cron
-rw-------. root root ... root
```

   - The `root` file exists with the proper permissions (read-write for the owner).

   c. Use the `cat` command to view the `/var/spool/cron/root` file.

```
# cat /var/spool/cron/root
* * * * * iostat
```

   - This confirms the `crontab` entry is valid.

   d. Use the `service` command to view the status of the `crond` daemon.

```
# service crond status
crond is stopped
```

   - Notice the `crond` service is not running. This is the cause of the problem.

2. Fix the problem of the `cron` job failing to run.

   Use the `service` command to start the `crond` daemon.

```
# service crond start
Starting crond:                                    [  OK  ]
```

3. Verify that the `cron` job is running every minute.
   a. After one minute, use the `tail` command to view the last few lines in the `/var/log/cron` log file.

```
# tail /var/log/cron
...
<date_time> host01 CROND[...] (root) CMD (iostat)
```

   - Notice there is now an entry in this file that states the date and time that the `cron` daemon ran the `iostat` command.

b. Use the `mail` command to view the results of your job.

```
# mail
...
N # Cron Daemon  <date_time> ... "Cron <root@host01> io"
```

- Notice you have mail that contains the output of the `iostat` command.

c. Press **q** and then **Enter** to quit the mail program.

```
& q
```

4. Remove the `root` user's `crontab` before continuing to the next practice.

```
# crontab -r
```

Practice 21-5: User Cannot Log In

Overview

In this practice, you diagnose and fix a problem which is preventing a user from being able to log in.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Add a new user.
   a. User the `useradd john` command to add user `john`.

```
# useradd john
```

   b. Use the `passwd john` command to assign a password (`password`) to user `john`.
      - Disregard the "BAD PASSWORD" warning.

```
# passwd john
Changing password for user john.
New password: password
BAD PASSWORD: it is based on a dictionary word
Retype new password: password
passwd: all authentication tokens updated successfully.
```

2. Verify you can log in as user `john`.
   a. Use the `exit` command to log out as user `root`.

```
# exit
logout
```

   - The "host01 login" prompt appears.

   b. Log in as user `john`. Password is `password`. Verify you can successfully log in as user `john`.

```
host01 login: john
Password: password
$ pwd
/home/john
$ whoami
john
```

   - You can successfully log in as user `john`.

   c. Use the `exit` command to log out as user `john`.

```
$ exit
logout
```

   - The "host01 login" prompt appears.

3. Log in as `root` and execute the `five.x` program from the `root` user's home directory.
   - Password is `oracle`.

```
host01 login: root
Password: oracle
# ./five.x
```

4. Attempt to log in as user `john` from the login prompt.
   a. Use the `exit` command to log out as user `root`.

```
# exit
logout
```

   - The "host01 login" prompt appears.

   b. Log in as the `john` user. Password is `oracle`.

```
host01 login: john
Password: password
Login incorrect
login:
```

   - Notice you cannot log in as user `john`.

5. Diagnose and fix the problem of user `john` not being able to log in.
   - Review Lesson 10, "User and Group Administration".
     - Which configuration file contains usernames? Is the user `john` present in this file?
     - Which configuration file contains user passwords? Is the user `john` present in this file?
     - If entries are missing in these files, how do you re-create the entries?
     - If passwords are corrupted or forgotten, how do you re-create passwords?
6. Verify you can successfully log in as user `john`.

Solution 21-5: User Cannot Log In

Steps

1. Diagnose and fix the problem of user `john` not being able to log in.
   a. Log in as the `root` user. Password is `oracle`.

```
host01 login: root
Password: oracle
```

   b. Use the `grep` command to search for `john` in the `/etc/passwd` file.

```
# grep john /etc/passwd
john:x:501:501::/home/john:/bin/bash
```

   - Notice the `john` entry is present.

   c. Use the `grep` command to search for `john` in the `/etc/shadow` file.

```
# grep john /etc/shadow
#john:$6$...:0:99999:7:::
```

   - Notice the `john` entry is commented out.

   d. Use the `vi` editor to modify `/etc/shadow`. Remove the `#` character from the beginning of the `john` line.

   - Because `/etc/shadow` is a read-only file, use `:wq!` to write, save, and exit `vi` after removing the `#` character.

```
# vi /etc/shadow
...
#john:$6$...:0:99999:7:::                    (old entry)
john:$6$...:0:99999:7:::                     (new entry)
```

   - Alternatively you could run the following command to re-create a password entry in the `/etc/shadow` file for user `john`.

```
# passwd john
Changing password for user john.
...
```

   e. Use the `exit` command to log out as user `root`.

```
# exit
logout
```

2. Verify you can log in as user `john`.

   a. Attempt to log in as user `john` from the log in prompt. Password is `password`.

```
host01 login: john
Password: password
$ whoami
john
```

   - Notice you can now log in as user `john`.

   b. Use the `exit` command to log out as user `john`.

```
$ exit
logout
```

Practice 21-6: File System Troubleshooting

Overview

In this practice, you diagnose and fix an unmountable file system.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Display the partition table on `/dev/xvdb`.

   Use the `fdisk -l` command to display the partition table on `/dev/xvdb`.

```
# fdisk -l /dev/xvdb
Disk /dev/xvdb: 5368 MB, 5368709120 bytes
...
    Device    Boot    Start   End    Blocks    Id    System
/dev/xvdb1                1   652   5237158+   83    Linux
```

   - Notice there is one partition on `/dev/xvdb` that consists of the entire device.

2. Make a file system on `/dev/xvdb1`.

   Use the `mkfs` command to make an **ext4** file system on `/dev/xvdb1`.

```
# mkfs -t ext4 /dev/xvdb1
mke2fs 1.43-WIP (20-Jun-2013)
Filesystem label=
OS type: Linux
...
Writing inode tables: done
Creating journal (32768 blocks): done
...
```

3. Mount the file system.

   a. Use the `mkdir` command to create `/Test`.

```
# mkdir /Test
```

   b. Use the `mount` command to mount `/dev/xvdb1` on `/Test`.

```
# mount /dev/xvdb1 /Test
```

4. Copy files to `/Test`.

   Use the `cp` command to copy `/boot/init*` to `/Test`. Use the `ls` command to display the contents of the `/Test` directory.

```
# cp /boot/init* /Test
# ls /Test
initramfs-2.6.32.431.el6.x86_64.img          lost+found
initramfs-3.8.13-16.2.1.el6uek.x86_64.img
```

5. Unmount the file system on /dev/xvdb1.

   Use the umount command to unmount /Test.

```
# umount /Test
```

6. Execute the six.x program from the root user's home directory.

```
# cd
# pwd
/root
# ./six.x
```

7. Attempt to mount the file system on /dev/xvdb1 on /Test.

   Use the mount command as shown in Task 3b.

```
# mount /dev/xvdb1 /Test
mount: you must specify the filesystem type
```

   - The mount command fails.

8. Diagnose and fix the problem with the file system.
   - Review Lesson 11, "File Systems".
     - How do you specify the file system type with the mount command?
     - What commands are available to check and repair file systems?

9. Ensure you can mount the file system on /Test.
   - Ensure the files (initramfs*) exists on /Test.

Solution 21-6: File System Troubleshooting

Steps

1. Diagnose the problem.

   a. Run the `mount` command but include the `-t ext4` option to specify the file system type.

```
# mount -t ext4 /dev/xvdb1 /Test
mount: wrong fs type, bad option, bad superblock on /dev/xvdb1,
       missing codepage or helper program, or other error
       in some cases useful info is found in syslog - try
       dmesg | tail or so
```

   b. Per the preceding suggestion, run the `dmesg | tail` command.

```
# dmesg | tail
...
EXT4-fs (xvdb1): VFS: Can't find ext4 filesystem
```

   - Notice the VFS error: "Can't find ext4 filesystem" on `xvdb1`.

   c. Use the `tail` command to view the last lines in the `/var/log/messages` file.

```
# tail /var/log/messages
...
<date_time> host01 kernel: EXT4-fs (xvdb1): VFS: Can't find ext4
filesystem
```

   - Notice the error message written to the `/var/log/messages` file is similar to the error message given by the `dmesg` command.

   d. Use the `tune2fs -l /dev/xvdb1` command to list the contents of the file system superblock.

```
# tune2fs -l /dev/xvdb1
tune2fs 1.43-WIP (20-Jun-2013)
tune2fs: Bad magic number in super-block while trying to open
/dev/xvdb1
Couldn't find valid filesystem superblock
```

   - Notice the "superblock" error.

   e. Use the `dumpe2fs /dev/xvdb1` command to dump file system information.

```
# dumpe2fs /dev/xvdb1
dumpe2fs 1.43-WIP (20-Jun-2013)
dumpe2fs: Bad magic number in super-block while trying to open
/dev/xvdb1
Couldn't find valid filesystem superblock
```

   - Notice this command reports the same "superblock" error on `/dev/xvdb1`.

2. Fix the file system.

   a. Use the `fsck /dev/xvdb1` command to check (and repair) the file system.

- Press **Enter** to accept the default **y** (yes) answer.

```
# fsck /dev/xvdb1
fsck from util-linux-ng 2.17.2
e2fsck 1.43-WIP (20-Jun-2013)
ext2fs_open2: Bad magic number in super-block
fsck.ext2: Superblock invalid, trying backup blocks...
/dev/xvdb1 was not cleanly unmounted, check forced.
Resize inode not valid. Recreate<y>? ENTER
Pass 1: Checking inodes, blocks, and sizes
...
Free blocks count wrong for group #0 ...
Fix<y>? ENTER
Free blocks count wrong for group #1 ...
Fix<y>? ENTER
Free blocks count wrong ...
Fix<y>? ENTER
Free inodes count wrong for group #0 ...
Fix<y>? ENTER
Free inodes count wrong ...
Fix<y>? ENTER
/dev/xvdb1: *****    FILE SYSTEM WAS MODIFIED *****
/dev/xvdb1: ...
```

b. Run the `fsck /dev/xvdb1` command a second time to check the file system.

```
# fsck /dev/xvdb1
fsck from util-linux-ng 2.17.2
e2fsck 1.43-WIP (20-Jun-2013)
/dev/xvdb1: clean ...
```

- Notice that the file system is fixed (clean).

3. Mount the file system.

a. Use the `mount` command to mount `/dev/xvdb1` on `/Test`.

```
# mount /dev/xvdb1 /Test
```

- The `mount` command is successful. The `fsck` command fixed the corrupt superblock.

b. Use the `ls` command to list the contents of the `/Test` directory.

```
# ls /Test
initramfs-2.6.32.431.el6.x86_64.img          lost+found
initramfs-3.8.13-16.2.1.el6uek.x86_64.img
```

- All original files are present in the directory.

Practice 21-7: Logical Volume Space is Exhausted

Overview

In this practice, you exhaust space in a logical volume and expand the file system to fix the problem.

Assumptions

You are the `root` user on the **host01** VM.

Tasks

1. Create three new partitions on `/dev/xvdd`.
   - Change the system ID on each new partition to `8e` (Linux LVM).

   a. Use the `fdisk` command to create a 1 GB **primary** partition on `/dev/xvdd` by using the following parameters:

```
# fdisk /dev/xvdd
...
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-652, default 1): ENTER
Using default value 1
Last cylinder, +cylinders or +size{K,M,G} (1-652, default 652):
+1G
```

   b. Use the "`t`" command to change the system ID on the new partition.

```
Command (m for help): t
Selected partition 1
Hex code (type L to list codes): 8e
Changed system type of partition 1 to 8e (Linux LVM)
```

   c. Continue using `fdisk` to create a second 1 GB **primary** partition on `/dev/xvdd` by using the following parameters:

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 2
First cylinder (133-652, default 133): ENTER
Using default value 133
```

```
Last cylinder, +cylinders or +size{K,M,G} (133-652, default
652): +1G
```

d. Use the "`t`" command to change the system ID on the new partition.

```
Command (m for help): t
Partition number (1-4): 2
Hex code (type L to list codes): 8e
Changed system type of partition 2 to 8e (Linux LVM)
```

e. Continue using `fdisk` to create a third 1 GB **primary** partition on `/dev/xvdd` by using the following parameters:

```
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
p
Partition number (1-4): 3
First cylinder (265-652, default 265): ENTER
Using default value 265
Last cylinder, +cylinders or +size{K,M,G} (265-652, default
652): +1G
```

f. Use the "`t`" command to change the system ID on the new partition.

```
Command (m for help): t
Partition number (1-4): 3
Hex code (type L to list codes): 8e
Changed system type of partition 3 to 8e (Linux LVM)
```

g. Print the new partition table.

```
Command (m for help): p
Disk /dev/xvdb: 5368 MB, 5368709120 bytes
...
    Device  Boot  Start     End    Blocks    Id    System
/dev/xvdd1            1     132   1060258+    8e    Linux LVM
/dev/xvdd2          133     264   1060290     8e    Linux LVM
/dev/xvdd3          265     396   1060290     8e    Linux LVM
```

h. Save the new partition table.

```
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
Syncing disks.
```

2. Initialize the new partitions for use by LVM (create physical volumes).
   a. List the partitions with the Linux LVM (**8e**) system ID.

```
# fdisk -l | grep 8e
/dev/xvdd1           1     132    1060258+    8e    Linux LVM
/dev/xvdd2         133     264    1060290     8e    Linux LVM
/dev/xvdd3         265     396    1060290     8e    Linux LVM
```

   b. Use the `pvcreate` command to create physical volumes on all three partitions.

```
# pvcreate /dev/xvdd[123]
Physical volume "/dev/xvdd1" successfully created
Physical volume "/dev/xvdd2" successfully created
Physical volume "/dev/xvdd3" successfully created
```

3. Create a volume group.
   a. Use the `vgcreate` command to create a volume group named **myvolg** from the three physical volumes.

```
# vgcreate myvolg /dev/xvdd[123]
  Volume group "myvolg" successfully created
```

   b. Use the `vgs` command to display attributes of the volume group.

```
# vgs
 VG       #PV #LV #SN Attr   VSize  VFree
 myvolg     3   0   0 wz--n- 3.02g  3.02g
```

   c. Use the `pvs` command to display information about the physical volumes.

```
# pvs
 PV         VG       Fmt   Attr PSize  PFree
 /dev/xvdd1 myvolg   lvm2  a--  1.01g  1.01g
 /dev/xvdd2 myvolg   lvm2  a--  1.01g  1.01g
 /dev/xvdd3 myvolg   lvm2  a--  1.01g  1.01g
```

4. Create a logical volume.
   a. Use the `lvcreate` command to create a 4 MB logical volume named **myvol** from the **myvolg** volume group.

```
# lvcreate -L 4m -n myvol myvolg
  Logical volume "myvol" created
```

   b. Use the `lvs` command to display attributes of the logical volume.

```
# lvs
 LV    VG     Attr   LSize Pool Origin Data% Move Log Cpy%...
 myvol myvolg -wi-a- 4.00m
```

5. Create and mount a file system on the logical volume.
   a. List the `/dev` entries for the **myvol** logical volume.

```
# ls -l /dev/myvolg/myvol
lrwxrwxrwx.  /dev/myvolg/myvol -> ../dm-0
# ls -l /dev/mapper/myvolg-myvol
lrwxrwxrwx.  /dev/mapper/myvolg-myvol -> ../dm-0
```

   b. Create an **ext4** file system on the **myvol** logical volume.

```
# mkfs.ext4 /dev/mapper/myvolg-myvol
mke2fs 1.43-WIP (20-Jun-2013)
Filesystem label=
OS type: Linux
...
Writing inode tables: done
Creating journal (1024 blocks): done
...
```

   c. Create a `/myvol` mount point.

```
# mkdir /myvol
```

   d. Mount the file system.

```
# mount /dev/mapper/myvolg-myvol /myvol
```

   e. Run the `df -h` command to display mounted file systems.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/mapper/myvolg-myvol
               2.9M   29K   2.7M    2%  /myvol
```

   f. Use the `vi` editor to add the following line to `/etc/fstab`.

```
/dev/mapper/myvolg-myvol /myvol  ext4  defaults  0  0
```

6. Exhaust space on logical volume.
   a. Use the `cd` command to change to the `/myvol` directory.

```
# cd /myvol
```

   b. Use the `dd if=/dev/zero of=bigfile bs=1024M count=5120` command to attempt to create a 5 MB file on `/myvol`.

```
# dd if=/dev/zero of=bigfile bs=1024 count=5120
dd: writing `bigfile': No space left on device
2910+0 records in
2909+0 records in
2978816 bytes (3.0 MB) copied...
```

   - Notice there is no space left on the device.

c. Use the `ls -l` command to display the contents of the `/myvol` directory.

```
# ls -l /myvol
-rw-r—r--. 1 root root 2978816 <date_time> bigfile
...
```

d. Run the `df -h` command to display mounted file systems.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/mapper/myvolg-myvol
               2.9M   2.9M    0M  100%  /myvol
```

- Notice the logical volume mounted on `/myvol` is full.

7. Determine what can be done to allocate more space to `/myvol`.
   - Review Lesson 12, “Storage Administration”.
     - What is the total size of the logical volume?
     - What volume group does the logical volume belong to?
     - What is the total size of the volume group?
     - Can you expand the logical volume by using available space in the volume group?
     - Do you need to add more space to the volume group before you can expand the logical volume?
     - What is the command that allows you to allocate more space from a volume group to a logical volume?
8. Allocate an additional 1 GB to `/myvol`
   - After allocating more space, ensure you can successfully create a 5 MB file on `/myvol`.

Solution 21-7: Logical Volume Space is Exhausted

Steps

1. Determine what is needed to increase the size of the `myvolg/myvol` logical volume and the file system.

   a. Use the `lvs` command to display attributes of the logical volume.

```
# lvs
 LV    VG     Attr     LSize  Pool Origin Data% Move Log Cpy%...
 myvol myvolg -wi-ao- 4.00m
```

   - Notice the total size of the `myvol` logical volume is 4 MB.
   - Notice the `myvol` logical volume belongs to the `mvolg` volume group.

   b. Use the `vgs` command to display attributes of the volume group.

```
# vgs
 VG       #PV #LV #SN Attr   VSize  VFree
 myvolg     3   1   0 wz--n- 3.02g  3.02g
```

   - Notice there is 3.02 GB available in the `myvolg` volume group.
   - Therefore, you can expand the logical volume by using available space in the volume group.
   - You do not need to add more space to the volume group.

2. Allocate an additional 1 GB to `/myvol`

   a. Use the `lvextend` command to increase the size of the `myvolg/myvol` logical volume and the file system by 1 GB.

   - Using the `-r` option resizes the underlying file system together with the logical volume.

```
# lvextend -L +1G -r myvolg/myvol
  Extending logical volume myvol to 1.00 GiB
  Logical volume myvol successfully resized
resize2fs 1.43-WIP (20-Jun-2013)
Filesystem at /dev/mapper/myvolg-myvol is mounted on /myvol; on-
line resizing required
old desc_blocks = 1, new_desc_blocks = 5
The file system on /dev/mapper/myvolg-myvol is now 1052672
blocks long.
```

   b. Use the `lvs` command to display attributes of the logical volume.

```
# lvs
 LV    VG     Attr     LSize  Pool Origin Data% Move Log Cpy%...
 myvol myvolg -wi-ao- 1.00g
```

   - Notice the `myvol` logical volume is now 1 GB in size.

c. Run the `df -h` command to display mounted file systems.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/mapper/myvolg-myvol
              1011M  3.0M   987M    1%  /myvol
```

- Notice there is now 987 MB of available space on `/myvol`.

3. Attempt to create the `bigfile` again.

a. Ensure the current directory is `/myvol` before running the `dd` command.

```
# cd /myvol
# dd if=/dev/zero of=bigfile bs=1024 count=5120
5120+0 records in
5120+0 records in
5242880 bytes (5.2 MB) copied...
```

- Notice you can now successfully create a 5 MB file on `/myvol`.

b. Use the `ls -l` command to display the contents of the `/myvol` directory.

```
# ls -l /myvol
-rw-r—r--. 1 root root 5242880 <date_time> bigfile
...
```

c. Run the `df -h` command to display mounted file systems.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/mapper/myvolg-myvol
              1012M  5.2M   985M    1%  /myvol
```

- Notice you still have 985 MB of space available on `/myvol`.

Practice 21-8: Network Connectivity Problem

Overview

In this practice, you diagnose and fix a network connectivity problem.

Assumptions

You are the root user on the **host01** VM.

Tasks

1. Verify network interface is configured properly.

   Use the ping command to contact **dom0** and **host03**. Use CTRL-C to abort the commands after connectivity is confirmed.

```
# ping dom0
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
CTRL-C
# ping host03
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
CTRL-C
```

2. Execute the eight.x program from the root user's home directory.

```
# cd
# pwd
/root
# ./eight.x
```

3. Repeat Task 1 to test network interface configuration.

   Use the ping command to contact **dom0** and **host03**.

```
# ping dom0
connect: Network is unreachable
# ping host03
connect: Network is unreachable
```

   - The remote systems are no longer reachable.

4. Diagnose and fix the network connectivity problem.
   - Review Lesson 13, "Network Configuration"
     - How do you display the configuration of your network?
     - Are the network interfaces up?
     - What are the IP addresses of **dom0** and **host03**?
     - Are **dom0** and **host03** on the same network as **host01**?

- Do you have a route to **dom0** and **host03**?
- How do you view the route table?
- In which files do you configure network interfaces?
- Are these network interface configuration files configured properly?
- Is the network service running?

5. Verify network connectivity to the remote hosts is working.
   a. Run the `ping dom0` command.
   b. Run the `ping host03` command.

Solution 21-8: Network Connectivity Problem

Steps

1. Diagnose the network connectivity problem.

   a. Use the `ifconfig` command to display the configuration of your network interfaces.

```
# ifconfig
eth0      Link encap:Ethernet HWaddr 00:16:3E:00:01:01
          inet addr:192.0.3.101 ...
          inet6 addr: ...
          UP ...
...

lo        Link encap:Local Loopback
          inet addr:127.0.0.1
...
```

   - Output shows that the `eth0` interface is `UP` and has an IP address of `192.0.3.101`.

   b. Use the `route` command to display the route table.

```
# route
Destination  Gateway   Genmask          Flags ...  Iface
link-local   *         255.255.0.0      U          eth0
192.0.3.0    *         255.255.255.0    U          eth0
```

   - The route table indicates the route to the `192.0.3.0` network is through `eth0`.

   c. Use the `cat` command to view the contents of the `/etc/hosts` file.

```
# cat /etc/hosts
127.0.0.1      localhost.localdomain localhost
192.0.2.1      example.com                dom0
192.0.2.101    host01.example.com         host01
192.0.2.102    host02.example.com         host02
192.0.2.103    host03.example.com         host03
```

   - The contents of the `/etc/hosts` file indicate the IP address should be configured with `192.0.2` network addresses and not `192.0.3`.

2. Fix the network connectivity problem.

   a. Use the `vi` editor to modify `/etc/sysconfig/network-scripts/ifcfg-eth0` and change the `IPADDR` directive as shown.

```
# vi /etc/sysconfig/network-scripts/ifcfg-eth0
...
IPADDR=192.0.3.101                (old value)
IPADDR=192.0.2.101                (new value)
...
```

b. Use the `service` command to restart the network service.

- Always restart the service whenever a configuration file is changed.

```
# service network restart
Shutting down interface eth0:              [  OK  ]
Shutting down loopback interface:          [  OK  ]
Bringing up loopback interface:            [  OK  ]
Bringing up interface eth0:                [  OK  ]
```

3. Verify network connectivity to the remote hosts is working.

Use the `ping` command to contact **dom0** and **host03**. Use `CTRL-C` to abort the commands after connectivity is confirmed.

```
# ping dom0
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.1): icmp_seq=1 ttl=64 ...
CTRL-C
# ping host03
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.103): icmp_seq=1 ttl=64 ...
CTRL-C
```

- Network connectivity to the remote hosts works.

Practice 21-9: NFS Permission Problem

Overview

In this practice, you diagnose and fix an NFS permission problem.

Assumptions

- You are the `root` user on the **host01** VM.
- You are the `root` user on the **host03** VM.

Tasks

Perform Task 1 from **host03**.

1. Configure **host03** as an NFS server.
   a. Use the `rpm` command to verify that the `nfs-utils` packages are installed.

```
[host03]# rpm -qa | grep nfs-utils
nfs-utils-lib-...
nfs-utils-...
```

   - In this example, the `nfs-utils` packages are installed.

   b. Use the `vi` editor and edit `/etc/exports` to add an entry to export `/home/oracle` to all client systems. Include the `rw` option.

```
[host03]# vi /etc/exports
/home/oracle *(rw)
```

   c. Use the `service` command to verify that the `rpcbind` service is started.

```
[host03]# service rpcbind status
rpcbind (pid ...) is running...
```

   - In this example, the `rpcbind` service is running.

   d. Verify that the `nfs` service is started.

```
[host03]# service nfs status
rpc.svcgssd is stopped
rpc.mountd is stopped
nfsd is stopped
rpc.rquotad is stopped
```

   - In this example, the services are not running and must be started.

   e. Start the `nfs` service and associated services.

```
[host03]# service nfs start
Starting NFS services:                    [  OK  ]
Starting NFS quotas:                      [  OK  ]
Starting NFS mountd:                      [  OK  ]
Starting NFS daemon:                      [  OK  ]
Starting RPC idmapd:                      [  OK  ]
```

f. Verify that the `nfslock` service is started.

```
[host03]# service nfslock status
rpc.statd (pid ...) is running...
```

- In this example, the `nfslock` service is running.

g. Use the `showmount` command to display exported file systems.

```
[host03]# showmount -e
Export list for host03.example.com:
/home/oracle *
```

h. Disable the firewall by running the following command.

- An `iptables` rule could be written to allow NFS connectivity.
- But for purposes of this practice, you simply stop the `iptables` service.

```
[host03]# service iptables stop
iptables: Flushing firewall rules:                        [  OK  ]
iptables: Setting chains to policy ACCEPT: filter         [  OK  ]
iptables: Unloading modules:                              [  OK  ]
```

i. Use the `ls` command to list the contents of `/home/oracle` on **host03**.

```
[host03]# ls /home/oracle
Desktop  Documents  Downloads  Music  Pictures  Public
Templates  Videos
```

Perform Tasks 2, 3, and 4 from **host01**.

2. Create a mount point in the `oracle` user's home directory on **host01**.

a. Use the `su - oracle` command to become the `oracle` user.

```
[host01]# su - oracle
```

b. As the `oracle` user, use the `mkdir` command to create a directory named `remote_home`.

```
[host01]$ mkdir remote_home
```

c. Use the `exit` command to log off as the `oracle` user.

```
[host01]$ exit
logout
```

- You are now the `root` user on **host01**.

3. Mount the exported `/home/oracle` file system on **host01**.

a. As the `root` user, use the `mount` command to mount the exported file system—`host03:/home/oracle`—on the local mount point, `/home/oracle/remote_home` with read-write permissions (`-o rw`).

```
[host01]# mount -t nfs -o rw host03:/home/oracle
/home/oracle/remote_home
```

b. Use the `df` command to verify that the NFS file system is mounted.

```
[host01]# df -h
Filesystem            Size  Used  Avail  Use%  Mounted on
...
host03:/home/oracle   1.9G   19M   1.8G    1%  /home/oracle...
```

4. Access the remote file system as the `oracle` user.

   a. From **host01**, use the `su - oracle` command to become the `oracle` user.

   - Use the `pwd` command to display the current directory.
   - Use the `ls -l` command to view the content of the current directory.

```
[host01]# su - oracle
[host01]$ pwd
/home/oracle
[host01]$ ls -l
drwx------ ... remote_home
```

   - Notice you are in the `oracle` user's home directory and that the `remote_home` directory exists.

   b. Use the `ls` command to list the contents of `remote_home`.

```
[host01]$ ls remote_home
Desktop  Documents  Downloads  Music  Pictures  Public
Templates  Videos
```

   - Notice the contents of `/home/oracle/remote_home` on **host01** are the same as `/home/oracle` on **host03** because they are the same directories.

Perform Task 5 from both **host01** and **host03**. Note the host name in the prompt.

5. Confirm you have read-write permission on the NFS file system.

   a. As the `oracle` user on **host01**, use the `vi` editor to create a "`test`" file on `remote_home` to verify you have read-write permission.

```
[host01]$ cd ~/remote_home
[host01]$ pwd
/home/oracle/remote_home
[host01]$ vi test
created from host01
```

   b. From **host03**, use the `su - oracle` command to become the `oracle` user.

   - Use the `pwd` command to display the current directory.
   - Use the `ls` command to view the content of the current directory.

```
[host03]# su - oracle
[host03]$ pwd
/home/oracle
[host01]$ ls
```

```
Desktop  Documents  Downloads  Music  Pictures  Public
Templates  test Videos
```

- Notice the presence of the "test" file that was created from **host01**.

c. As the `oracle` user on **host03**, use the `vi` editor to edit the "test" file.

```
[host03]$ vi test
created from host01
edited from host03
```

d. From **host01**, use the `cat` command to view the contents of the edited "test" file in `/home/oracle/remote_home`.

```
[host01]$ cat ~/remote_home/test
created from host01
edited from host03
```

- This confirms you have read-write permission.

Perform Tasks 6, 7, 8, and 9 from **host01**.

6. Unmount the NFS file system.

a. From **host01**, use the `exit` command to log out as the `oracle` user.

```
[host01]$ exit
logout
```

- You are now the `root` user on **host01**.

b. Use the `umount` command to unmount `/home/oracle/remote_home`.

```
[host01]# umount /home/oracle/remote_home
```

7. On **host01**, execute the `nine.x` program from the `root` user's home directory.

```
[host01]# cd
[host01]# pwd
/root
[host01]# ./nine.x
```

8. Mount the exported `/home/oracle` file system on **host01**.

a. As the `root` user, use the `mount` command to mount the exported file system—`host03:/home/oracle`—on the local mount point, `/home/oracle/remote_home` with read-write permissions (`-o rw`).

```
[host01]# mount -t nfs -o rw host03:/home/oracle
/home/oracle/remote_home
```

b. Use the `df` command to verify that the NFS file system is mounted.

```
[host01]# df -h
Filesystem              Size  Used  Avail  Use%  Mounted on
...
host03:/home/oracle     1.9G   18M   1.8G    1%  /home/oracle...
```

9. Access the remote file system as the `oracle` user.

a. Use the `su - oracle` command to become the `oracle` user.

- Use the `pwd` command to display the current directory.
- Use the `ls -l` command to view the content of the current directory.

```
[host01]# su - oracle
[host01]$ pwd
/home/oracle
[host01]$ ls -l
drwx------ ... remote_home
```

- Notice you are in the `oracle` user's home directory and that the `remote_home` directory exists.

b. Use the `ls` command to list the contents of `remote_home`.

```
[host01]$ ls remote_home
ls: cannot open directory remote_home/: Permission denied
```

- Notice you get a permission error when attempting to view the contents of the NFS file system.

c. Use the `exit` command to log off as the `oracle` user.

```
[host01]$ exit
logout
```

- You are now the `root` user on **host01**.

10. Diagnose and fix the cause of the permission error.
    - Review Lesson 14, "File Sharing".
      - Since the `mount` command succeeded, you know that:
        - Network connectivity from **host01** to **host03** exists
        - The file system is being shared by the server
        - Host name resolution (DNS) is not a problem
        - The firewall is not blocking NFS packets
      - Check the share options on the server.
      - Check the permissions on the NFS share.
      - Check the mount options on the client.
      - Check the UIDs and the GIDs of the `oracle` user on the client and the server.
      - View the `/var/log/secure` log file for messages that could indicate the cause of the problem.
11. After fixing the problem, ensure the `oracle` user from **host01** has read-permission on the `/home/oracle/remote_home/test` file.

Note: Task 12 is a clean-up task. Only perform this task after the NFS permission problem is fixed.

12. After ensuring the problem is fixed, unmount the NFS share and restore the hosts to their beginning state.

    a. As the `root` user on **host01**, use the `umount` command to unmount `/home/oracle/remote_home`.

    ```
    [host01]# umount /home/oracle/remote_home
    ```

    b. As the `root` user on **host03**, use the `vi` editor and delete all lines in `/etc/exports`.

    ```
    [host03]# vi /etc/exports
    /home/oracle *(rw)                      delete this entry
    ```

    c. As the `root` user on **host03**, use the `service` command to stop the `nfs` service.

    ```
    [host03]# service nfs stop
    Shutting down NFS daemon:                         [  OK  ]
    ...
    ```

    d. As the `root` user on **host03**, use the `service` command to start the `iptables` service.

    ```
    [host03]# service iptables start
    iptables: Applying firewall rules                 [  OK  ]
    ```

Solution 21-9: NFS Permission Problem

Steps

1. Diagnose the cause of the permission error.
   a. As the `root` user on **host03**, view the share options on the NFS file system.

```
[host03]# cat /var/lib/nfs/etab
/home/oracle
*(rw,sync,wdelay,hide,nocrossmnt,secure,root_squash,...
```

   - The NFS share has read-write (`rw`) permission, which is correct.

   b. View the permissions on the NFS file system.

```
[host03]# ls -ld /home/oracle
drwx------... oracle  oracle  ...  /home/oracle/
[host03]# ls -l /home/oracle/test
-rw-rw-r--... oracle  oracle  ...  /home/oracle/test
```

   - The `/home/oracle` directory has `rwx` permission for the owner (`oracle`), which is correct.
   - The `/home/oracle/test` file has `rw` permission for the owner (`oracle`) and the group (`oracle`), which is correct.

   c. As the `root` user on **host01**, view the `mount` options on the NFS file system.

```
[host01]# mount
...
host03:/home/oracle on /home/oracle/remote_home type nfs
(rw,vers=4,addr=192.0.2.103,clientaddr=192.0.2.101)
```

   - The NFS share is mounted with read-write (`rw`) permission, which is correct.

   d. From **host03**, use the `grep` command to view the UID and GID for `oracle` in the `/etc/passwd` and the `/etc/group` files.

```
[host03]# grep oracle /etc/passwd
oracle:x:500:500:Oracle Student:/home/oracle:/bin/bash
[host03]# grep oracle /etc/group
oracle:x:500:
```

   - Notice the `oracle` user's UID and GID are 500 on **host03**.

   e. From **host01**, use the `grep` command to view the UID and GID for `oracle` in the `/etc/passwd` and the `/etc/group` files.

```
[host01]# grep oracle /etc/passwd
oracle:x:555:555:Oracle Student:/home/oracle:/bin/bash
[host01]# grep oracle /etc/group
oracle:x:555:
```

   - Notice the `oracle` user's UID and GID are 555 on **host01**.
   - This difference in UIDs and GIDs is the cause of the permission error.

f. On **host01**, use the `tail` command to view the latest messages written to the `/var/log/secure` file.

```
[host01]# tail /var/log/secure
...
<date_time> host01 userdel[...]: delete user 'oracle'
<date_time> host01 userdel[...]: removed group 'oracle' ...
<date_time> host01 useradd[...]: new group: name=oracle, ...
<date_time> host01 useradd[...]: new user: name=oracle, ...
...
```

- Notice the `oracle` user was deleted on **host01** and re-added with UID and GID of `555`.

2. Fix the permission problem by changing `oracle` UID and GID for `oracle` on **host01** from `555` to `500`.
   - The `oracle` UID and GID on **host01** needs to be same as `oracle` user and group (500) on **host03**.
   - If possible, the easiest way is simply to delete the `oracle` user on **host01** and then re-add with the proper UID and GID.
   - The following steps are provided to make the changes without deleting and re-adding the user.

a. Before changing UIDs and GIDs, use the `umount` command to unmount the NFS file system on **host01**.

```
[host01]# umount /home/oracle/remote_home
```

b. Use the `usermod -u` command to change the UID for the `oracle` user.

```
[host01]# usermod -u 500 oracle
```

c. Use the `groupmod -g` command to change the GID for the `oracle` group.

```
[host01]# groupmod -g 500 oracle
```

d. Use the following `find` command to change the owner permissions to the new UID.

- Redirect error messages to `/dev/null`.

```
[host01]# find / -user 555 -exec chown -h 500 {} \; 2> /dev/null
```

e. Use the following `find` command to change the group permissions to the new GID.

- Redirect error messages to `/dev/null`.

```
[host01]# find / -group 555 -exec chgrp -h 500 {} \; 2>
/dev/null
```

f. Use the `usermod -g` command to change the GID for the `oracle` user.

- This command may return a "`usermod: no changes`" message, which is of no concern.

```
[host01]# usermod -g 500 oracle
```

3. On **host01**, mount the exported `/home/oracle` file system on **host01**.

   a. As the `root` user, use the `mount` command to mount the exported file system—`host03:/home/oracle`—on the local mount point, `/home/oracle/remote_home` with read-write permissions (`-o rw`).

```
[host01]# mount -t nfs -o rw host03:/home/oracle
/home/oracle/remote_home
```

   b. Use the `df` command to verify that the NFS file system is mounted.

```
[host01]# df -h
Filesystem              Size  Used  Avail  Use%  Mounted on
...
host03:/home/oracle     1.9G  5.0M   1.8G    1%  /home/oracle...
```

4. Attempt to access the remote file system as the `oracle` user.

   a. From **host01**, use the `su - oracle` command to become the `oracle` user.
      - Use the `pwd` command to display the current directory.
      - Use the `ls -l` command to view the content of the current directory.

```
[host01]# su - oracle
[host01]$ pwd
/home/oracle
[host01]$ ls -l
drwx------ ... remote_home
```

      - Notice you are in the `oracle` user's home directory and that the `remote_home` directory exists.

   b. Use the `ls` command to list the contents of `remote_home`.

```
[host01]$ ls remote_home
Desktop  Documents  Downloads  Music  Pictures  Public
Templates  test Videos
```

      - You can now view the contents of the NFS file system.

   c. Ensure you have read-permission by using the `vi` command to edit the "`test`" file.

```
[host01]$ cd ~/remote_home
[host01]$ vi test
<make some changes to the content>
```

      - You are able to successfully read and write to files on the NFS file system.

5. After ensuring the problem is fixed, unmount the NFS share and restore the hosts to their beginning state.

   a. Use the `exit` command to log off as the `oracle` user.

```
[host01]$ exit
logout
```

      - You are now the `root` user on **host01**.

b. As the root user on **host01**, use the umount command to un-mount /home/oracle/remote_home.

```
[host01]# umount /home/oracle/remote_home
```

c. As the root user on **host03**, use the vi editor and delete all lines in /etc/exports.

```
[host03]# vi /etc/exports
/home/oracle *(rw)                    delete this entry
```

d. As the root user on **host03**, use the service command to stop the nfs service.

```
[host03]# service nfs stop
Shutting down NFS daemon:                          [  OK  ]
...
```

e. As the root user on **host03**, use the service command to start the iptables service.

```
[host03]# service iptables start
iptables: Applying firewall rules                  [  OK  ]
```

Practice 21-10: Remote Access Problem

Overview

In this practice, you diagnose and fix a remote access problem.

Assumptions

- You are the `root` user on the **host01** VM.
- You are the `root` user on the **host03** VM.

Tasks

1. Ensure remote connectivity is working from **host03** to **host01**.
   a. As the `root` user on **host03**, confirm that you can use the `ssh` command to connect to **host01**.
      - Password is `oracle`.
      - Use the `hostname` command to confirm that you did connect.

```
[host03]# ssh host01
root@host01's password: oracle
Last login: ...
[host01]# hostname
host01.example.com
```

   b. Use the `logout` command to log off.

```
[host01]# logout
Connection to host01 closed.
```

2. Ensure remote connectivity is working from **host01** to **host03**.
   a. As the `root` user on **host01**, confirm that you can use the `ssh` command to connect to **host03**.
      - Password is `oracle`.
      - Use the `hostname` command to confirm that you did connect.

```
[host01]# ssh host03
root@host03's password: oracle
Last login: ...
[host03]# hostname
host03.example.com
```

   b. Use the `logout` command to log off.

```
[host03]# logout
Connection to host03 closed.
```

3. From **host01**, execute the `ten.x` program from the `root` user's home directory.

```
[host01]# cd
[host01]# pwd
```

```
/root
[host01]# ./ten.x
```

4. Ensure remote connectivity is working from **host01** to **host03**.
   a. As the `root` user on **host01**, confirm that you can use the `ssh` command to connect to **host03**.

```
[host01]# ssh host03
root@host03's password: oracle
Last login: ...
[host03]# hostname
host03.example.com
```

   b. Use the `logout` command to log off.

```
[host03]# logout
Connection to host03 closed.
```

5. Ensure remote connectivity is working from **host03** to **host01**.

   As the `root` user on **host03**, confirm that you can use the `ssh` command to connect to **host01**.

```
[host03]# ssh host01
ssh: connect to host host01 port 22: No route to host
```

   - Notice the `ssh` command fails.

6. Diagnose and fix the remote connectivity problem.
   - Think about what you can and what you cannot do:
     - You can `ssh` from **host01** to **host03**
     - You cannot `ssh` from **host03** to **host01**
     - Therefore, the problem is most likely a configuration issue on **host01**.
   - Can you `ping` **host01**? If so, then there is nothing wrong with the network interface configuration.
   - Review Lesson 15, "OpenSSH".
     - Ensure the `sshd` service is configured properly and that the service is running.
   - Notice the error message, "`ssh: connect to host host01 port 22: No route to host`."
     - View the `/var/log/messages` log file for entries related to `ssh` or port `22`.
     - View the `/var/log/secure` log file for entries related to `ssh` or port `22`.
   - Review Lesson 16, "Pluggable Authentication Modules (PAM)".
     - Are there any PAM modules that are causing the problem with `sshd`?
   - Review Lesson 17, "Security Administration".
     - Is the firewall prohibiting a connection on port `22`?
     - Is there a TCP wrapper configured that is causing the problem?

7. After fixing the problem, ensure you can `ssh` from **host03** to **host01**.

Solution 21-10: Remote Access Problem

Steps

1. Attempt to diagnose the problem by testing network connectivity.

   From **host03**, use the `ping host01` command to test network connectivity to **host01**.

```
[host03]# ping host01
64 bytes from example.com (192.0.2.101): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.101): icmp_seq=1 ttl=64 ...
64 bytes from example.com (192.0.2.101): icmp_seq=1 ttl=64 ...
CTRL-C
```

   - Notice the `ping` command verifies network connectivity to **host01** exists.

2. Attempt to diagnose the problem by checking `sshd` configuration.

   a. From **host01**, use the `service` command to view the status of the `sshd` service.

```
[host01]# service sshd status
openssh-daemon (pid ...) is running...
```

   - The `sshd` service is running.

   b. From **host03**, use the `service` command to view the status of the `sshd` service.

```
[host03]# service sshd status
openssh-daemon (pid ...) is running...
```

   - The `sshd` service is running.

   c. Use the `grep` command to search for "`ssh`" in the `/var/log/messages*` files.

   - Repeat the command but search for "`port 22`".

```
[host01]# grep ssh /var/log/messages*
[host01]# grep "port 22" /var/log/messages*
```

   - Notice that neither command returned any output.

   d. Use the `grep` command to search for "`ssh`" in the `/var/log/secure*` files.

   - Repeat the command but search for "`port 22`".

```
[host01]# grep ssh /var/log/secure*
...
[host01]# grep "port 22" /var/log/secure*
...
```

   - Both commands returned output but the <*date_time*> stamp on the log file entries does not correspond to the <*date_time*> of the problem.

3. Attempt to diagnose the problem by checking PAM configuration.
   - It is unlikely the problem is caused by PAM.
   - If PAM were the cause of the problem, a log file entry would usually be written to the `/var/log/secure` file.
   - However, some of the steps used to investigate PAM configuration follow.

a. Use the `ls` command to determine if there are any PAM modules related to `ssh`.

```
[host01]# ls /etc/pam.d/*ssh*
/etc/pam.d/sshd              /etc/pam.d/ssh-keycat
```

- Notice there are two PAM configuration files that are related to `ssh`.

b. Use the `cat` command to view the contents of the `/etc/pam.d/sshd` file.

```
[host01]# cat /etc/pam.d/sshd
#%PAM-1.0
auth       required    pam_sepermit.so
auth       include     password-auth
...
```

- Notice the first PAM module is `pam_sepermit.so`.

c. View the man pages for the `pam_sepermit` authentication module.

```
[host01]# man pam_sepermit
...
    pam_sepermit - PAM module to allow/deny login depending
    On SELinux enforcement state
...
```

- Notice that when SELinux is in enforcing mode, the `pam_sepermit` authentication module allows or denies login.

d. SELinux stands for "Security-Enhanced Linux" and is covered in a later course but for the purposes of this practice, use the `sestatus` command to display information about SELinux.

```
[host01]# sestatus
SELinux status:    enabled
...
Current mode:      enforcing
...
```

- Notice that SELinux is enabled and is in enforcing mode.

e. Use the `setenforce 0` command to temporarily change SELinux to "permissive" mode.

- Temporarily setting SELinux to "permissive" mode and re-testing is a good practice to perform to ensure SELinux is not the cause of a problem with a service.
- Use the `sestatus` command to display the change in the SELinux mode.

```
[host01]# setenforce 0
[host01]# sestatus
SELinux status:    enabled
...
Current mode:      permissive
...
```

- Notice the "Current mode" is now set to "permissive."

f. Test if setting SELinux to permissive mode allows the `ssh` connection from **host03** to **host01**.

```
[host03]# ssh host01
ssh: connect to host host01 port 22: No route to host
```

- Notice the `ssh` command still fails. Therefore SELinux is not the cause of the problem.

g. Use the `setenforce 1` command to change SELinux back to “enforcing” mode.

- Use the `sestatus` command to display the change in the SELinux mode.

```
[host01]# setenforce 1
[host01]# sestatus
SELinux status:    enabled
...
Current mode:      enforcing
...
```

- Notice the “Current mode” is now set to “enforcing.”

You could continue viewing man pages for the other PAM modules listed in the `/etc/pam.d/sshd` and `/etc/pam.d/ssh-keycat` files to determine if any of these are the cause of the problem. However, as was previously stated, entries are usually written to the `/var/log/secure` log file when PAM is denying access.

4. Attempt to diagnose the problem by checking firewall (`iptables`) configuration.

a. Use the `service` command to view the status of the `iptables` service.

```
[host01]# service iptables status
...
```

- Notice that the `iptables` service is running.

b. The `ssh` service uses port `22`, which can be confirmed by using the `grep ssh /etc/services` command.

```
[host01]# grep ssh /etc/services
ssh        22/tcp     # The Secure Shell (SSH) Protocol
ssh        22/udp     # The Secure Shell (SSH) Protocol
...
```

c. View the status of the `iptables` service but pipe the output to `grep` and search for `22`.

```
[host01]# services iptables status | grep 22
```

- No output indicates there is no `iptables` rule for this port number.

5. Stop the `iptables` service on **host01** and re-test remote connectivity.

a. Use the `service` command to stop the `iptables` service.

```
[host01]# service iptables stop
...
```

b. From **host03**, use the `ssh` command to attempt to connect to **host01**.

- Password is `oracle`.
- Use the `hostname` command to confirm that you did connect.

```
[host03]# ssh host01
root@host01's password: oracle
Last login: ...
[host01]# hostname
host01.example.com
```

- Connection is successful with the `iptables` service stopped on **host01**.

c. Use the `logout` command to log off.

```
[host01]# logout
Connection to host01 closed.
```

6. Fix the remote connectivity problem by creating an `iptables` rule on **host01** to trust `ssh`.
   - Stopping `iptables` on **host01** allows the `ssh` connection.
   - However, to keep this service running you need to create a rule that trusts the `ssh` service on port # `22`.

a. Use the `service` command to start the `iptables` service.

```
[host01]# service iptables start
iptables: Applying firewall rules
```

b. Use the `iptables` command to list rules in the `INPUT` chain and include line numbers.

```
[host01]# iptables -L INPUT --line-numbers
Chain INPUT (policy ACCEPT)
num  target   prot  opt  source     destination
1    ACCEPT   all   --   anywhere   anywhere   state ...
2    ACCEPT   icmp  --   anywhere   anywhere
3    ACCEPT   all   --   anywhere   anywhere
4    REJECT   all   --   anywhere   anywhere   reject-with...
```

- You want to insert the new rule before the `REJECT` entry on line 4.

c. Use the `iptables` command to insert the "`ssh`" rule before line 4 with the following characteristics from the command line:

- Chain = `INPUT`
- Protocol = `tcp`
- State = `NEW`
- Destination port = `ssh`
- Target = `ACCEPT`

```
[host01]# iptables -I INPUT 4 -p tcp -m state --state NEW --
dport ssh -j ACCEPT
```

- This rule accepts incoming `tcp` traffic for `ssh`.

d. Use the `iptables` command to list rules in the `INPUT` chain and include line numbers.

```
[host01]# iptables –L INPUT --line-numbers
Chain INPUT (policy ACCEPT)
num  target   prot  opt  source     destination
1    ACCEPT   all   --   anywhere   anywhere   state ...
2    ACCEPT   icmp  --   anywhere   anywhere
3    ACCEPT   all   --   anywhere   anywhere
4    ACCEPT   tcp   --   anywhere   anywhere   state NEW tcp
dpt:ssh
5    REJECT   all   --   anywhere   anywhere   reject-with...
```

- Notice the new line 4 rule that accepts `tcp` traffic when the destination port is `ssh`.

e. Use the `service` command to save the `iptables` rules.

- After making any rule changes, always save the `iptables` rules.
- Saving `iptables` rules updates the `/etc/sysconfig/iptables` file.

```
[host01]# service iptables save
iptables: Saving firewall rules to /etc/sysconfig/iptables...
```

f. Use the `cat` command to view the `/etc/sysconfig/iptables` file.

```
[host01]# cat /etc/sysconfig/iptables
...
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 –j ACCEPT
...
```

- Notice that in the configuration file for `iptables`, the new rule has the actual port number, `22`, for `ssh`.

g. Use the `service` command to restart the `iptables` service.

- After making any rule changes, restart the `iptables` service.

```
[host03]# service iptables restart
iptables: Setting chains to policy ACCEPT: filter   [  OK  ]
iptables: Flushing firewall rules:                  [  OK  ]
iptables: Unloading modules:                        [  OK  ]
iptables: Applying firewall rules:                  [  OK  ]
```

7. Ensure remote connectivity is working.

a. From **host03**, confirm that you can use the `ssh` command to connect to **host01**.

- Password is `oracle`.
- Use the `hostname` command to confirm that you did connect.

```
[host03]# ssh host01
root@host01's password: oracle
Last login: ...
[host01]# hostname
host01.example.com
```

b. Use the `logout` command to log off.

```
[host01]# logout
Connection to host01 closed.
```

Practice 21-11: Log File Is Not Getting Updated

Overview

In this practice, you diagnose and fix a problem with a log file not getting updated.

Assumptions

- You are the `root` user on the **host01** VM.
- You are the `root` user on the **host03** VM.

Tasks

1. From **host01**, set the `oracle` user's password to `oracle`.

   Use the `passwd oracle` command to change the password.

   - Ignore the "BAD PASSWORD" messages.

```
[host01]# passwd oracle
Changing password for user oracle.
New password: oracle
BAS PASSWORD: it is based on a dictionary word
BAS PASSWORD: is too simple
Retype new password: oracle
passwd: all authentication tokens updated successfully.
```

2. From **host03**, confirm you can use the `ssh` command to log in to **host01** as the `oracle` user.

   a. Use the `exit` command to log out as the `root` user.

   - Use the `whoami` command to confirm you are the `oracle` user.

```
[host03]# exit
logout
[host03]$ whoami
oracle
```

   b. Use the `ssh` command to log in to **host01**.

   - Password is `oracle`.
   - Use the `hostname` command to verify you were able to log in.

```
[host03]$ ssh host01
oracle@host01's password: oracle
Last login: ...
[host01]$ hostname
host01.example.com
```

c. Use the `exit` command to log off **host01**.

```
[host01]$ exit
logout
Connection to host01 closed.
```

3. From **host01**, save a copy of the `/var/log/secure` file.

   Use the `cp` command to copy `/var/log/secure` to `~/secure_before`.

```
[host01]# cp /var/log/secure ~/secure_before
```

- You review the differences in these copies later in this practice.

4. From **host01**, configure the `pam_nologin` authentication module to prevent non-`root` login.
   - This exercise was performed in the PAM lesson, practice 16-2.

   Use the `vi` editor and create the `/etc/nologin` file with the following contents:

```
[host01]# vi /etc/nologin
No logins allowed at this time.
```

5. As the `oracle` user on **host03**, attempt to log in to **host01**.

   Use the `ssh` command to connect to **host01**.

   - Password is `oracle`.

```
[host03]$ ssh host01
oracle@host01's password: oracle
No logins allowed at this time.
Connection closed by 192.0.2.101
```

- Notice that the connection is denied by the PAM authentication module.

6. From **host01**, view the new entries in the `/var/log/secure` log file.

   Use the `diff` command to view the differences in the `/var/log/secure` log file and the copy of the log file you made in Task 3.

```
[host01]# diff /var/log/secure ~/secure_before
< <date_time> host01 [sshd[...]: Failed password for oracle from
192.0.2.103 port ... ssh2
< <date_time> host01 [sshd[...]: fatal: Access denied for user
oracle by PAM account configuration
```

- Notice that two entries were written to this file when the `oracle` user attempted to `ssh` from **host03** to **host01**.
- Also notice that one entry specifically references PAM account configuration.

7. From **host01**, execute the `eleven.x` program from the `root` user's home directory.

```
[host01]# cd
[host01]# pwd
/root
[host01]# ./eleven.x
```

8. From **host01**, save a copy of the /var/log/secure file.

   Use the cp command to copy /var/log/secure to ~/secure_before.

   - Answer **y** to overwrite the file.

```
[host01]# cp /var/log/secure ~/secure_before
cp: overwrite '/root/secure_before'? y
```

   - You review the differences in these copies later in this practice.

9. As the oracle user on **host03**, attempt to log in to **host01**.

   Use the ssh command to connect to **host01**. Password is oracle.

```
[host03]$ ssh host01
oracle@host01's password: oracle
No logins allowed at this time.
Connection closed by 192.0.2.101
```

   - Notice that the connection is denied by the PAM authentication module.

10. From **host01**, view the new entries in the /var/log/secure log file.

    Use the diff command to view the differences in the /var/log/secure log file and the copy of the log file you made in Task 8.

```
[host01]# diff /var/log/secure ~/secure_before
```

    - Notice that there are no differences in the files.
    - The /var/log/secure log file is not getting updated as expected.

11. Diagnose and fix the problem of the log file not getting updated.
    - Review Lesson 20, "System Logging".
      - Is the logging daemon running?
      - Is logging configured for the /var/log/secure log file?

12. After fixing the problem, ensure the /var/log/secure log file is getting updated when you attempt to ssh from **host03** to **host01** as the oracle user.

13. Return **host01** back to the original state.

    Use the rm command to remove the /etc/nologin file.

```
# rm /etc/nologin
rm: remove regular file '/etc/nologin'? y
```

Solution 21-11: Log File Is Not Getting Updated

Steps

1. Diagnose the cause of the logging problem.

   a. As the `root` user on **host01**, view the status of the `rsyslog` service.

```
[host01]# service rsyslog status
rsyslogd (pid ...) is running...
```

   - The `rsyslogd` service is running.

   b. Use the `grep` command to search for `/var/log/secure` in the `rsyslog` configuration file.

```
[host01]# grep /var/log/secure /etc/rsyslog.conf
#authpriv.*                                 /var/log/secure
```

   - Notice the configuration file contains an entry for `/var/log/secure`; however, the entry is commented out (proceeded by a `#` sign).

2. Fix the problem.

   a. Use the `vi` editor to remove the `#` from the beginning of the line containing `/var/log/secure`.

```
[host01]# vi /etc/rsyslog.conf
...
#authpriv.*           /var/log/secure              (old entry)
authpriv.*            /var/log/secure              (new entry)
...
```

   b. Use the `service` command to restart the `rsyslog` service.

```
[host01]# service rsyslog restart
Shutting down system logger:                       [  OK  ]
Starting system logger:                            [  OK  ]
```

3. Verify the `/var/log/secure` log file is getting updated.

   a. From **host01**, use the `cp` command to copy `/var/log/secure` to `~/secure_before`.

   - Answer **y** to overwrite the file.

```
[host01]# cp /var/log/secure ~/secure_before
cp: overwrite '/root/secure_before'? y
```

   b. As the `oracle` user on **host03**, use the `ssh` command to connect to **host01**.

   - Password is `oracle`.

```
[host03]$ ssh host01
oracle@host01's password: oracle
No logins allowed at this time.
Connection closed by 192.0.2.101
```

- Notice that the connection is denied by the PAM authentication module.

c. From **host01**, use the `diff` command to view the differences in the `/var/log/secure` log file and the copy, `~/secure_before`.

```
[host01]# diff /var/log/secure ~/secure_before
< <date_time> host01 [sshd[...]: Failed password for oracle from
192.0.2.103 port ... ssh2
< <date_time> host01 [sshd[...]: fatal: Access denied for user
oracle by PAM account configuration
```

- Notice that entries are now being written to the `/var/log/secure` file as expected.

4. Return **host01** back to the original state.

Use the `rm` command to remove the `/etc/nologin` file.

```
# rm /etc/nologin
rm: remove regular file '/etc/nologin'? y
```

Appendix A: Source Code for Problem-Causing Executables

Overview

In this appendix, the source code for the executables that cause the problems are given.

Practice 21-2: System Boots into Single-User Mode

The “`two.x`” script appends the word `single` to the end of the kernel line for the default kernel using the following commands:

```
#!/bin/bash
cp /boot/grub/grub.conf /boot/grub/grub.conf.save
sed –i –e ‘/vmlinuz-3.8.13/s/$/ single/’ /boot/grub/grub.conf
```

This result of this program follows:

```
# cat /boot/grub/grub.conf
...
default=0
...
title Oracle Linux Server Unbreakable Kernel (3.8.13-16...
        root (hd0,0)
        kernel /vmlinuz-3.8.13-16... single
...
```

Practice 21-3: Status Commands Fail

The “`three.x`” script runs the following `umount` command to unmount the `proc` file system.

```
#!/bin/bash
umount –l /proc
```

Practice 21-4: `cron` Job Fails to Run

The “`four.x`” script stops the `crond` service by running the following command:

```
#!/bin/bash
service crond stop > /dev/null
```

Practice 21-5: User Cannot Log In

The “`five.x`” script inserts the “`#`” sign at the beginning of the `john` line in the `/etc/shadow` file by running the following commands:

```
#!/bin/bash
cp /etc/shadow /etc/shadow.save
sed –i –e ‘/john/s/^/#/’ /etc/shadow
```

The result of this program follows:

```
# cat /etc/shadow
...
#john:$6$...:0:99999:7:::
```

Practice 21-6: File System Troubleshooting

The "`six.x`" script runs the following `dd` command to corrupt the file system superblock.

```
#!/bin/bash
dd if=/dev/zero of=/dev/xvdb bs=1024 skip=1000 count=300 2>
/dev/null
```

Practice 21-8: Network Connectivity Problem

The "`eight.x`" script changes the IP address for **host01** from `192.0.2.101` to `192.0.3.101` by running the following commands:

```
#!/bin/bash
cd /etc/sysconfig/network-scripts
/bin/cp ifcfg-eth0 ~/ifcfg-eth0.save
sed -e 's/2.101/3.101/' ifcfg-eth0 > ifcfg-eth0.new
/bin/mv ifcfg-eth0.new ifcfg-eth0
service network restart > /dev/null
cd
```

The result of this program follows:

```
# cat /etc/sysconfig/network-scripts/ifcfg-eth0
...
IPADDR=192.0.3.101
...
# ifconfig eth0
eth0      Link encap: Ethernet  HWaddr: 00:16:3E:00:01:01
          inet addr: 192.0.3.101 ...
...
```

Practice 21-9: NFS Permission Problem

The "`nine.x`" script deletes the `oracle` user, re-adds the `oracle` user with UID=`555`, and re-creates the `/home/oracle/remote_home` directory by running the following commands:

```
#!/bin/bash
userdel -r oracle
useradd -u 555 oracle
mkdir /home/oracle/remote_home
chown -R oracle.oracle /home/oracle
```

The result of this program follows:

```
# grep oracle /etc/passwd
```

```
oracle:x:555:555::/home/oracle:/bin/bash
# ls -l /home/oracle
drwxr-xr-x. 2 oracle oracle 4096 <date_time> remote_home
```

Practice 21-10: Remote Access Problem

The “ten.x” script removes the iptables rule on **host01**, which trusts the sshd service and restarts the iptables service by running the following commands:

```
#!/bin/bash
cp /etc/sysconfig/iptables /etc/sysconfig/iptables.save
sed –i '/22/d' /etc/sysconfig/iptables
service iptables restart > /dev/null
```

Practice 21-11: Log File Is Not Getting Updated

The “eleven.x” script inserts the “#” sign at the beginning of the secure line in the /etc/rsyslog.conf file and restarts the rsyslog service by running the following commands:

```
#!/bin/bash
cp /etc/rsyslog.conf /etc/rsyslog.conf.save
sed –i –e '/secure/s/^/#/' /etc/rsyslog.conf
service rsyslog restart > /dev/null
```

The result of this program follows:

```
# cat /etc/rsyslog.conf
...
#authpriv.*                              /var/log/secure
```

Appendixes: Remote Access Options

Chapter 22

Appendix

Appendixes Overview

The three appendixes show various options for accessing your student PC remotely:

- Appendix A: Using an NX Client to Connect to **dom0**
- Appendix B: Using an NX Player to Connect to **dom0**
- Appendix C: Using VNC (TightVNC) to Connect Directly to VM Guests

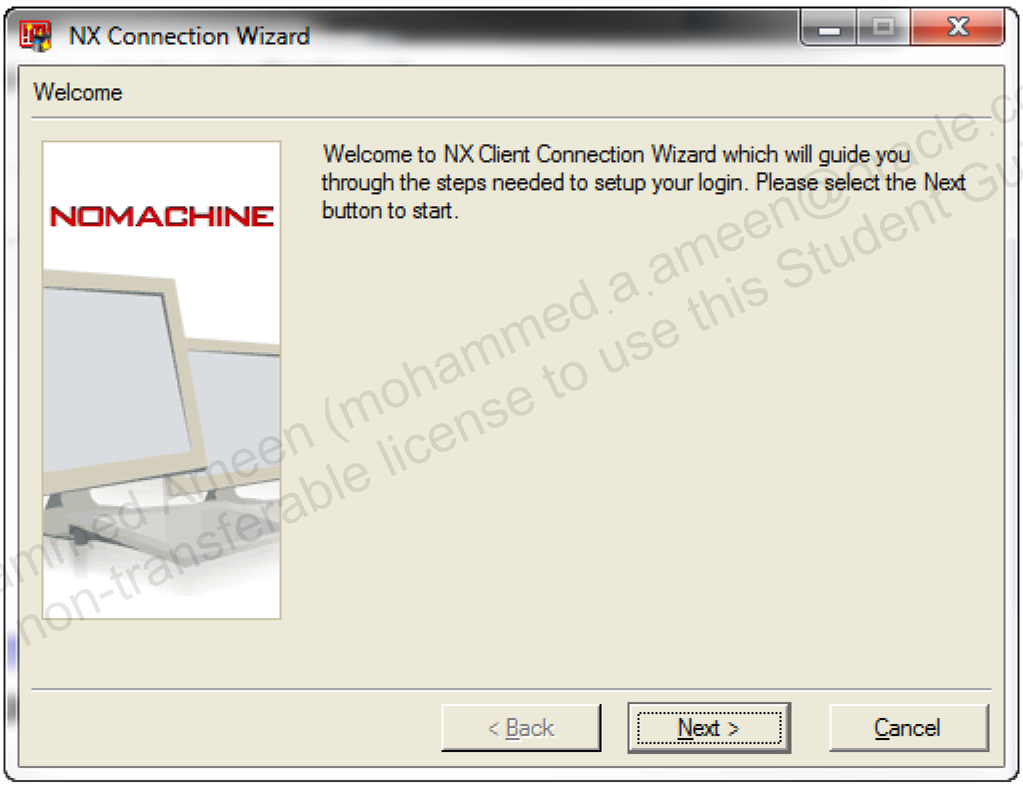
Appendix A: Using an NX Client to Connect to dom0

Overview

This appendix discusses accessing your student PC (**dom0**) remotely by using NX Client. The NX Client in this appendix is NX Client for Windows, Version 3.5.0-9.

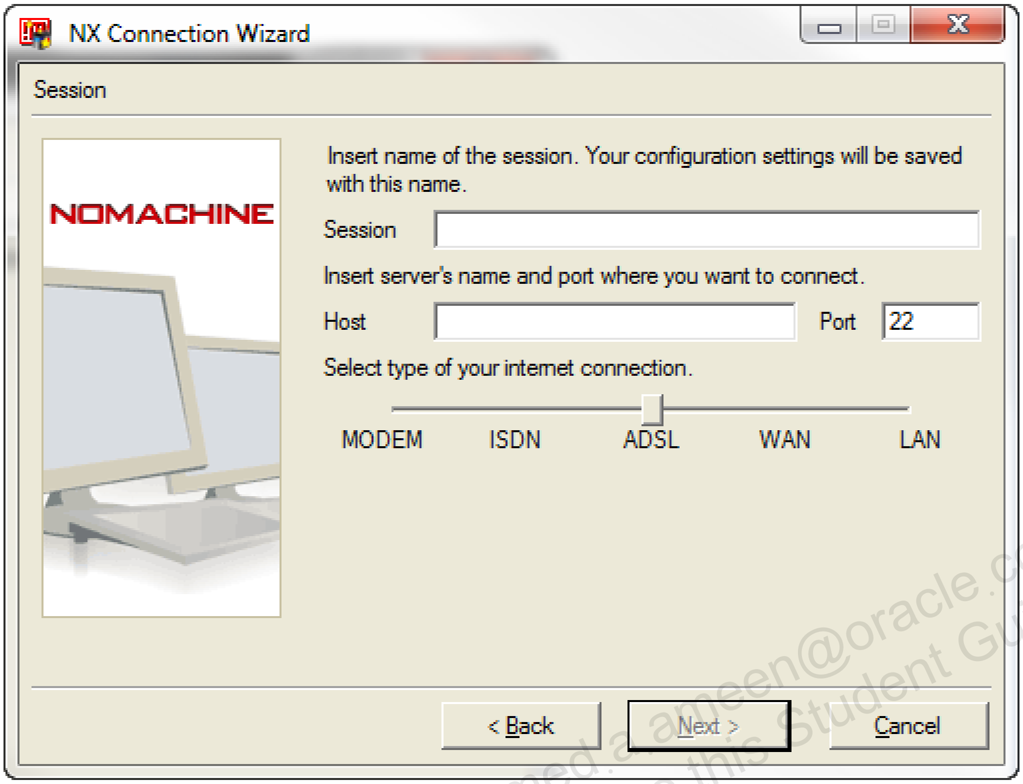
Steps

1. Install **NX Client** (if necessary) from http://www.nomachine.com/download.php.
2. Run **NX Client** (for example, select **NX Client for Windows** from the Windows Start menu).
    - An NX Connection Wizard steps you through creating the initial session.
    - The following Welcome window appears.



    a. Click **Next**.

- The following Session window appears.



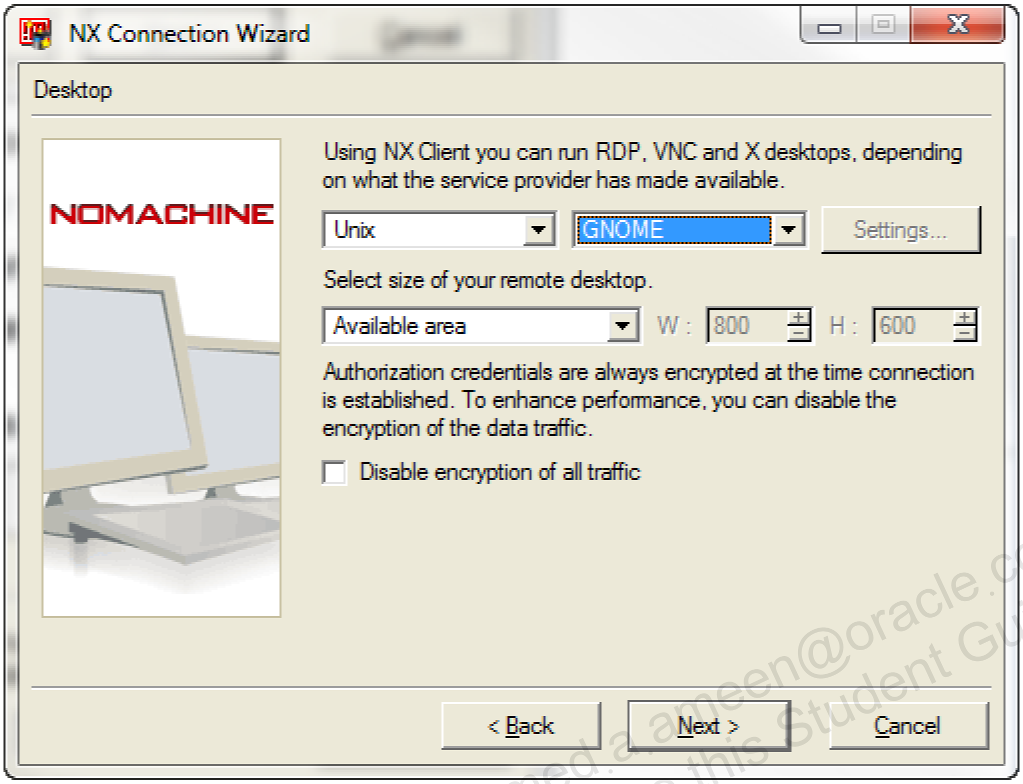
b. Enter anything you like for **Session** (for example, `OracleLinux`).

c. Enter the IP address (provided by your instructor) for **Host**.

d. Accept the remaining defaults and click **Next**.

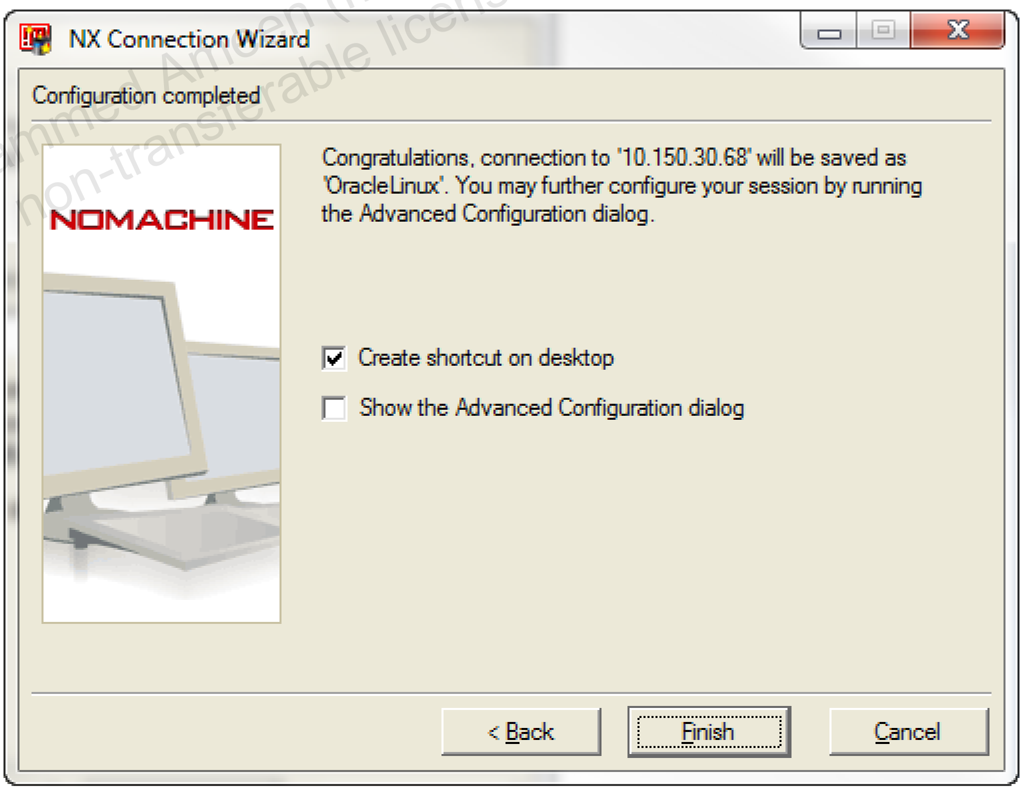
e. The Desktop window appears. Change KDE to **GNOME** by selecting from the drop-down list.

- Your window should look like the following:



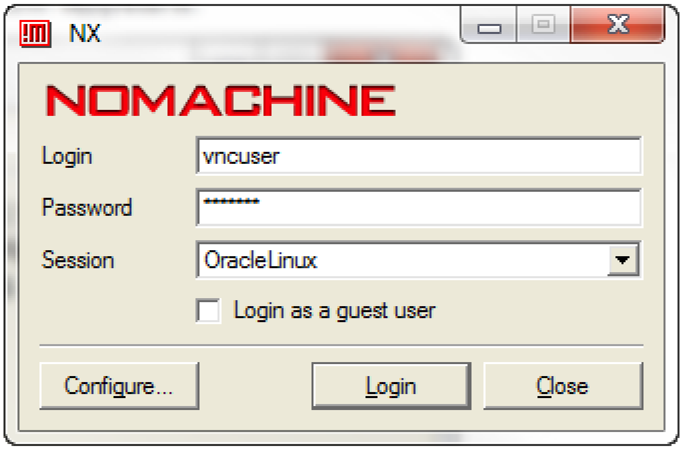
f. Accept all other defaults and click **Next**.

- The following Configuration completed window appears.



g. Click **Finish**.

- The NX Login window appears.



h. For **Login**, enter `vncuser`.

i. For **Password**, enter `vnctech`.

j. Your **Session** defaults to the session that you just created. In this example, the **Session** is OracleLinux. Your session name may be different.

k. Click **Login**.

- The **dom0** GNOME virtual desktop window appears.
- Future connections will bypass the configuration wizard and only bring up the NX Login window.

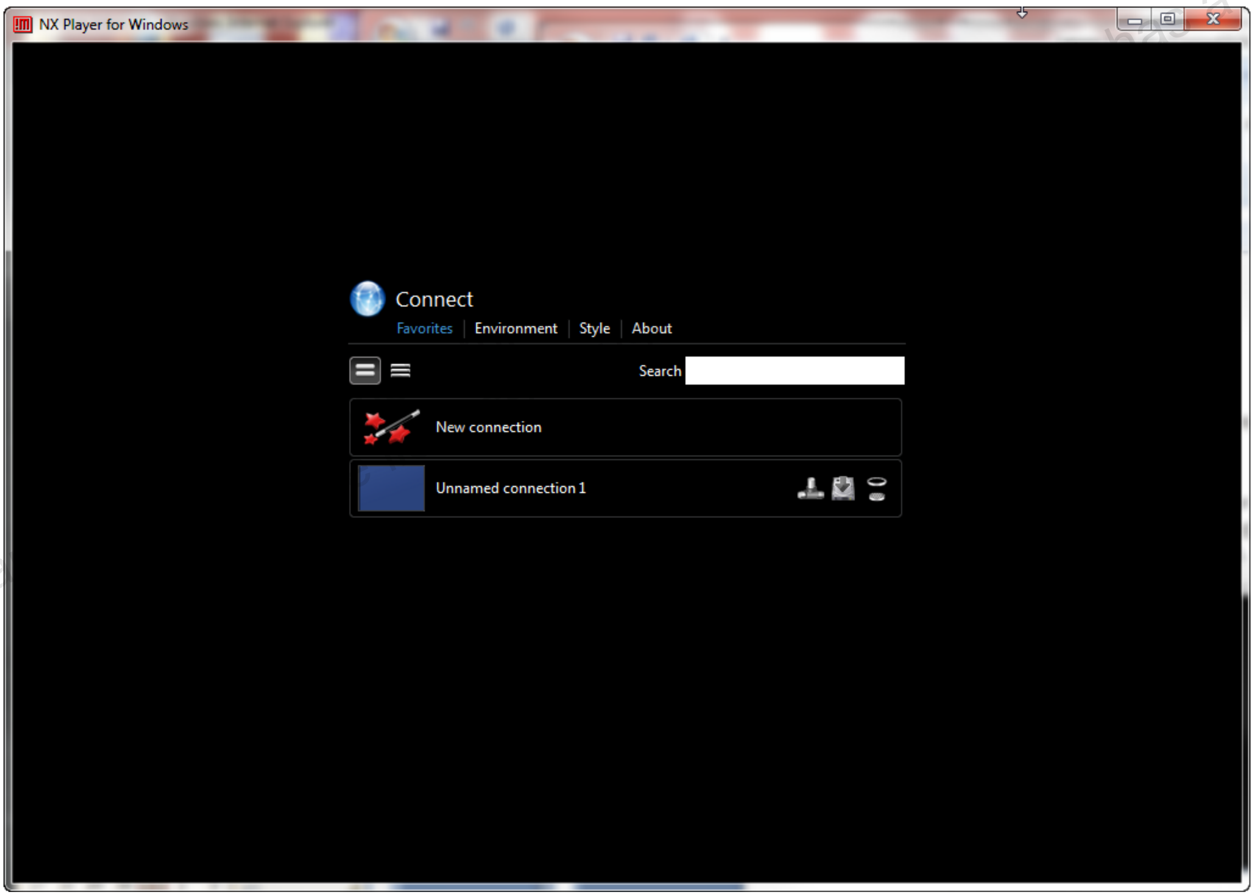
Appendix B: Using an NX Player to Connect to dom0

Overview

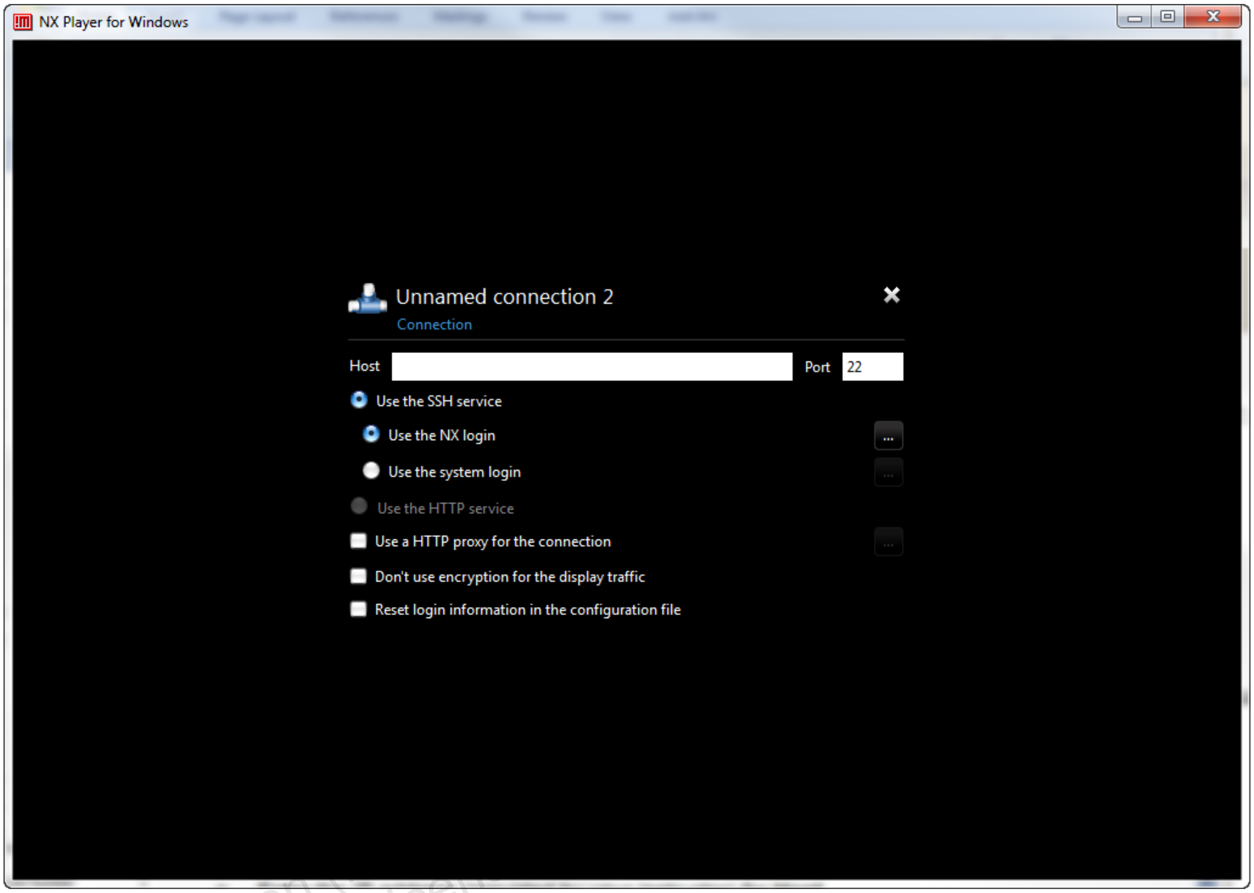
This appendix discusses accessing your student PC (**dom0**) remotely using NX Player. The NX Player in this appendix is NX Player for Windows, Preview 5, version 4.0.132.

Steps

1. Install **NX Player** (if necessary) from http://www.nomachine.com/download.php.
2. Run **NX Player** (for example, select **NX Player for Windows** from the Windows Start menu).
   a. Ensure that the **Favorites** tab is selected.
      - The Connect window appears.



b. Click **New connection** to display the following window.



c. Enter the IP address (provided by your instructor) for **Host**.

d. Accept the defaults:

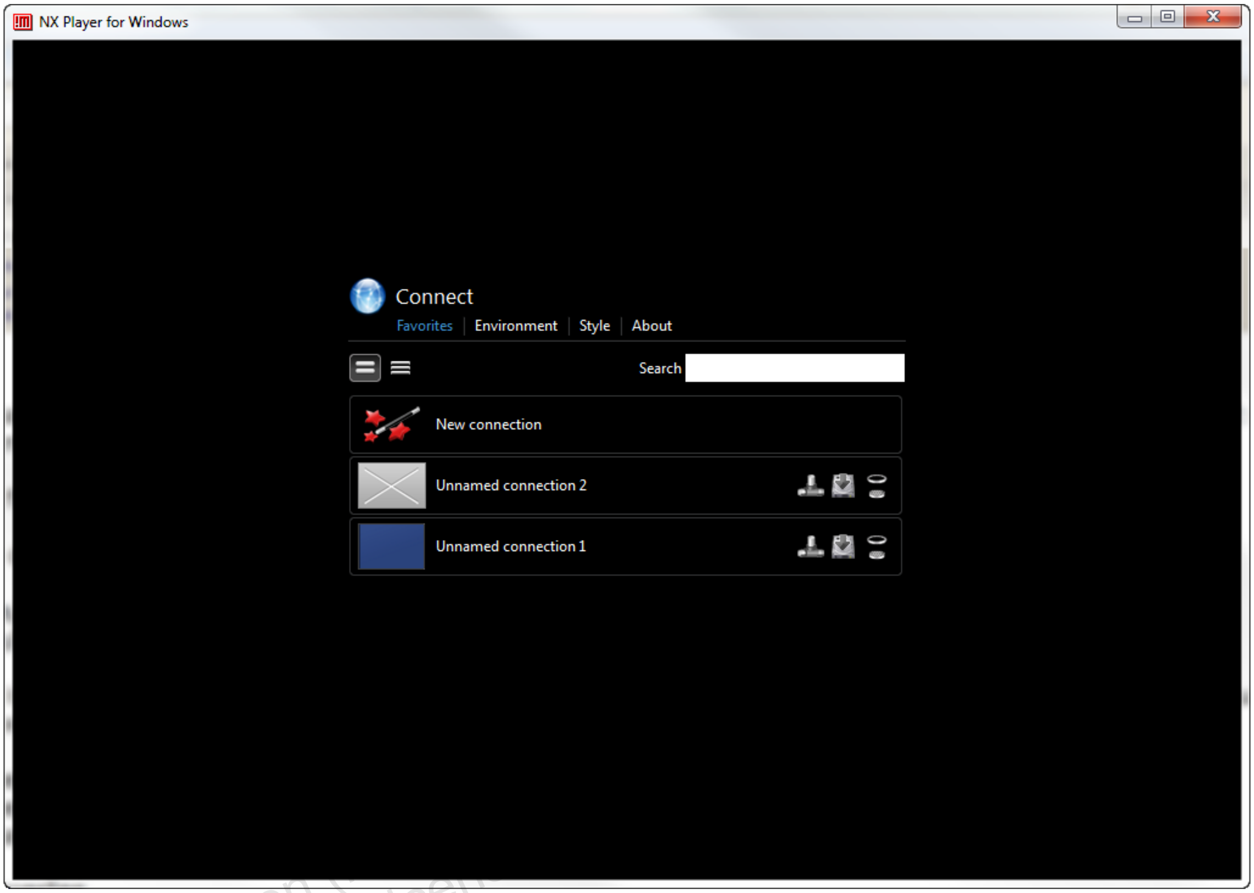
1) Port 22

2) Use the SSH service

3) Use the NX login

e. Note the connection name. In this example it is **Unnamed connection 2**. Yours is most likely **Unnamed connect 1**.

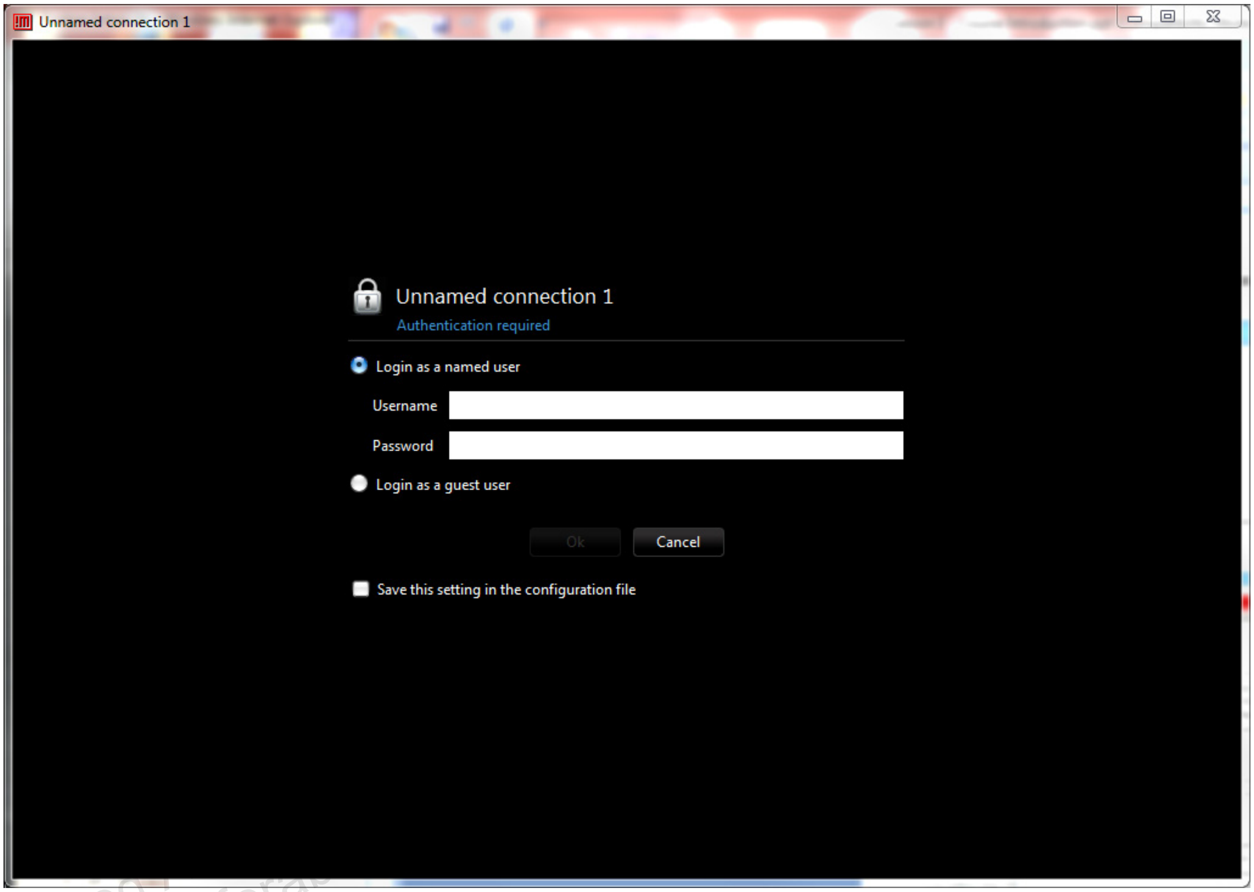
f. Press **Enter**. The following window appears.



g. Click the connection that you just created (**Unnamed connection 1**, for example).

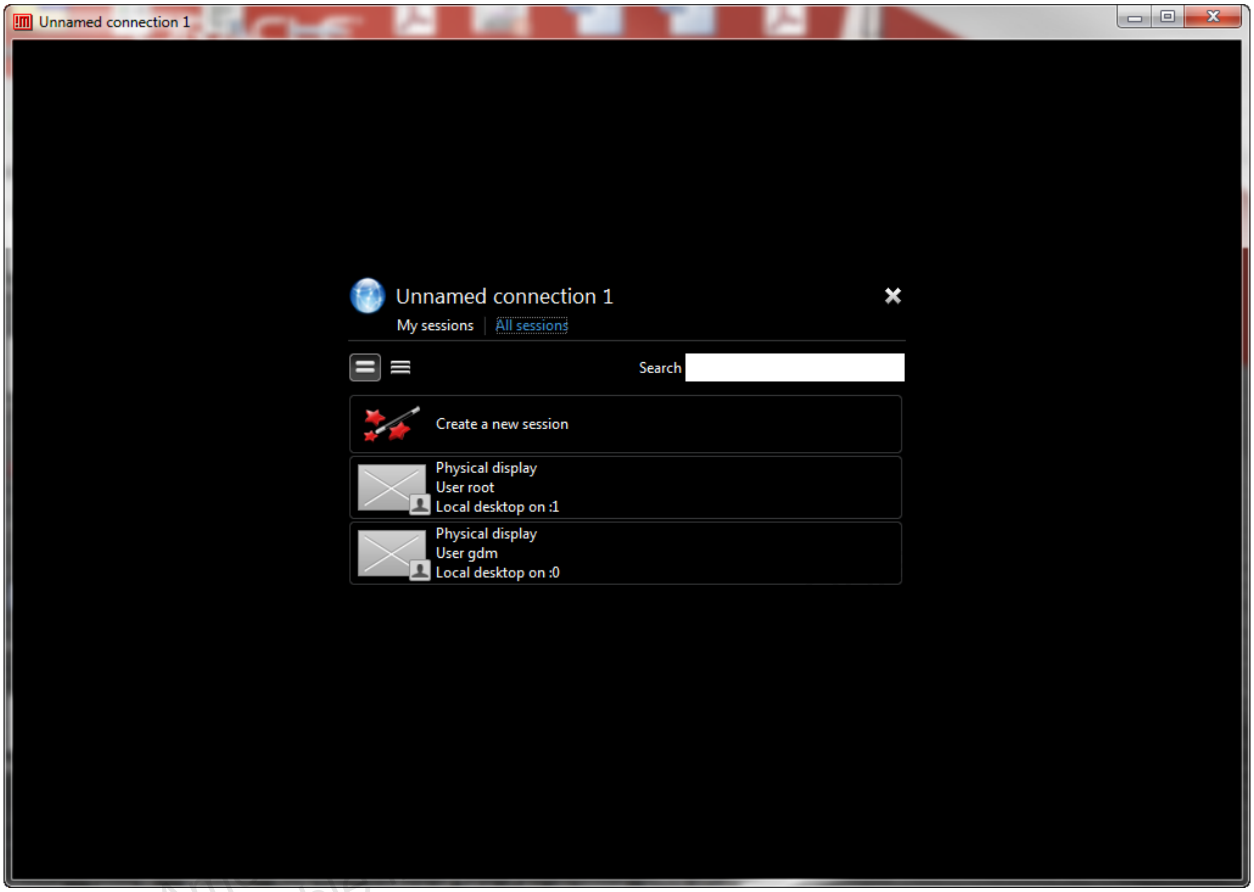
- The Login window appears.

3. Log in.
   - The window shown in the following screenshot appears, prompting for login authentication.



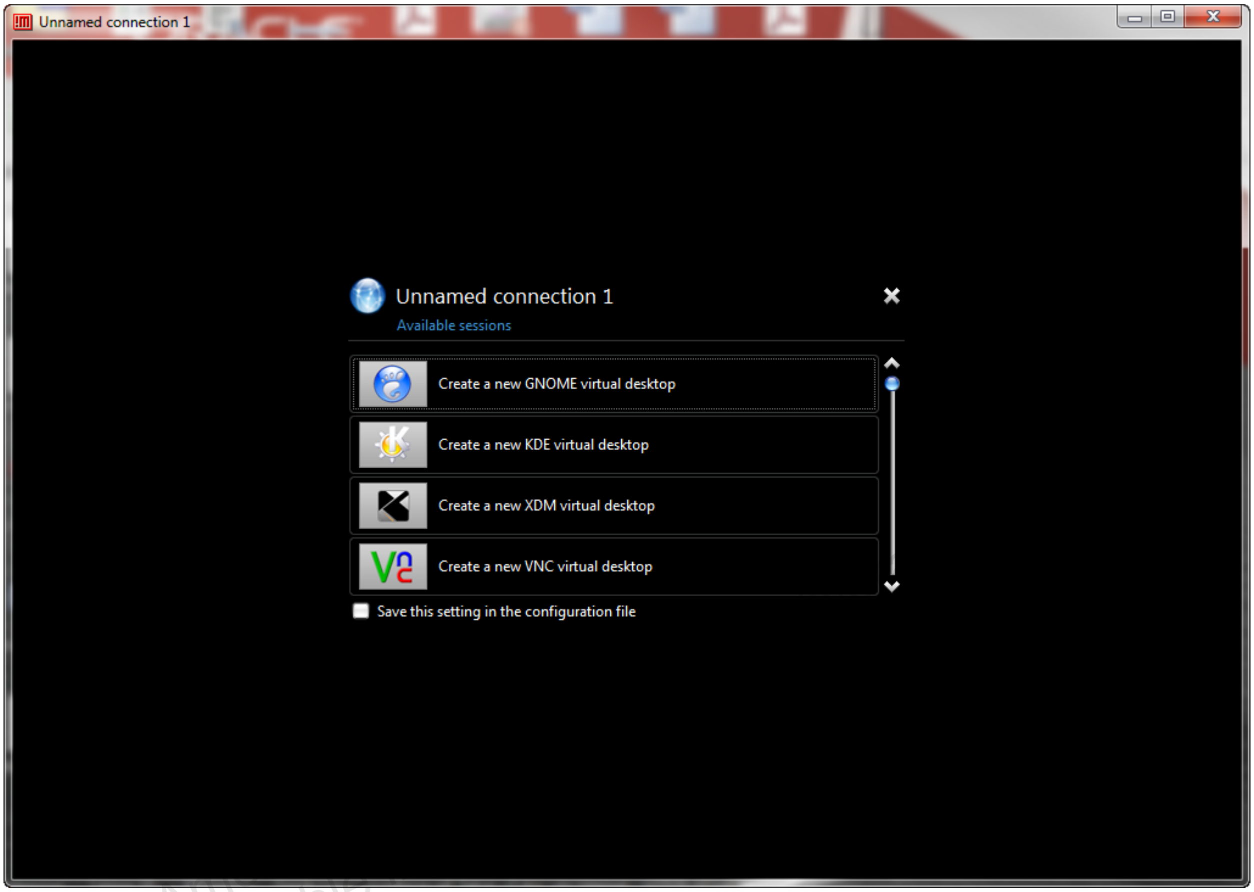
   a. Ensure that **Login as a named user** is selected.
   b. For **Username**, enter `vncuser`.
   c. For **Password**, enter `vnctech`.
   d. Click **Ok**.

4. Create a new session.
   - The window shown in the following screenshot appears.

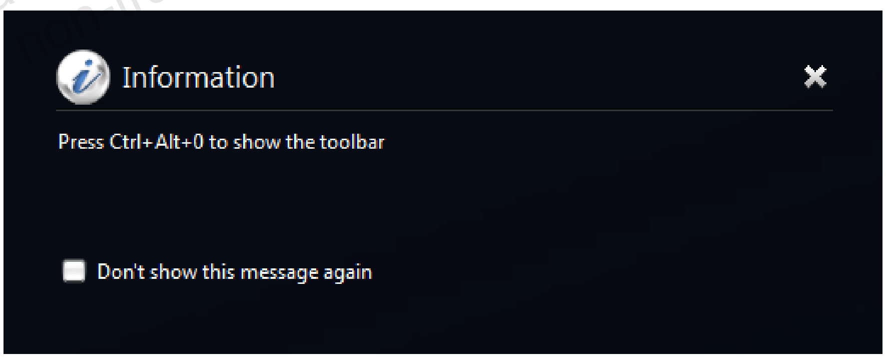


   a. Click **Create a new session**.

5. Create a new GNOME virtual desktop.
   - The window shown in the following screenshot appears.



   a. Click **Create a new GNOME virtual desktop**.
   b. Click the **X** in the **Information** message box to close the box.



   - The **dom0** GNOME virtual desktop window appears.

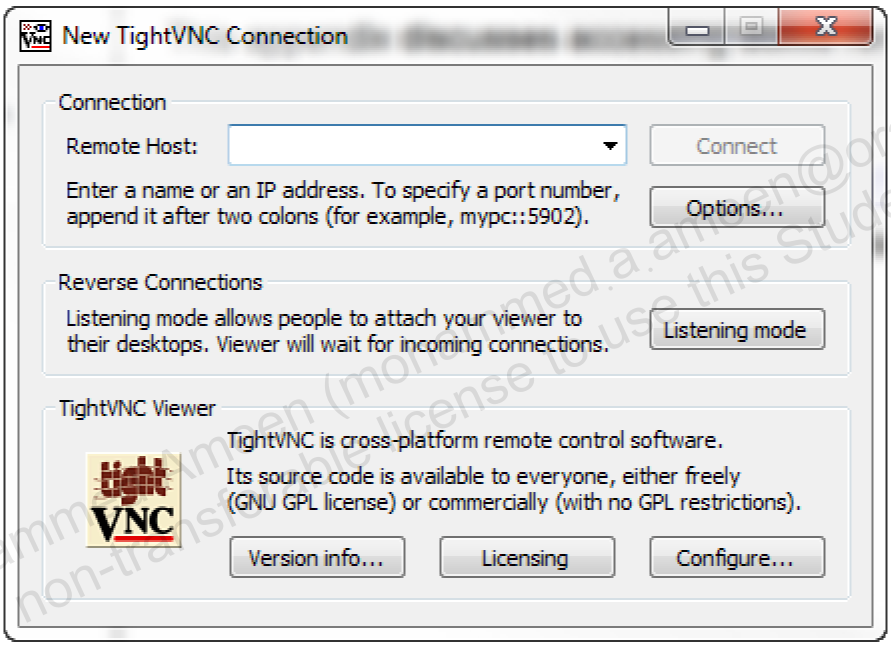
Appendix C: Using VNC (TightVNC) to Connect Directly to VM Guests

Overview

This appendix discusses accessing the VM guest systems that directly uses VNC (TightVNC). It is not recommended to connect to **dom0** or to the **host03** VM by using VNC. Both **dom0** and **host03** have the GNOME user interface, which causes various problems when connecting using VNC.

Steps

1. Install **tightvnc** (if necessary) from http://www.tightvnc.com/.
2. Run **TightVNC Viewer** (for example, select **TightVNC Viewer** from the Windows Start menu).
   - The following New TightVNC Connection window appears.



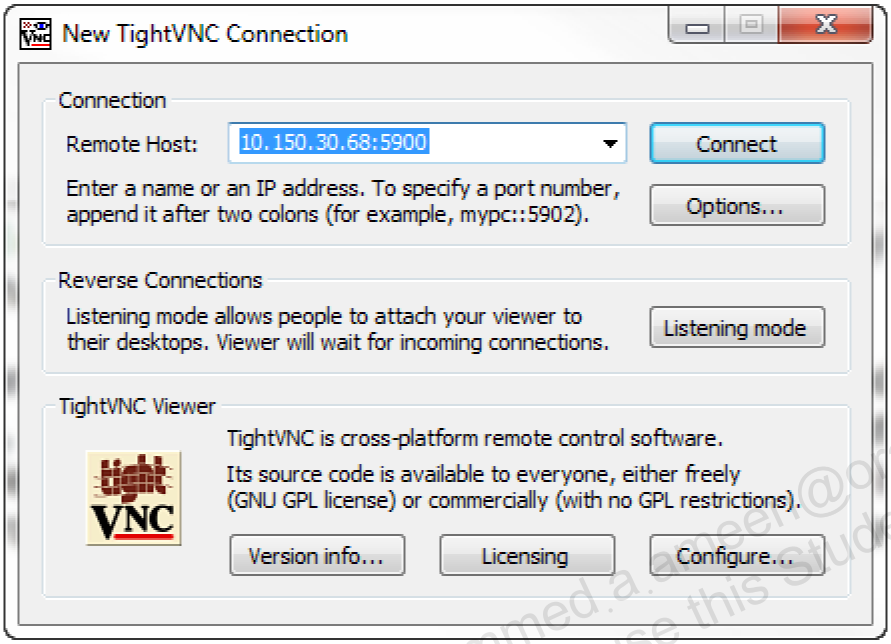
3. Connect directly to your **host01** virtual machine.
   - The following assumptions are made:
     - The **host01** VM was created first (has a port number of `5900`).
     - The **host02** VM was created next (has a port number of `5902`).
     - The **host03** VM was created last (has a port number of `5903`).
   - The output of the following commands (from **dom0** as `root`) indicates that this assumption is true.

```
# xm list -l host01 | grep location
          (location 0.0.0.0:5900)
# xm list -l host02 | grep location
          (location 0.0.0.0:5902)
# xm list -l host03 | grep location
          (location 0.0.0.0:5903)
```

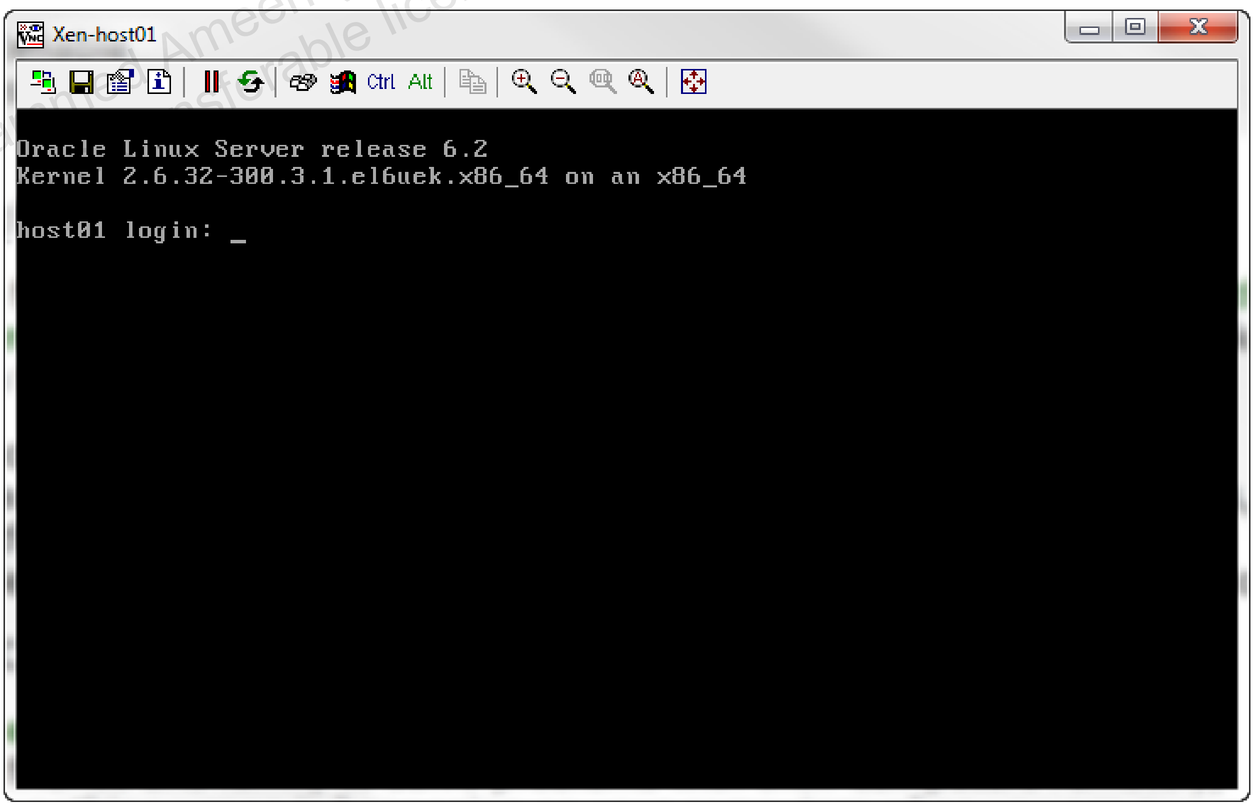
- Enter the IP address (provided by your instructor), followed by the port number to connect directly a VM guest.

a. To connect directly to the **host01** VM, enter the following.

  - In this example, the IP address of your student PC is 10.150.30.68. Your IP address is different.



1) Click **Connect**.
   - A terminal window appears.



2) Log in as `root` password with `0racle` (leading zero, not letter O).

3) Enter the `hostname` command to confirm that you are logged in to **host01**.

```
# hostname
host01.example.com
```

4) Log off by entering the `exit` command.

5) Close the VNC window by clicking the **X** in the top-right corner of the window.