ORACLE UNIVERSITY

Oracle Linux 7: System Administration

Activity Guide | Volume 1
D88168GC10
Edition 1.0 | January 2015 | D89910

Learn more from Oracle University at **oracle.com/education/**

ORACLE

Author

Craig McBride

Technical Contributors and Reviewers

Yasar Akthar, Gavin Bowe, Avi Miller, Chris Potter, Tim Hill, Manish Kapur, Wim Coekaerts, Al Flournoy, Joel Goodman, Harald Van Breederode, Michele Dady, Steve Miller, Antoinette O'Sullivan

This book was published using: Oracle Tutor

Table of Contents

Practices for Lesson 1: Course Introduction

Chapter 1

Course Practice Environment: Security Credentials

For OS usernames and passwords, see the following:

- If you are attending a classroom-based or live virtual class, ask your instructor or LVC producer for OS credential information.
- If you are using a self-study format, refer to the communication that you received from Oracle University for this course.

For product-specific credentials used in this course, see the following table:

Product-Specific Credentials		
Virtual Machines/Application	**Username**	**Password**
host01/OS	root	oracle
host01/OS	oracle	oracle
host02/OS	root	oracle
host02/OS	oracle	oracle
host03/OS	root	oracle
host03/OS	oracle	oracle

Practices for Lesson 1: Overview

Practices Overview

In these practices, you will:

- Log in to your classroom PC and become familiar with the Oracle VM Server for x86 environment installed on your classroom PC
- Connect to the virtual machines used for the hands-on practices and become familiar with the VM guest configurations

Assumptions

- Your instructor has assigned a student PC to you.
- The student PC is running OVM 3.2.1.
- 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 7.
- You install Oracle Linux 7 on **host03** in Practice 3.

Practice 1-1: Exploring the dom0 Environment

Overview

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

Assumptions

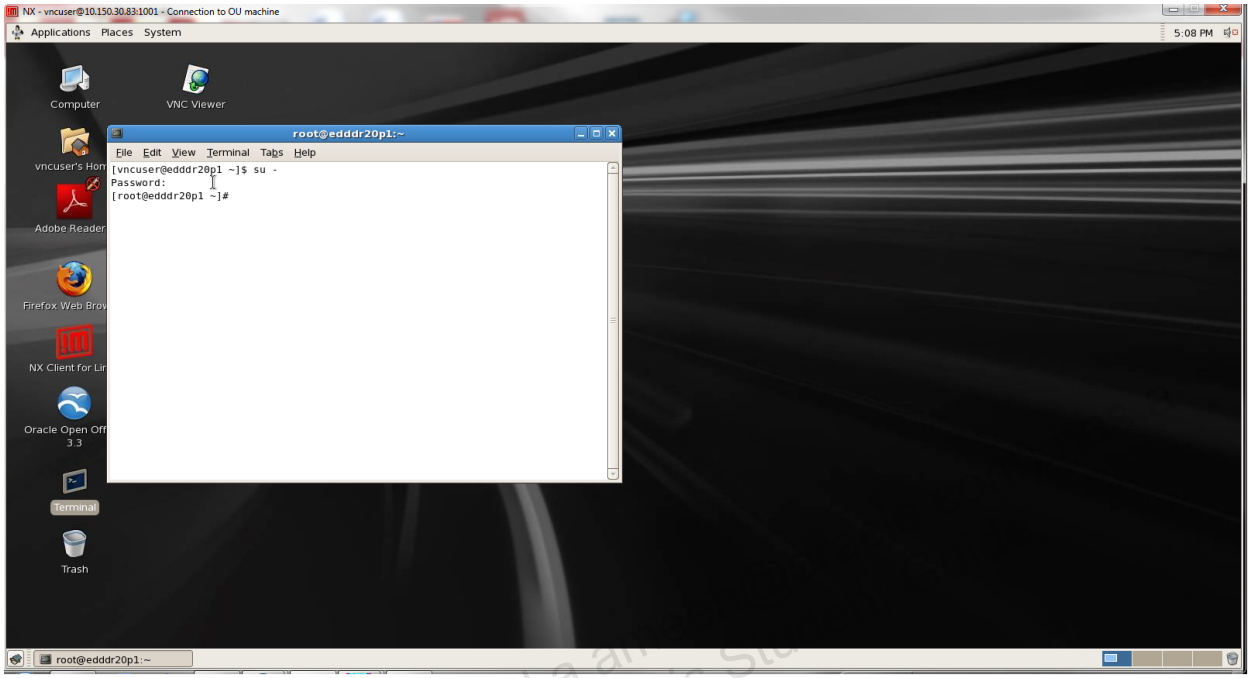
- Your instructor has assigned a student PC to you.
- Your student PC is running Oracle VM Server for x86, version 3.2.1.
- You are logged in to your student PC as `vncuser` with the password `vnctech`.
- The GNOME desktop is installed on **dom0**.
- There are three guests (virtual machines): **host01**, **host02**, and **host03**.

Tasks

1. Open a terminal window.
   - Begin this task from the **dom0** GNOME virtual desktop window as shown in the following screenshot:

   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 in the following screenshot:



Become the `root` user by using the `su -` command. The `root` password is `oracle`. Confirm that you are `root` by printing the user identity with the `whoami` command:

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

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

a. Use the `cat` command to view the contents of the `/etc/ovs-release` file.

```
# cat /etc/ovs-release
Oracle VM server release 3.2.1
```

b. Use the `uname -a` command to display the operating system version.

```
# uname -a
Linux edddr20p1 2.6.39-300.22.2.el5uek #1 SMP Fri Jan 4 12:40:29
PST 2013 x86_64 x86_64 x86_64 GNU/Linux
```

- In this example, the Linux kernel is `2.6.39-300.22.2.el5uek`.
- The host name is `edddr20p1` (your host name is different).

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

   Use the `ifconfig -a` command to display the network configuration. Only partial output is shown.

```
# ifconfig -a
...
bond0     Link encap:Ethernet ...
          inet addr:10.150.30.83 ...
...
eth0      Link encap:Ethernet ...
...
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
...
```

   - In this example, the network interface for **dom0** is assigned an IP address of `10.150.30.83`. The IP address of your system is different.
   - The `lo` interface is a software loopback interface that identifies the `localhost`. It is always assigned an IP address of `127.0.0.1`.
   - The `virbr0` interface is a `xen` bridge interface used by VM guests. It is assigned an IP address of `192.0.2.1`.
   - The `virbr1` interface is a second `xen` bridge interface used by VM guests. It is assigned an IP address of `192.168.1.1`.
   - You also note the `vif<#>.<#>` entries. These are 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**.

   a. Explore the top level of the `/OVS` directory. (Only partial output is shown.)

```
# ls -l /ovs
drwxr-xr-x ... iso_pool
drwxr-xr-x ... publish_pool
drwxr-xr-x ... running_pool
drwxr-xr-x ... seed_pool
drwxr-xr-x ... sharedDisk
```

   - There are five 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 VM directories exist, for VMs **host01**, **host02**, and **host03**.

c. Explore the **host01** VM directory.

```
# cd /OVS/running_pool/host01
# ls -l
-rw-r--r-- ... system.img
-rw-r--r-- ... u01.img
-rw-r--r-- ... u02.img
-rwxr-xr-x ... vm.cfg
```

- The `system.img` file is the operating system virtual disk.
- The `u01.img` and `u02.img` files are utility virtual disks that are used in various practices in this course.
- The `vm.cfg` file is the configuration file for the virtual machine. This file is read when the virtual machine is created.

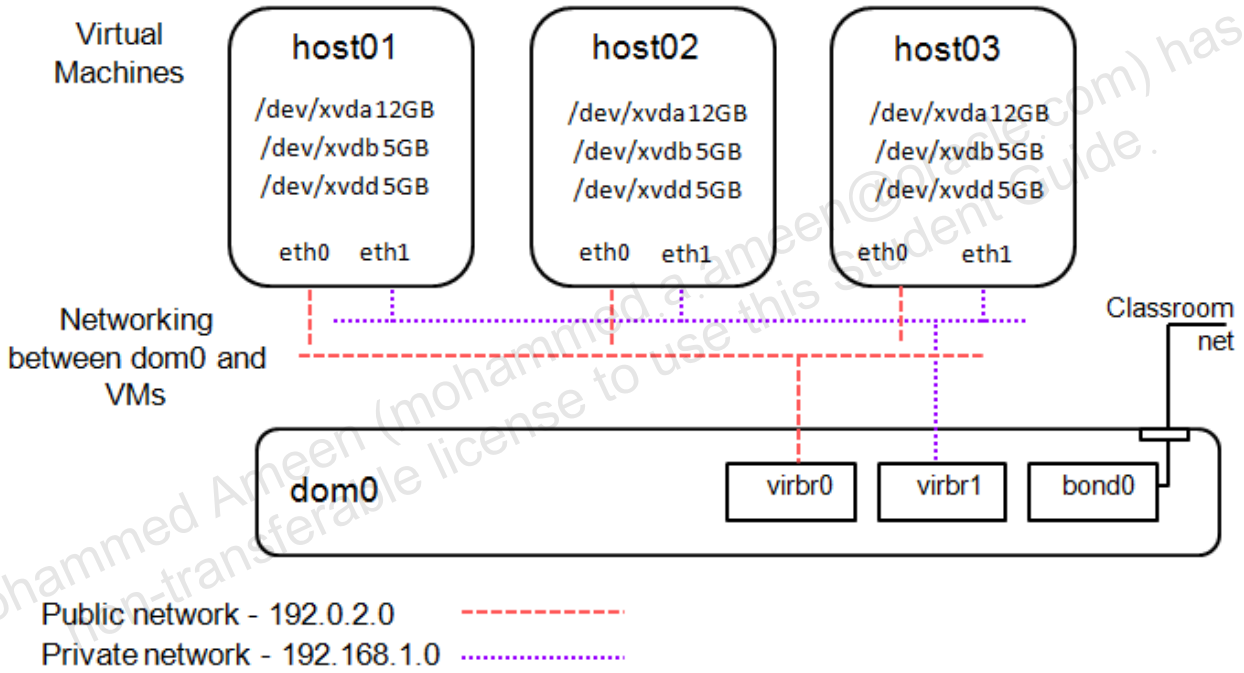
d. View the `vm.cfg` file.

```
# cat vm.cfg
name = 'host01'
builder = 'hvm'
memory = 2048
boot = 'cd'
disk = [ 'file:/OVS/running_pool/host01/system.img,hda,w',
         'file:/OVS/running_pool/host01/u01.img,hdb,w',
         'file:/OVS/running_pool/host01/u02.img,hdd,w',
         'file:/OVS/seed_pool/OracleLinux-R7-GA-Server-x86_64-
dvd.iso,hdc:cdrom,r']
vif = [ 'mac=00:16:3e:00:01:01, bridge=virbr0',
        'mac=00:16:3e:00:02:01, 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'
```

- Note that there are three virtual disks represented by the three `.img` files.
- Note that the Oracle Linux 7 `dvd.iso` is mounted on a virtual CD ROM device.
- Note that 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.



e. Explore the `/OVS/seed_pool` directory:

```
# cd /OVS/seed_pool
# ls -l
drwxr-xr-x  ...  btrfs
drwxr-xr-x  ...  ksplice
-rw-r--r--  ...  OracleLinux-R7-GA-Server-x86_64-dvd.iso
drwxr-xr-x  ...  oswbb
drwxr-xr-x  ...  sfws
-rwxr-xr-x  ...  system01.tgz
-rwxr-xr-x  ...  system02.tgz
-rwxr-xr-x  ...  system03.tgz
drwxr-xr-x  ...  ts_scripts
drwxr-xr-x  ...  ts_source
-rwxr-xr-x  ...  u01_01.tgz
-rwxr-xr-x  ...  u01_02.tgz
-rwxr-xr-x  ...  u01_03.tgz
-rwxr-xr-x  ...  u02_01.tgz
-rwxr-xr-x  ...  u02_02.tgz
-rwxr-xr-x  ...  u02_03.tgz
-rwxr-xr-x  ...  vm01.cfg
-rwxr-xr-x  ...  vm02.cfg
-rwxr-xr-x  ...  vm03.cfg
```

- This directory contains many files used to create the initial environment.
- Oracle Linux 7 is installed on the **host01** and **host02** VMs from the `OracleLinux-R7-GA-Server-x86_64-dvd.iso` file in this directory.
- Other files in this directory are used in various practices throughout this course.

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 here is a sample, the `ID` and `Time(s)` values will be different on your system.

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

   - You have three guests (**host01**, **host02**, and **host03**) running.

2. Shut down a VM.

   Use the `xm shutdown -w <VM name>` command to shut down the **host01** VM. The `-w` option tells the system to wait until all services in the domain are shut down cleanly. Run `xm list` to display the running VMs.

   - The `xm shutdown` command takes a few seconds to complete.

```
# xm shutdown -w host01
Domain host01 terminated
All domains terminated
# xm list
Name                  ID     Mem  VCPUs       State      Time(s)
Domain-0               0    1024      2      r-----       289.6
host02                 2    2048      1      -b----       159.0
host03                 3    2048      1      -b----        13.2
```

   - Note that **host01** is no longer active.

3. Start a VM.

   Use the `xm create <config_file>` command to start the **host01** VM. `<config_file>` is named `vm.cfg` and is located in the `/OVS/running_pool/<VM_name>` directory. Run `xm list` to display the running VMs.

```
# cd /OVS/running_pool/host01
# xm create vm.cfg
Using config file "./vm.cfg".
Started domain host01 (id=#)
```

```
# xm list
Name                        ID     Mem  VCPUs       State      Time(s)
Domain-0                     0    1024      2      r-----       304.5
host01                       4    2048      1      r-----        18.7
host02                       2    2048      1      -b----       159.0
host03                       3    2048      1      -b----        13.2
```

- Note that **host01** is now active.
- The State column for **dom0** and **host01** shows 'r' (run state). The State column for **host02** and **host03** shows 'b' (blocked). The following describes these values:
  - r: The domain is currently running and healthy.
  - b: The domain is blocked, and not running or runnable. This can be caused because the domain is waiting on IO (a traditional wait state) or has gone to sleep because there was nothing else for it to do.

Practice 1-3: Exploring the host01 VM

Overview

In this practice, you perform the following:

- Log in to **host01**.
- View the Oracle Linux version on **host01**.
- View the Unbreakable Enterprise Kernel version on **host01**.
- View the storage devices available on **host01**.
- View the network configuration on **host01**.

Assumptions

- You are logged on to **dom0** as the `root` user.
- The **host01** VM guest is running.

Tasks

1. Connect to **host01** guest by using **vncviewer**.
   a. From **dom0**, determine the vnc port number for **host01** by running the `xm list -l host01 | grep location` command.

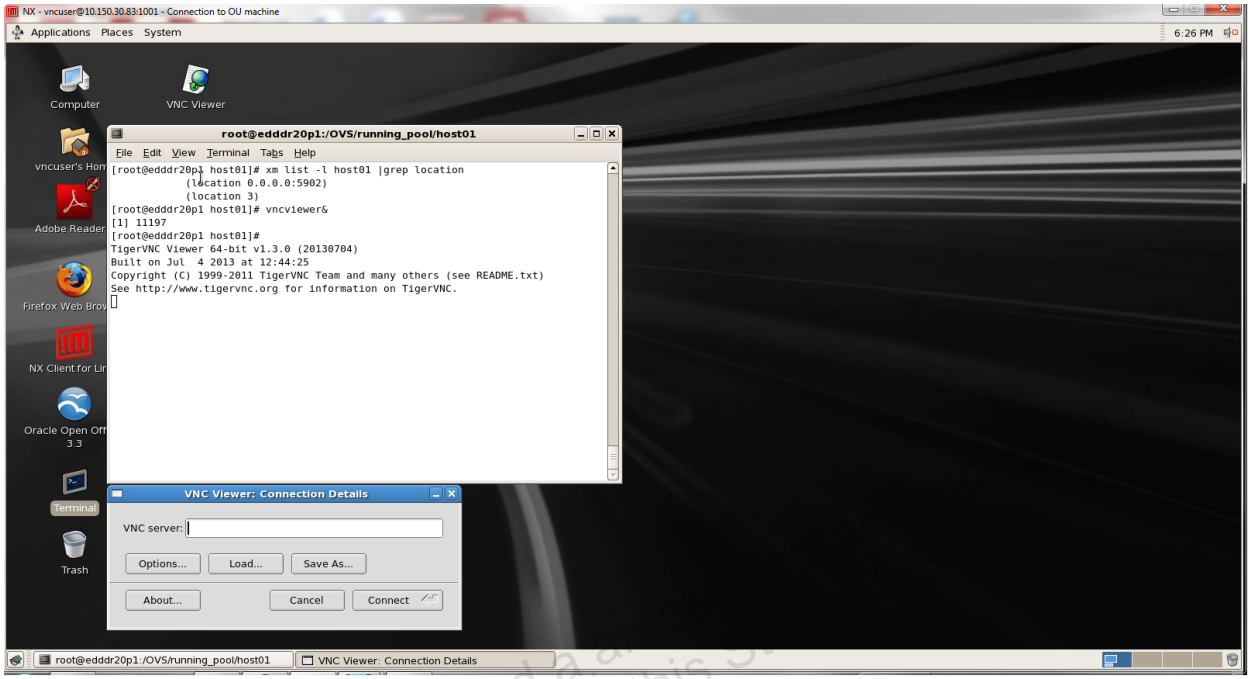
```
[dom0]# xm list -l host01 | grep location
                (location 0.0.0.0:5902)
                (location 3)
```

   - The sample shown indicates that the port number is **5902**.

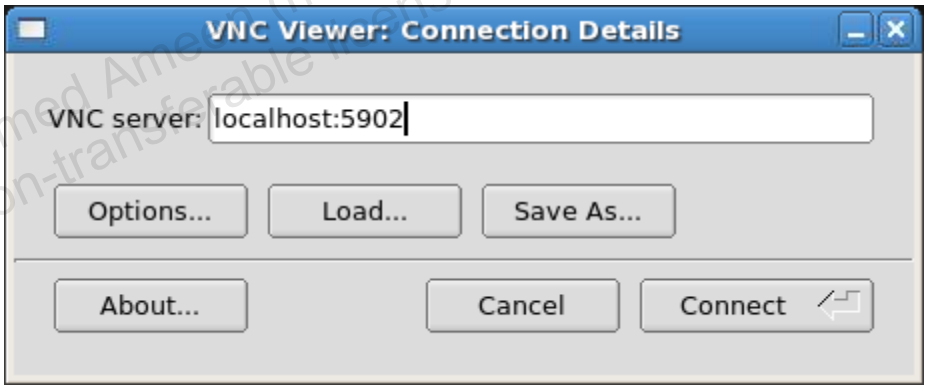
b. From **dom0**, run the `vncviewer&` command:

```
[dom0]# 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 `5902`, enter `localhost:5902` and click **Connect**.

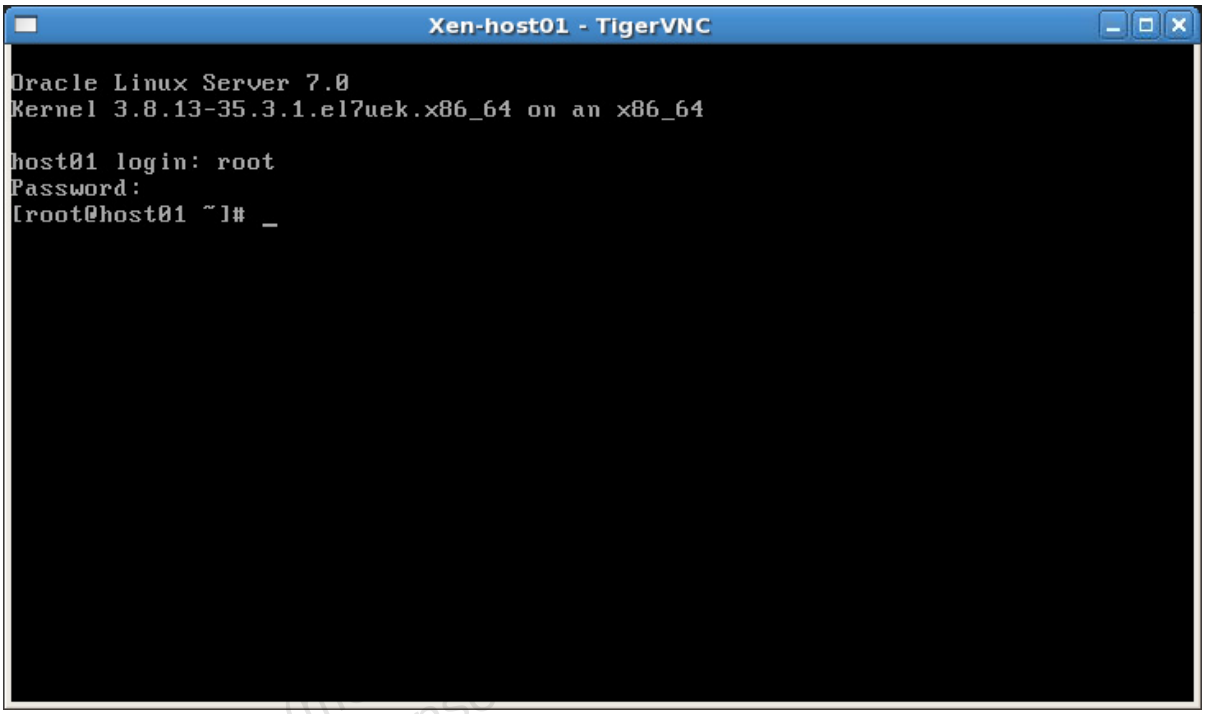


- You could also enter `localhost:2` (omit the "`590`" from the port number).

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.

```
host01 login: root
Password: oracle
```



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`.

Alternatively, from **dom0**, use the `ssh` command to connect to the **host01** VM guest.

- Because this is the first time you have logged in by using `ssh`, the command checks to make sure that you are connecting to the host that you think you are connecting to. Enter **`yes`**.
- The `root` password is `oracle` (all lowercase).
- If you get a message, "**ssh: connect to host host01 port 22: No route to host**," wait a few seconds to allow **host01** to boot and then run the `ssh host01` command again.

```
[dom0]# 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
[root@host01 ~]# hostname
host01.example.com
```

- The `hostname` command confirms you have successfully logged in to **host01**.

3. Explore the **host01** VM guest.

   a. Use the `cat` command to view the contents of the `/etc/oracle-release` file.

```
# cat /etc/oracle-release
Oracle Linux Server release 7.0
```

   b. Use the `uname -r` command to determine your running kernel version.

```
# uname -r
3.8.13-35.3.1.el7uek.x86_64
```

   - You are running the Unbreakable Enterprise Kernel (UEK) Release 3.1. The UEK became the default kernel beginning with Oracle Linux 5.5.

   c. Use the `fdisk` command to view the storage devices.

```
# fdisk -l | grep /dev
Disk /dev/xvda: 12.9 GB, 12884901888 bytes, 25165824 sectors
/dev/xvda1   *      ...                83   Linux
/dev/xvda2          ...                83   Linux
/dev/xvda3          ...                83   Linux
/dev/xvda4          ...                 5   Extended
/dev/xvda5          ...                82   Linux swap / Solaris
Disk /dev/xvdb: 5368 MB, 5368709120 bytes, 10485760 sectors
Disk /dev/xvdd: 5368 MB, 5368709120 bytes, 10485760 sectors
```

   - Three devices are available: `/dev/xvda`, `/dev/xvdb`, and `/dev/xvdd`.
   - Do not run the following commands, this is information only:
     - The `/dev/xvda` disk device represents a 12 GB system image file created with the following command (in the `/OVS/running_pool/host01` directory on **dom0**):

```
# dd if=/dev/zero of=system.img bs=1M count=12288
```

     - The `/dev/xvdb` disk device represents a 5 GB utility image file created with the following command (in the `/OVS/running_pool/host01` directory on **dom0**):

```
# dd if=/dev/zero of=u01.img bs=1M count=5120
```

     - The `/dev/xvdd` disk device represents a 5 GB utility image file created with the following command (in the `/OVS/running_pool/host01` directory on **dom0**):

```
# dd if=/dev/zero of=u02.img bs=1M count=5120
```

   - The `/dev/xvda` device has Oracle Linux 7 Minimal Install packages installed.

   d. Use the `ip addr` command to display the network interfaces. Only partial output is shown.

```
# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP>   mtu 65536 qdisc noqueue ...
```

```
    link/loopback 00:00:00:00:00:00: brd 00:00:00:00:00:00
    inet addr:127.0.0.1/8 scope host lo
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:01:01 brd ff:ff:ff:ff:ff:ff
    inet 192.0.2.101/24 brd 192.0.2.255 scope global eth0
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:01 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.101/24 brd 192.168.1.255 scope global eth1
...
```

- The system has two Ethernet network interfaces, `eth0` and `eth1`.
- The `eth0` interface is on the `192.0.2` subnet, has an IPv4 address of `192.0.2.101`, and provides access to **dom0** and the other VM guest systems.
- The `eth1` interface is on the `192.168.1` subnet and has an IPv4 address of `192.168.1.101`.

e. View the `/etc/hosts` file on **host01**.

```
# 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
```

f. Use the `exit` command to log off **host01**.

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

Practice 1-4: Exploring the host02 VM

Overview

In this practice, you perform the following:

- Log in to **host02**.
- View the Oracle Linux version on **host02**.
- View the Unbreakable Enterprise Kernel version on **host02**.
- View the storage devices available on **host02**.
- View the network configuration on **host02**.

Assumptions

- You are logged on to **dom0** as the `root` user.
- The **host02** VM guest is running.

Tasks

1. Explore the **host02** VM guest.
   a. Use the `ssh` command to log in to **host02**.
      - Because this is the first time you have logged in using `ssh`, the command checks to make sure that you are connecting to the host that you think you are connecting to. Enter **`yes`**.
      - The `root` password is `oracle` (all lowercase).

```
# ssh host02
The authenticity of host 'host02 (192.0.2.102)' can't be
established. RSA key fingerprint is ...
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'host02,192.0.2.102' (RSA) to the
list of known hosts.
root@host02's password: oracle
[root@host02 ~]# hostname
host02.example.com
```

      - The `hostname` command confirms whether you have successfully logged in to **host02**.

   b. Use the `cat` command to view the contents of the `/etc/oracle-release` file.

```
# cat /etc/oracle-release
Oracle Linux Server release 7.0
```

   c. Use the `uname -r` command to determine your running kernel version.

```
# uname -r
3.8.13-35.3.1.el7uek.x86_64
```

   d. Use the `fdisk` command to view the storage devices.

```
# fdisk -l | grep /dev
Disk /dev/xvda: 12.9 GB, 12884901888 bytes, 25165824 sectors
/dev/xvda1   *      ...                83   Linux
```

```
/dev/xvda2           ...                 83   Linux
/dev/xvda3           ...                 83   Linux
/dev/xvda4           ...                  5   Extended
/dev/xvda5           ...                 82   Linux swap / Solaris
Disk /dev/xvdb: 5368 MB, 5368709120 bytes, 10485760 sectors
Disk /dev/xvdd: 5368 MB, 5368709120 bytes, 10485760 sectors
```

- Three devices are available: `/dev/xvda`, `/dev/xvdb`, and `/dev/xvdd`.
- Do not run the following commands, this is information only:
  - The `/dev/xvda` disk device represents a 12 GB system image file created with the following command (in the `/OVS/running_pool/host02` directory on **dom0**):
    ```
    # dd if=/dev/zero of=system.img bs=1M count=12288
    ```
  - The `/dev/xvdb` disk device represents a 5 GB utility image file created with the following command (in the `/OVS/running_pool/host02` directory on **dom0**):
    ```
    # dd if=/dev/zero of=u01.img bs=1M count=5120
    ```
  - The `/dev/xvdd` disk device represents a 5 GB utility image file created with the following command (in the `/OVS/running_pool/host02` directory on **dom0**):
    ```
    # dd if=/dev/zero of=u02.img bs=1M count=5120
    ```
- The `/dev/xvda` device has Oracle Linux 7 Server with GUI packages installed plus the following Add-Ons:
  - FTP Server
  - File and Storage Server
  - Network File System Client

e. Use the `ip addr` command to display the network interfaces. Only partial output is shown.

```
# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP>   mtu 65536 qdisc noqueue ...
    link/loopback 00:00:00:00:00:00:00: brd 00:00:00:00:00:00
    inet addr:127.0.0.1/8 scope host lo
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:01:02 brd ff:ff:ff:ff:ff:ff
    inet 192.0.2.102/24 brd 192.0.2.255 scope global eth0
...
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc ...
    link/ether 00:16:3e:00:02:02 brd ff:ff:ff:ff:ff:ff
...
```

- The system has two Ethernet network interfaces, `eth0` and `eth1`.
- The `eth0` interface is on the `192.0.2` subnet, has an IPv4 address of `192.0.2.102`, and provides access to **dom0** and the other VM guest systems.
- The `eth1` interface does not have an IP address assigned.

f. View the `/etc/hosts` file on **host02**.

- This file contains default installation settings.

```
# cat /etc/hosts
127.0.0.1       localhost localhost.localdomain  localhost4 ...
::1             localhost localhost.localdomain  localhost6 ...
```

g. Use the `exit` command to log off **host02**.

```
# exit
logout
Connection to host02 closed.
```

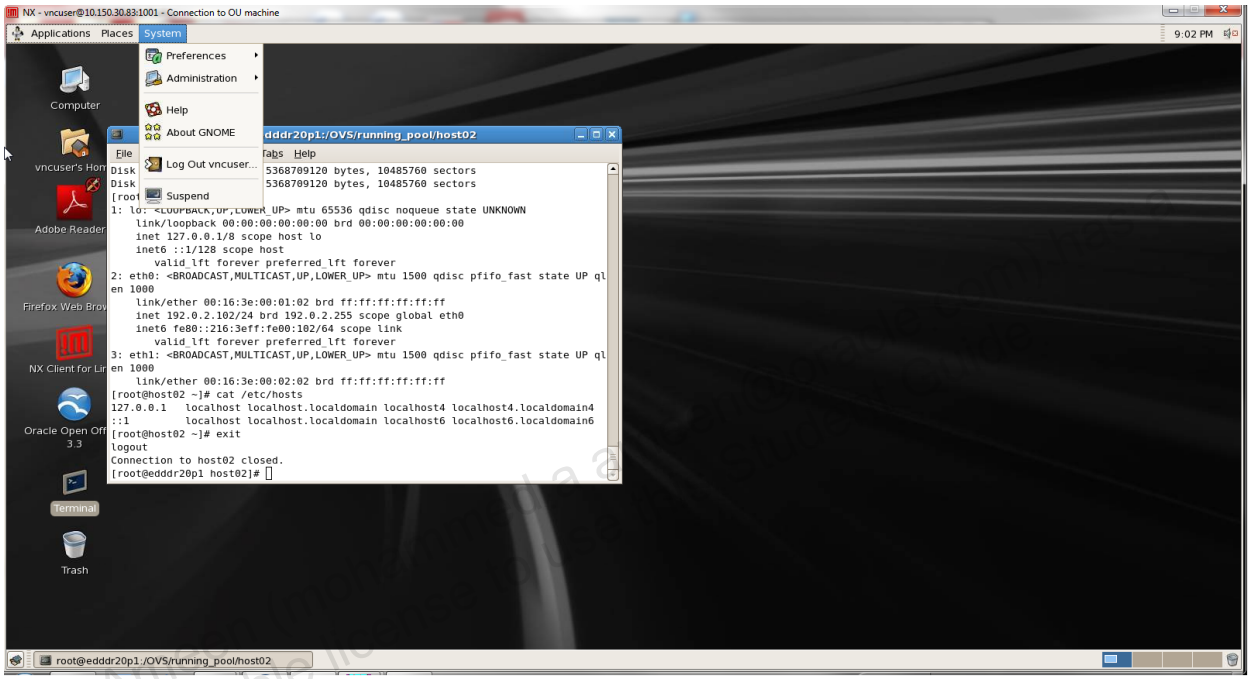
Practice 1-5: Logging Off from Your Student PC

Overview

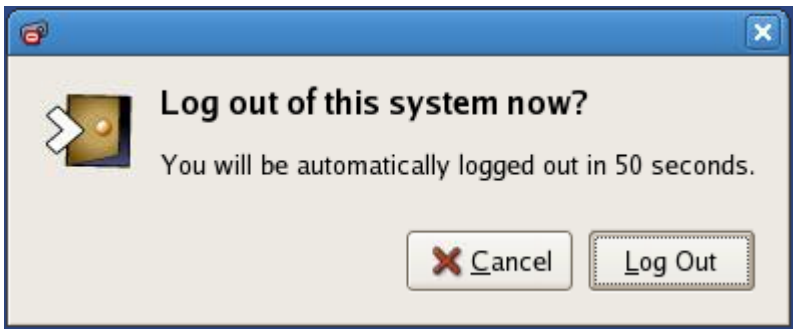
In this practice, you learn how to log off from your system.

Tasks

1. Learn how to log off your student PC.
   a. Open the **System** menu on the GNOME desktop.



   b. Select **Log Out vncuser** from the System menu.
      - You would click the **Log Out** button to log out.
      - Do not log out, however, until the end of each day of training.
   c. Click the **Cancel** button to stay logged in.



      - 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: Overview

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

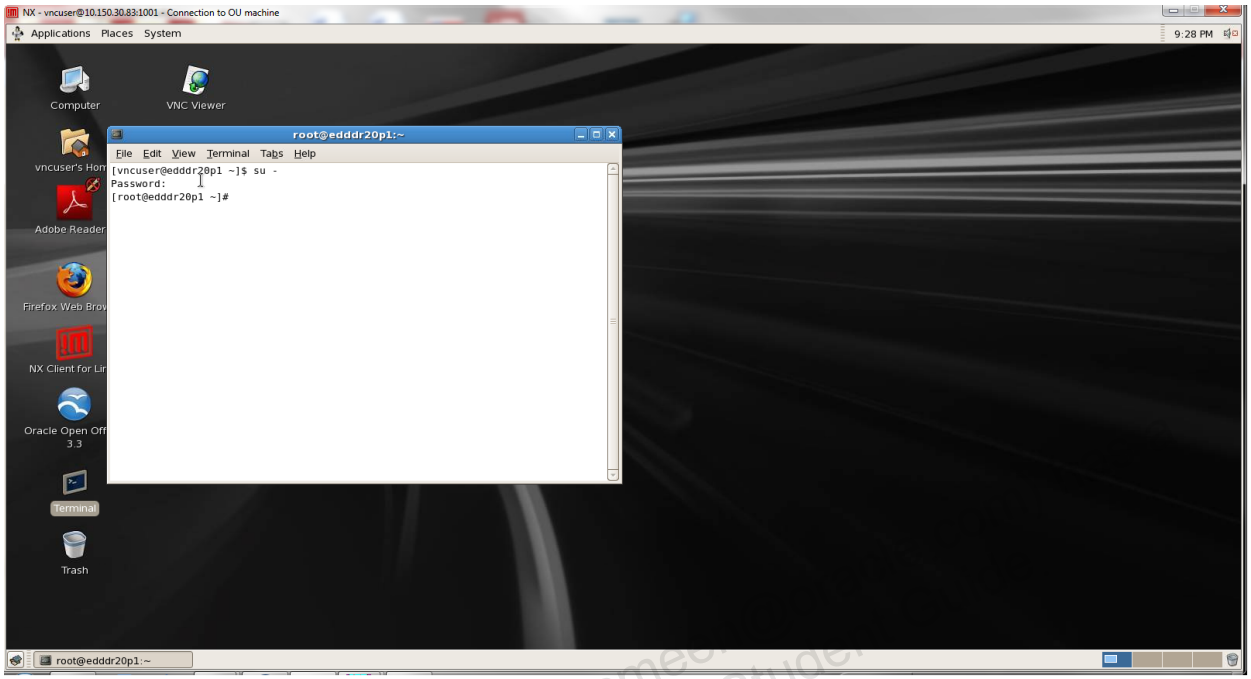
1. Open a terminal window.

- 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 the **host01** guest as the `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
3.8.13-35.3.1.el7uek.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-3.10.0-123.el7.x86_64
kernel-uek-3.8.13-35.3.1.el7uek.x86_64
```

```
...
```

- Note that the following two `kernel` packages are installed (associated `firmware` packages, and `headers` and `tools` packages are also installed):
  - Unbreakable Enterprise Kernel package (`kernel-uek-3.8.13-35.3.1.el7uek.x86_64`)
  - Red Hat–compatible kernel package (`kernel-3.10.0-123.el7.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--. os-release
-rw-r--r--. redhat-release
lrwxrwxrwx. system-release - > oracle-release
# cat oracle-release
Oracle Linux Server release 7.0
# cat redhat-release
Red Hat Enterprise Linux Server release 7.0 (Maipo)
```

- 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 7

Chapter 3

Practices for Lesson 3: Overview

Practices Overview

In these practices, you:

- Install Oracle Linux 7 on the **host03** virtual machine
- Log in to **host03** and perform shutdown operations

Practice 3-1: Installing Oracle Linux 7

Overview

In this practice, you install Oracle Linux 7 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.
   a. From **dom0**, use the `xm destroy` command as follows:
      - Destroy the virtual machine and create it again in step 1c.

```
[dom0]# xm destroy host03
```

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

```
[dom0]# cd /OVS/running_pool/host03
```

   c. Use the `xm create` command to create the **host03** VM as follows:

```
[dom0]# xm create vm.cfg
Using config file "./vm.cfg".
Started domain host03 (id=...)
```

2. Connect to the **host03** guest by using **vncviewer**.
   a. Determine the vnc port number for **host03** by running the `xm list -l host03 | grep location` command.

```
[dom0]# xm list -l host03 | grep location
                (location 0.0.0.0:5904)
                (location 3)
```

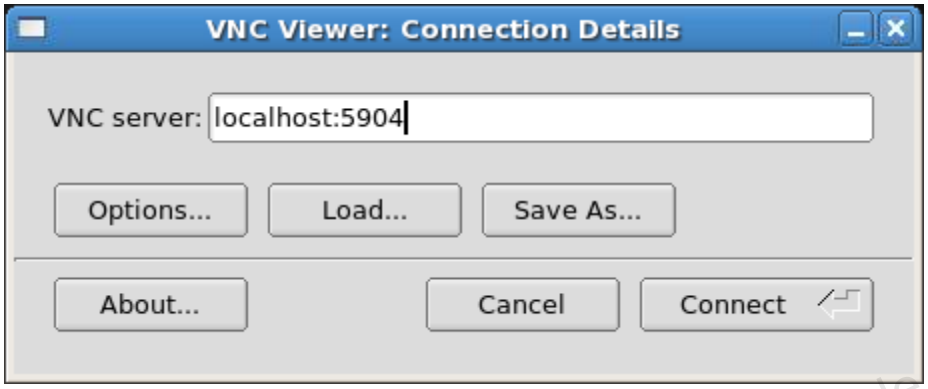
      - The sample shown indicates that the port number is `5904`.

b. Run the `vncviewer&` command.

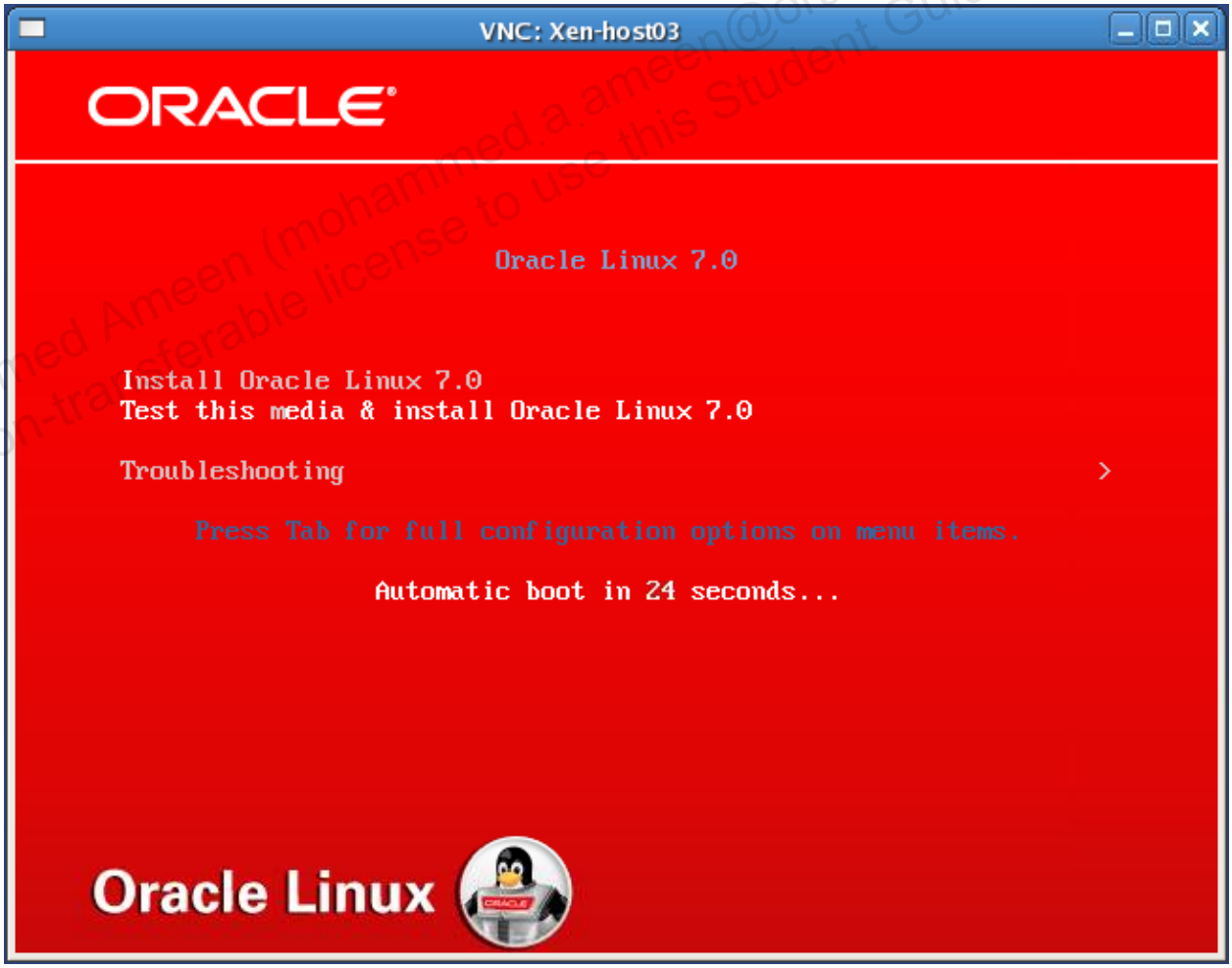
```
[dom0]# vncviewer&
```

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

c. Enter `localhost:<port_number>`, substituting the port number displayed from the previous `xm list -l host03 | grep location` command. For example, if the port number is `5904`, enter `localhost:5904` and click **Connect**.

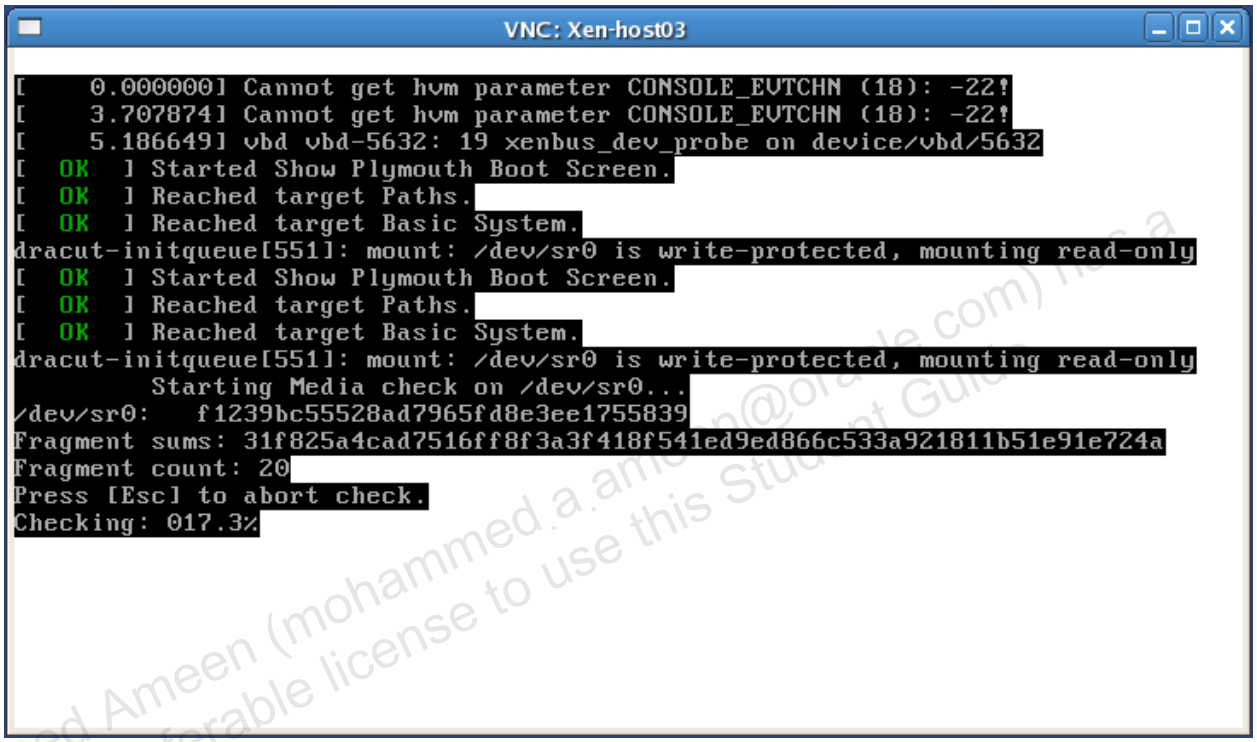


- The Oracle Linux 7 installation menu appears as shown:



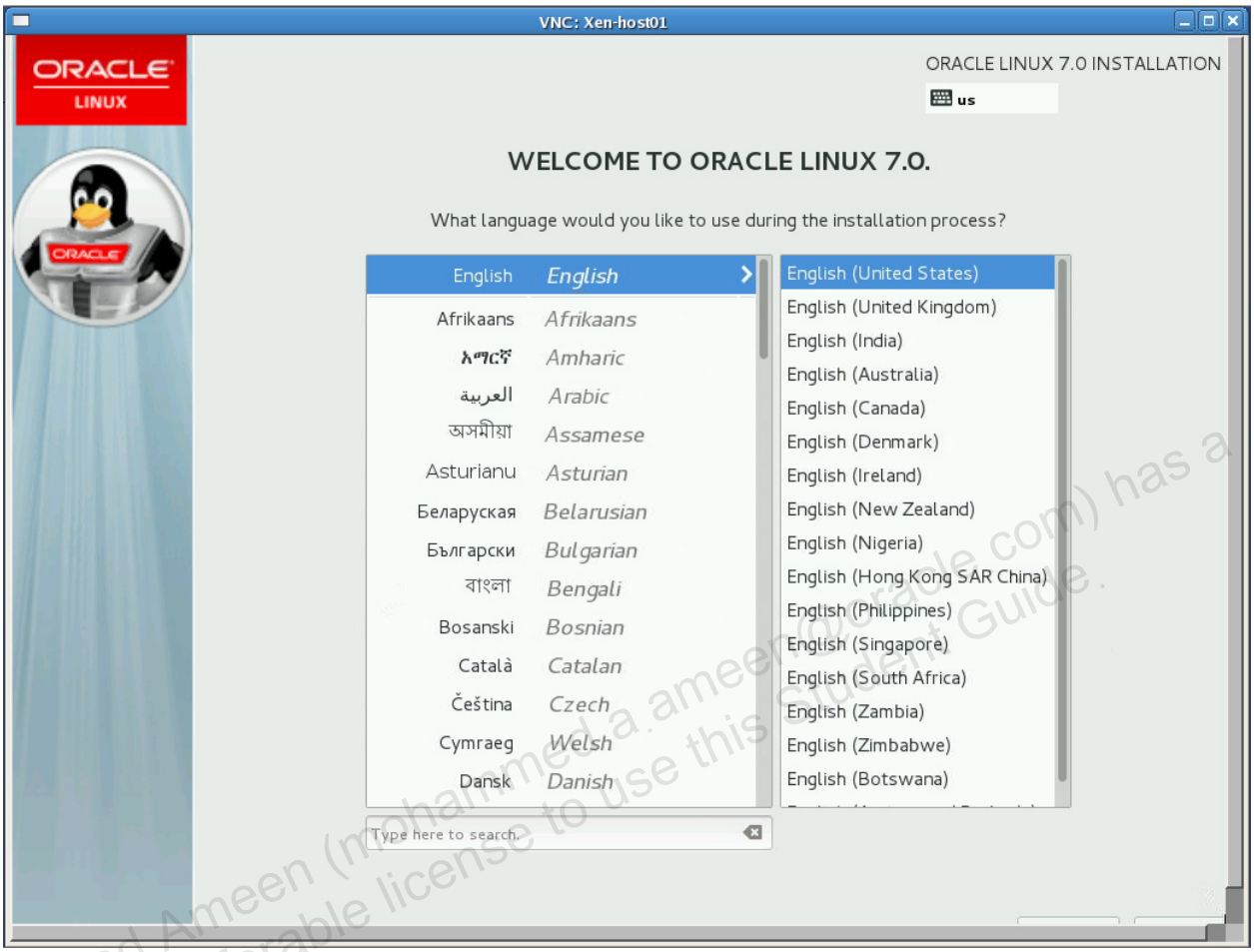
- Notice that the default option is **Test this media & install Oracle Linux 7.0**. This option is automatically selected after 60 seconds. Therefore, the menu may or may not appear, depending on how quickly you connected to **host03** using **vncviewer**.

3. Use the up arrow to highlight **Install Oracle Linux 7.0** from the menu. Press Enter to select this option.
   - If you do not see the Oracle Linux 7.0 menu, it means the 60 second timeout has expired and the media test has begun. In this case, a screen similar to the following appears:



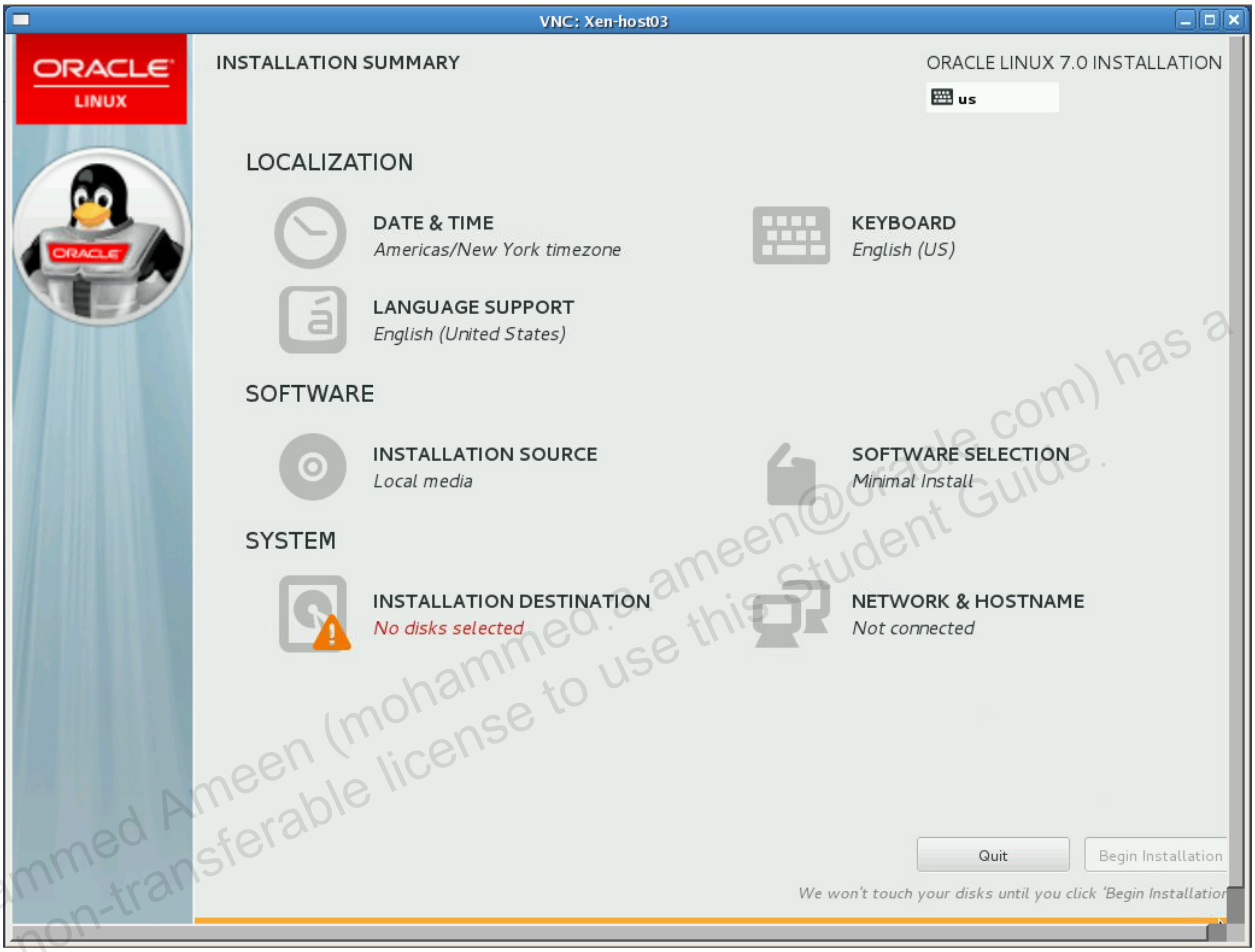
   - You can press the Esc key to abort the media test, or do nothing to let the test run to completion.

4. The first window to appear after choosing to install Oracle Linux 7 is the LANGUAGE SELECTION window.



a. Select the appropriate language.
   If taking this class in the United States, select **English (United States)**.
b. Scroll down if necessary and click **Continue**.

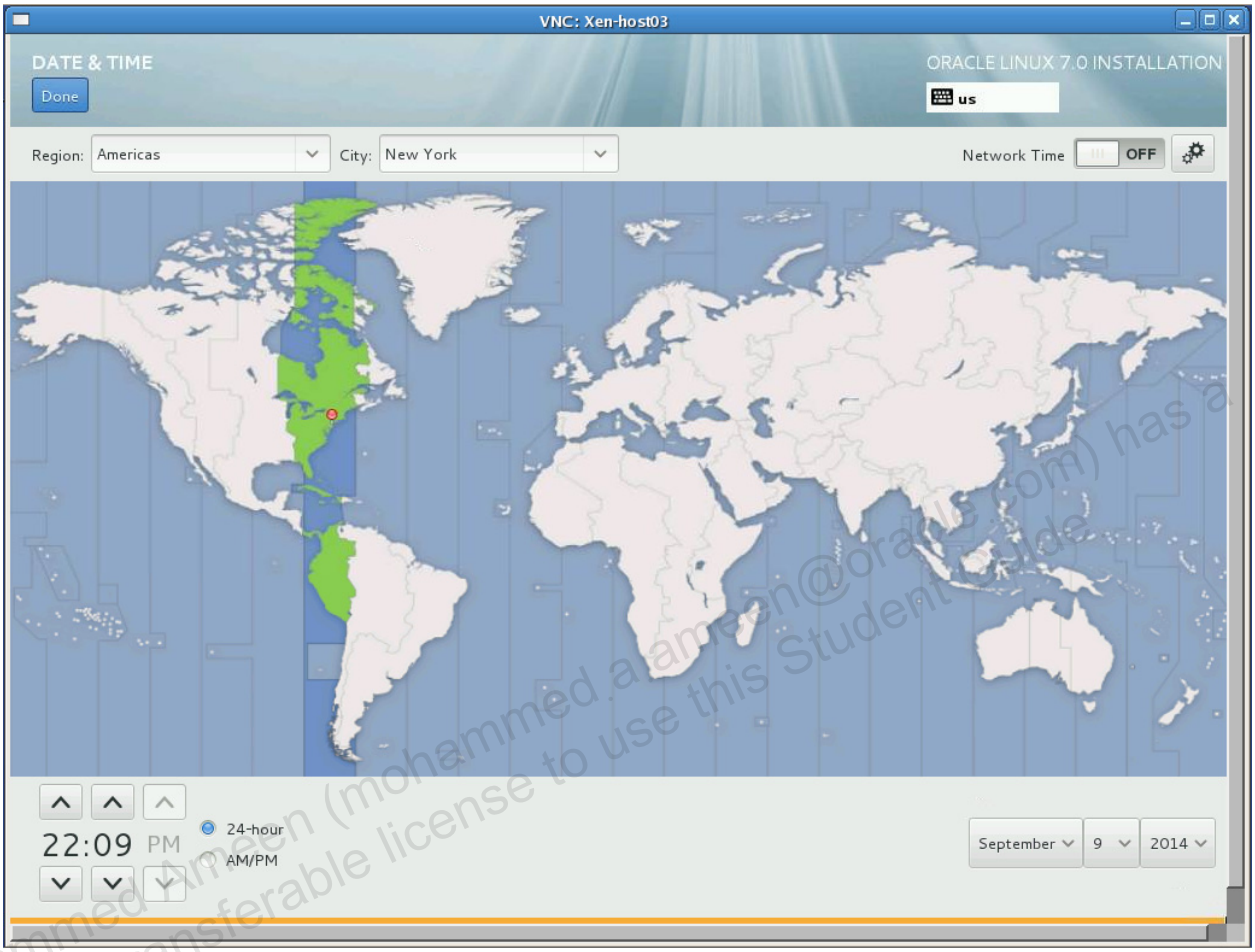
5. The INSTALLATION SUMMARY window appears next.
   - You can select the installation options under LOCALIZATION, SOFTWARE, or SYSTEM in any order.
   - This INSTALLATION SUMMARY window reappears after you complete each installation option.



For this exercise, begin by selecting the LOCALIZATION menu options.

6. Configure DATE & TIME.
   a. Select the DATE & TIME menu option.

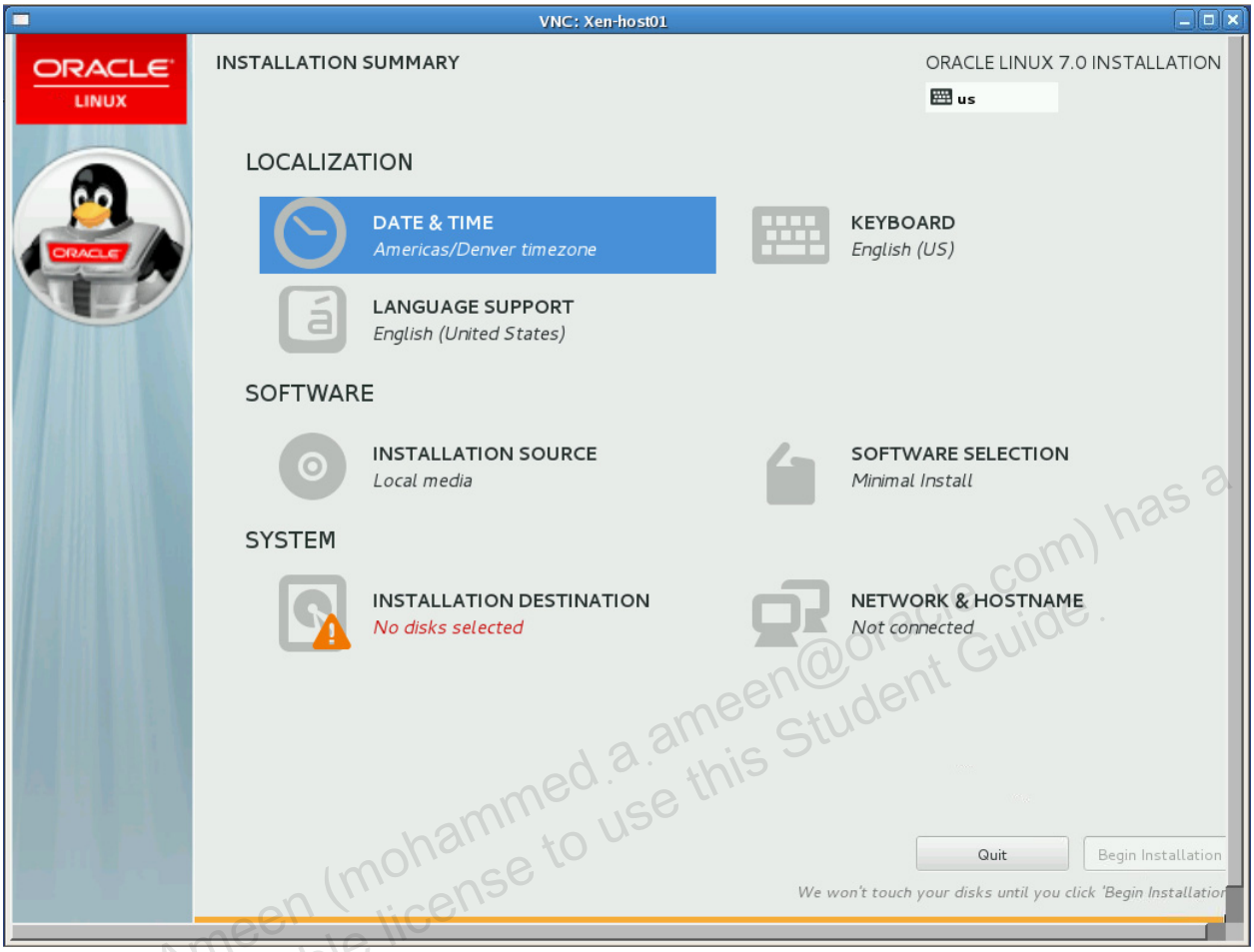
   The DATE & TIME window appears.



   b. 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.
   c. Set the correct date and time at the bottom of this window.
   d. Scroll up if necessary and click **Done**.

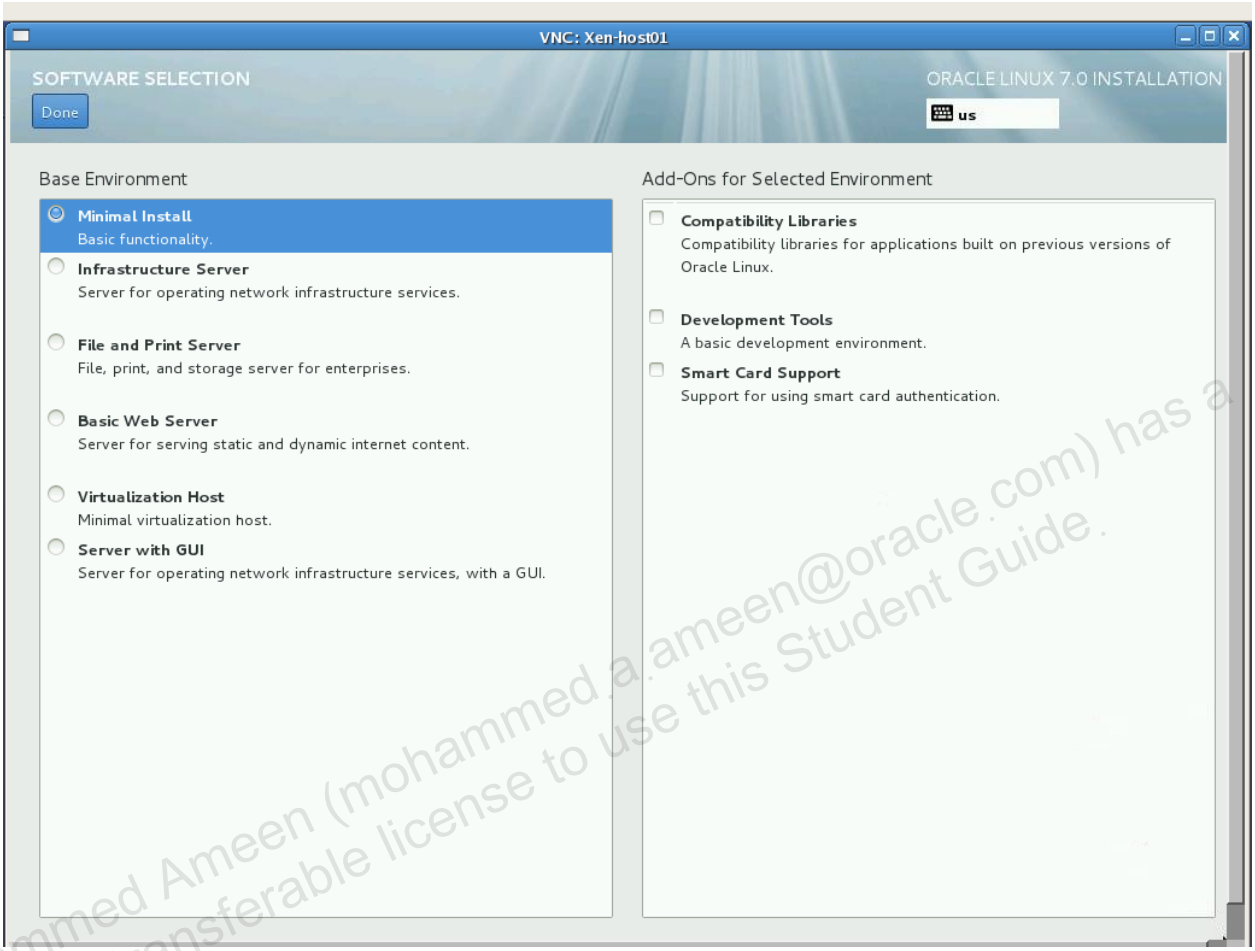
7. The INSTALLATION SUMMARY window appears.



a. Select the LANGUAGE SUPPORT menu option.
   No changes are necessary on this window.
b. Scroll up if necessary and click **Done**. The Installation Summary window appears.
c. Select the KEYBOARD menu option.
   No changes are necessary on this window.
d. Scroll up if necessary and click **Done**. The Installation Summary window appears.
e. Select the INSTALLATION SOURCE menu option.
   No changes are necessary on this window.
f. Scroll up if necessary and click **Done**. The Installation Summary window appears.

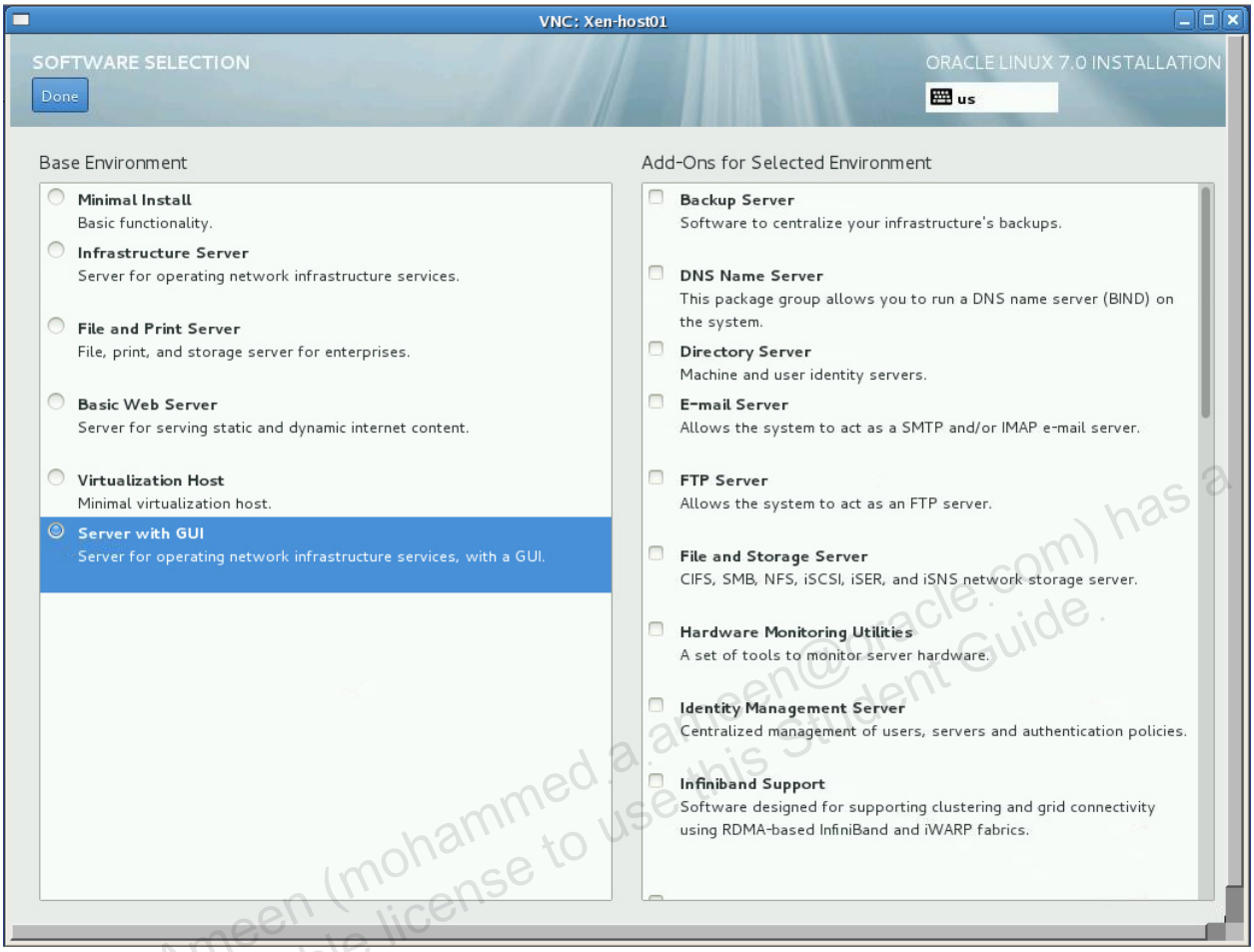
8. Configure SOFTWARE SELECTION.
   a. Select the SOFTWARE SELECTION menu option.

   The SOFTWARE SELECTION window appears.



   b. Select the **Server with GUI** option as the Base Environment.

The Add-Ons for Selected Environment changes as shown:

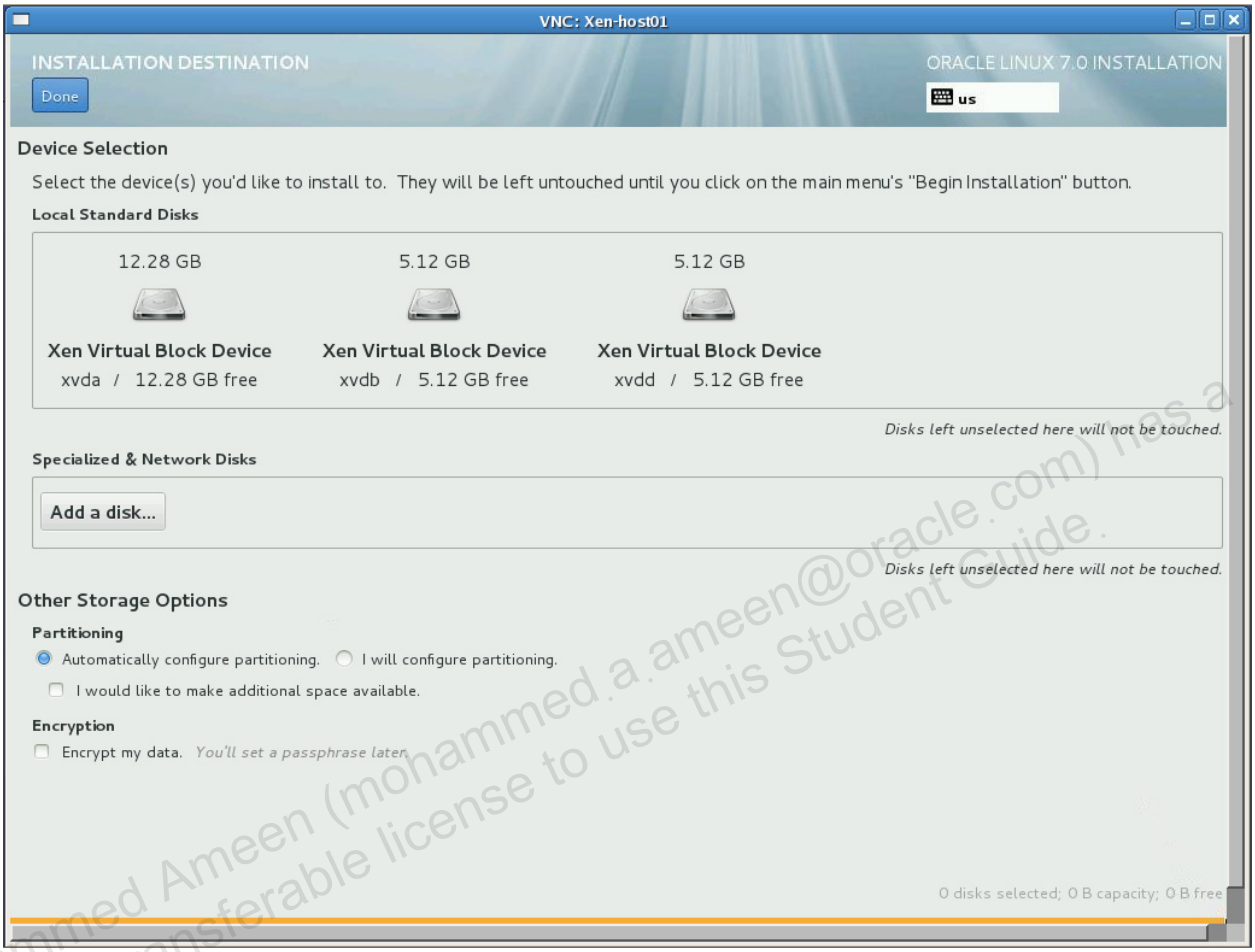


c. Select the following Add-Ons:
   1) FTP Server
   2) File and Storage Server
   3) Network File System Client

d. Scroll up if necessary and click **Done**. The Installation Summary window appears.

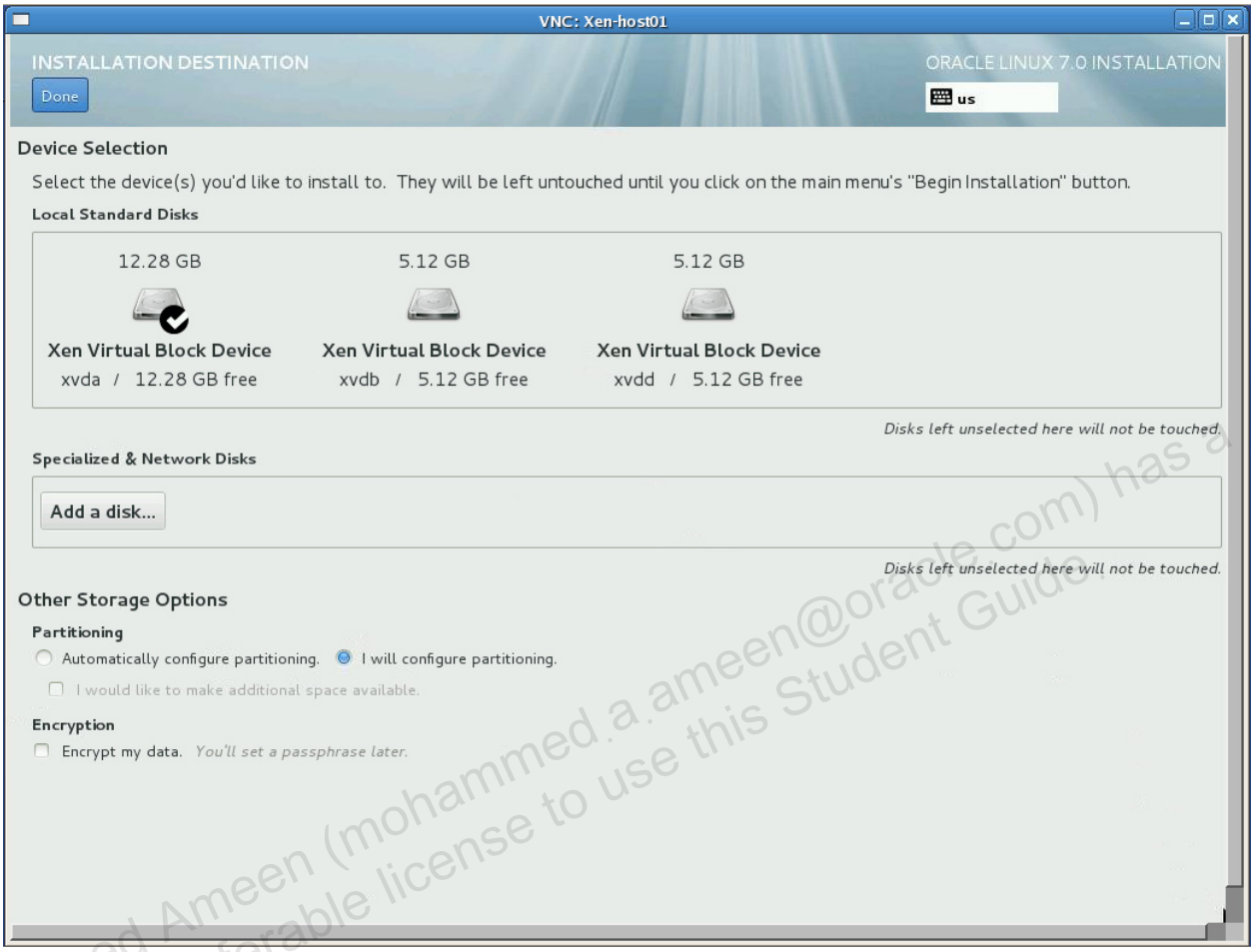
9. Configure INSTALLATION DESTINATION.
   a. Select the INSTALLATION DESTINATION menu option.

   The INSTALLATION DESTINATION window appears.



   b. Under Device Selection, Local Standard Disks, select the **12.28 GB** disk (`xvda`).

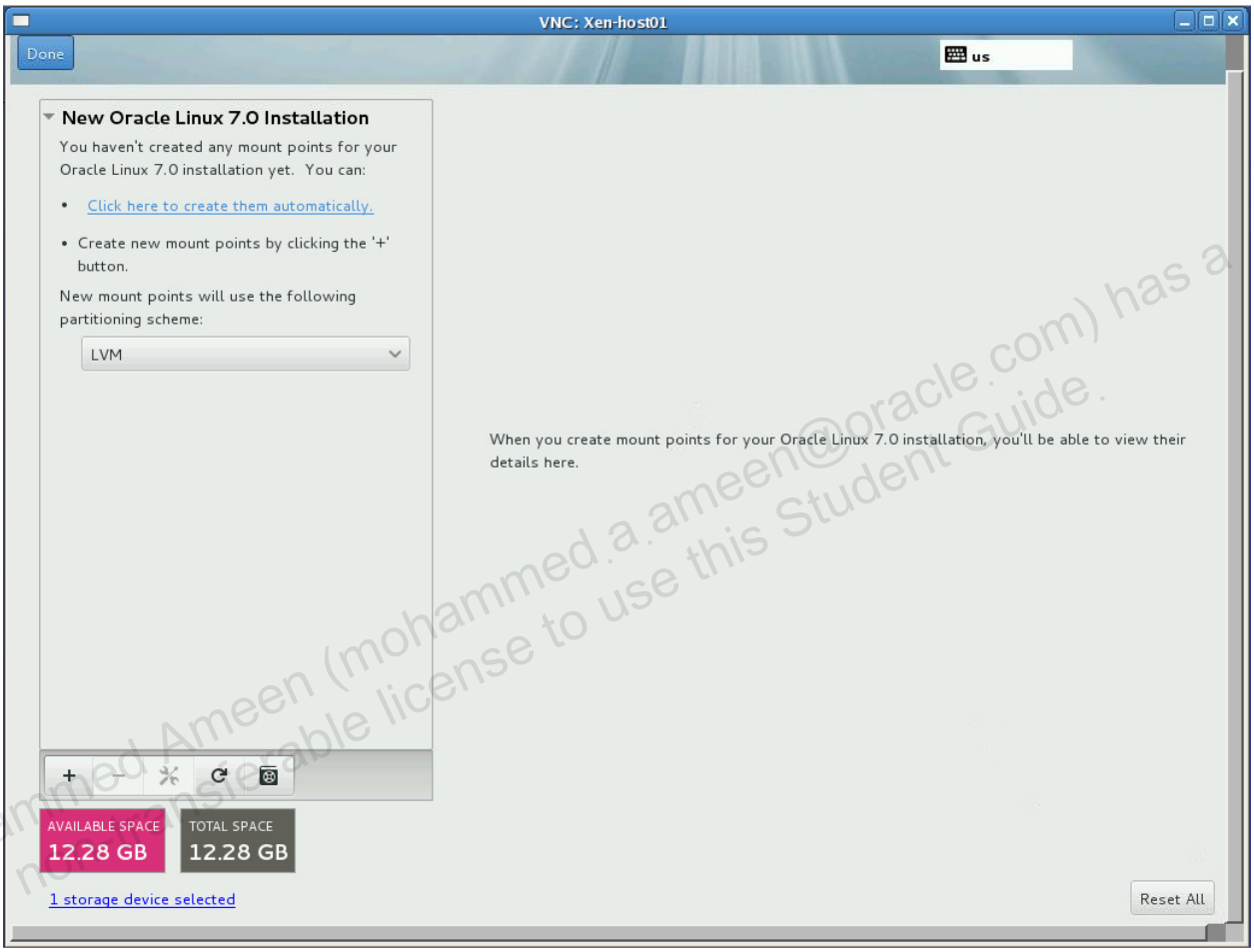
c. Under Other Storage Options, Partitioning, select **I will configure partitioning**. The screen appears as shown, with a checkmark on the 12.28 GB disk.



d. Scroll up if necessary and click **Done**.

10. Manually create the `/boot` mount point with the following characteristics:
    - Mount Point: `/boot`
    - File System Type: **ext4**
    - Size: **500 MB**
    - Partition: **xvda1**

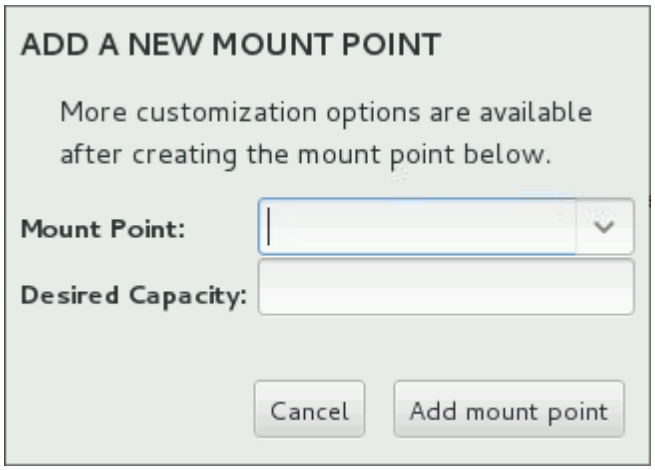
    The following window appears:



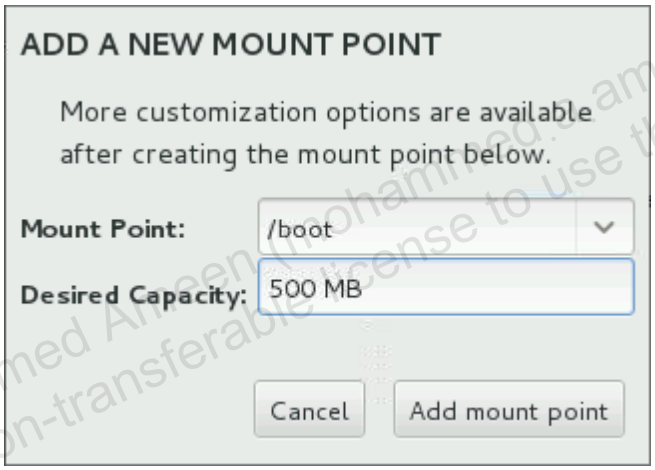
    a. Click the drop-down menu that currently displays LVM to display the following menu:

    b. Select **Standard Partition** from the pop-up menu.

c. Click the + menu option to Add a new mountpoint.
   - The ADD A NEW MOUNT POINT dialog box appears:

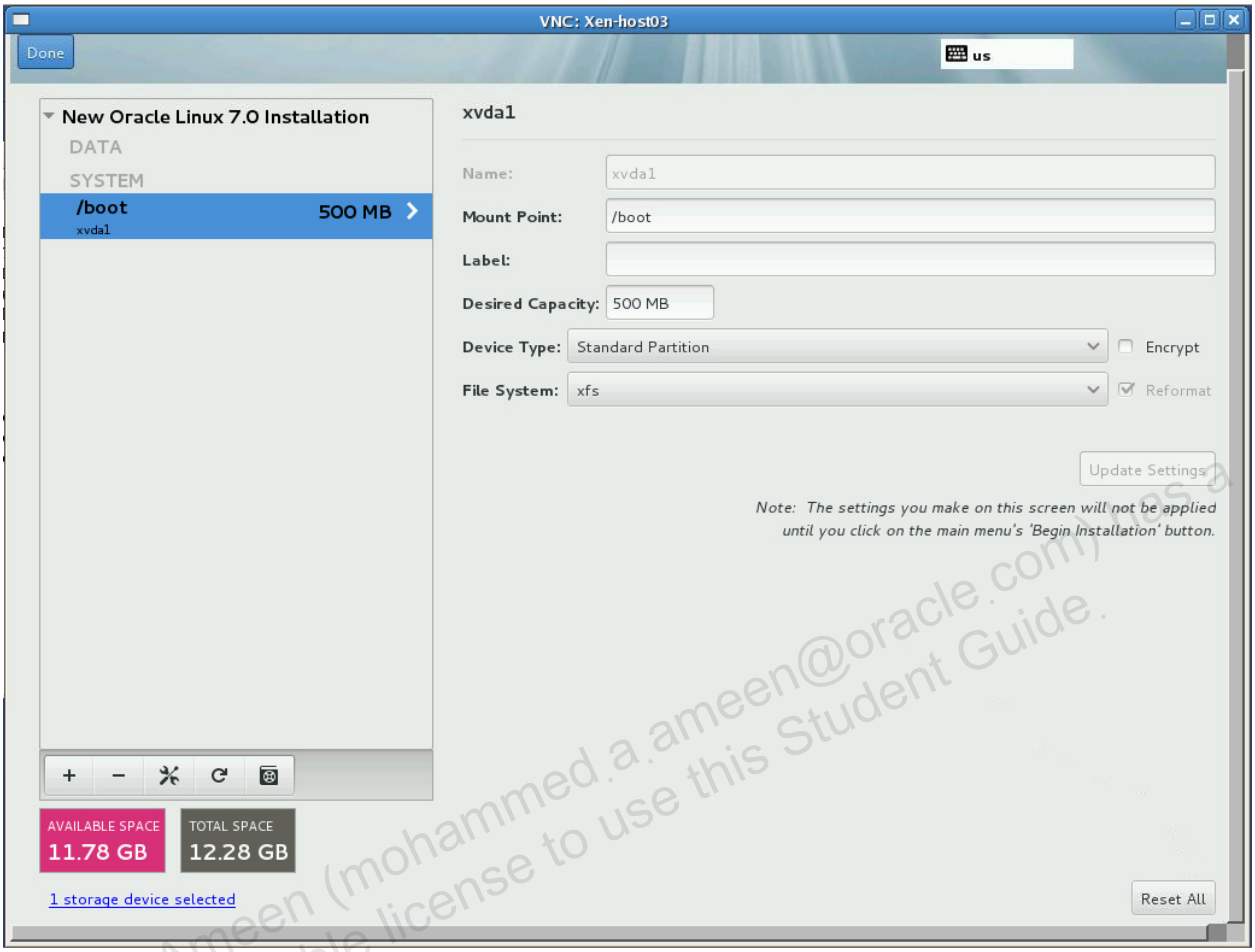


d. Click the down arrow on the **Mount Point** field and select `/boot`.
e. Enter a **Desired Capacity** of **500 MB**.
   - Your window appears as follows:

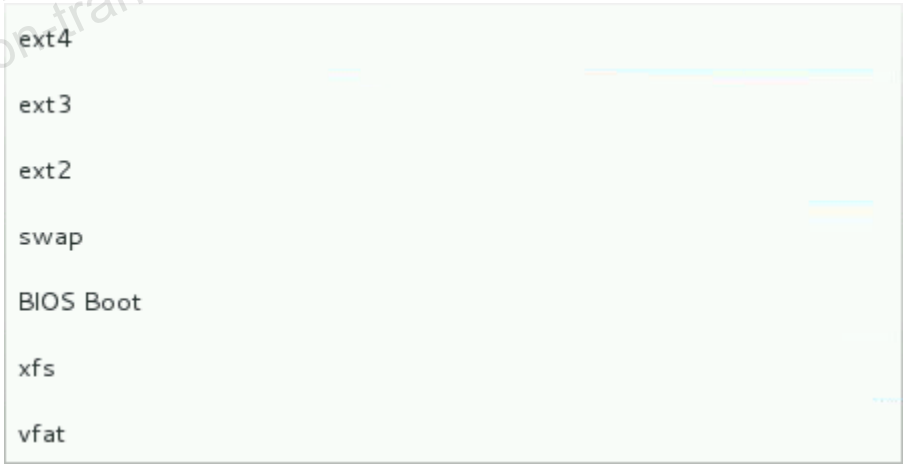


f. Click **Add mount point**.

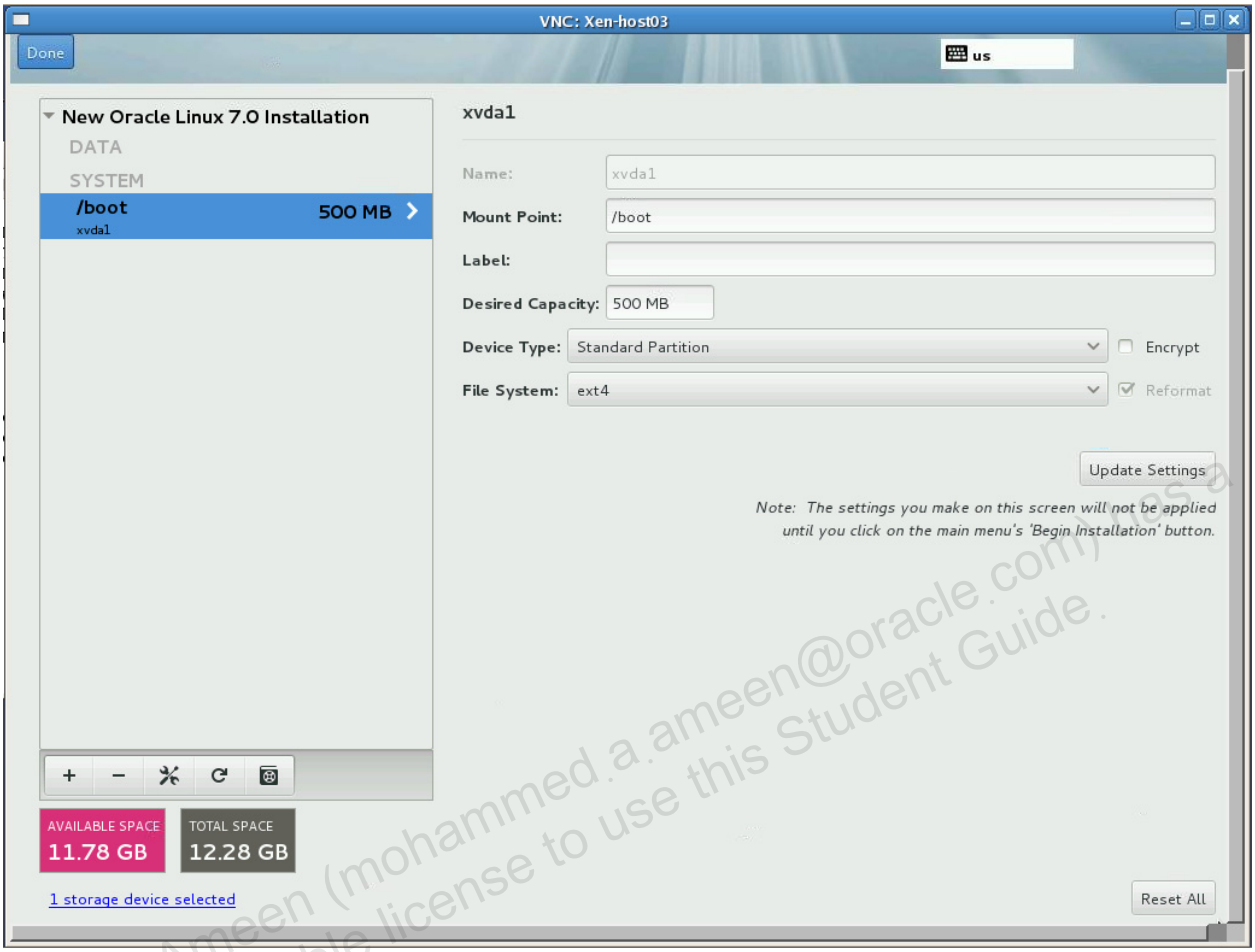
The partitioning window shows the `/boot` mount point on the **xvda1** partition.



g. With `/boot` selected, click the File System down arrow and select **ext4** as the file system type from the pop-up menu.



The partition window now shows File System as **ext4** for `/boot`.



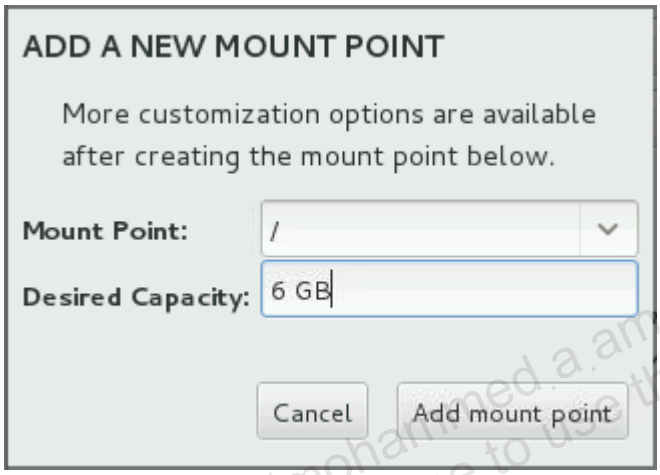
h. Click **Update Settings**.

11. Manually create the root (/) mount point with the following characteristics:
    - Mount Point: /
    - File System Type: **ext4**
    - Size: **6 GB**
    - Partition: **xvda2**

    a. Click the plus (**+**) sign menu option.
    - The ADD A NEW MOUNT POINT window appears.

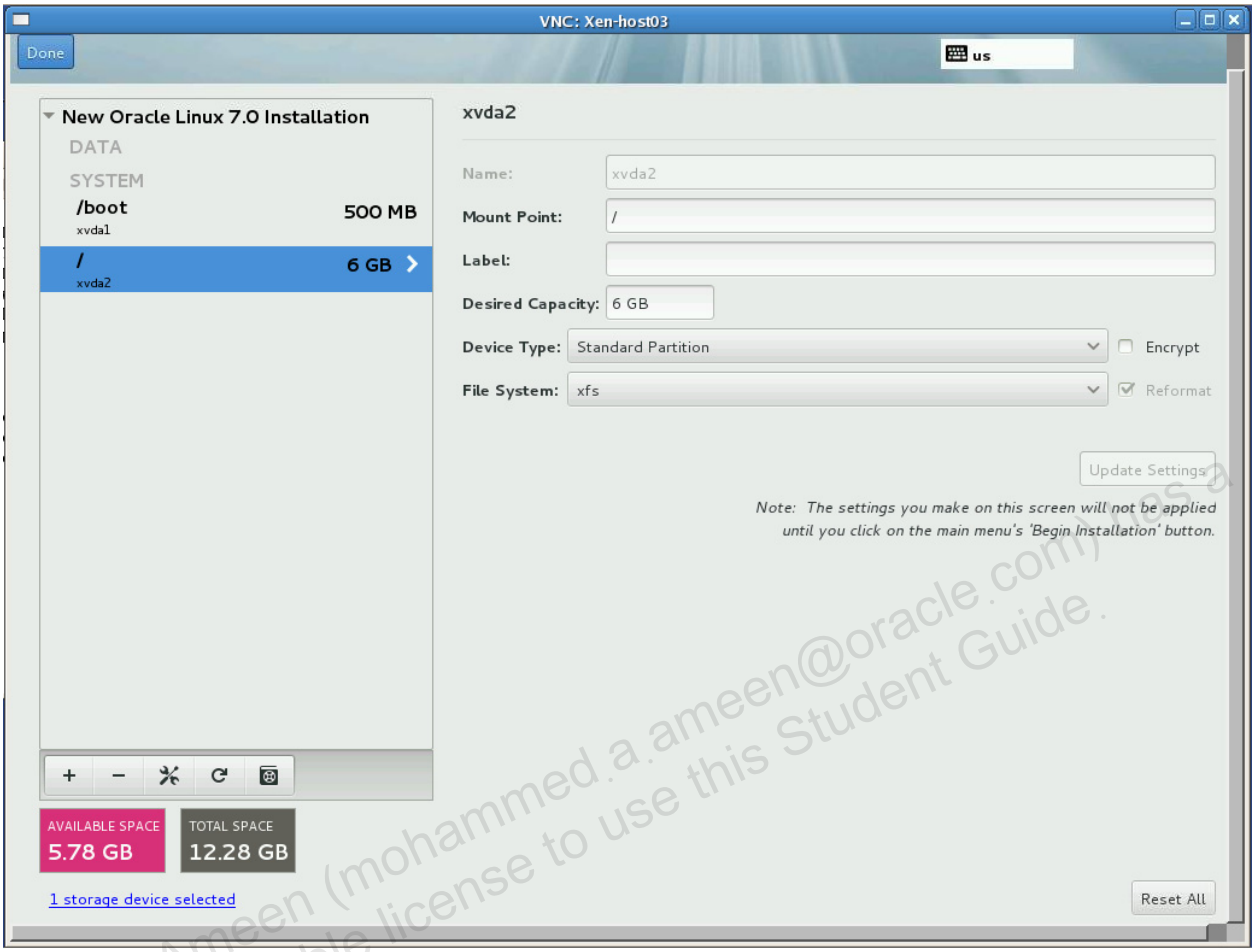
    b. Click the down arrow on the **Mount Point** field and select /.

    c. Enter a **Desired Capacity** of **6 GB**.
    - Your window appears as follows:

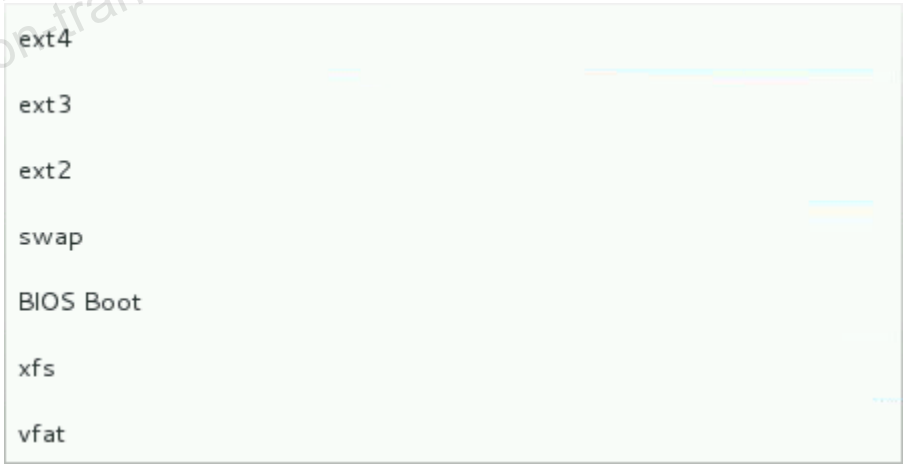


    d. Click **Add mount point**.

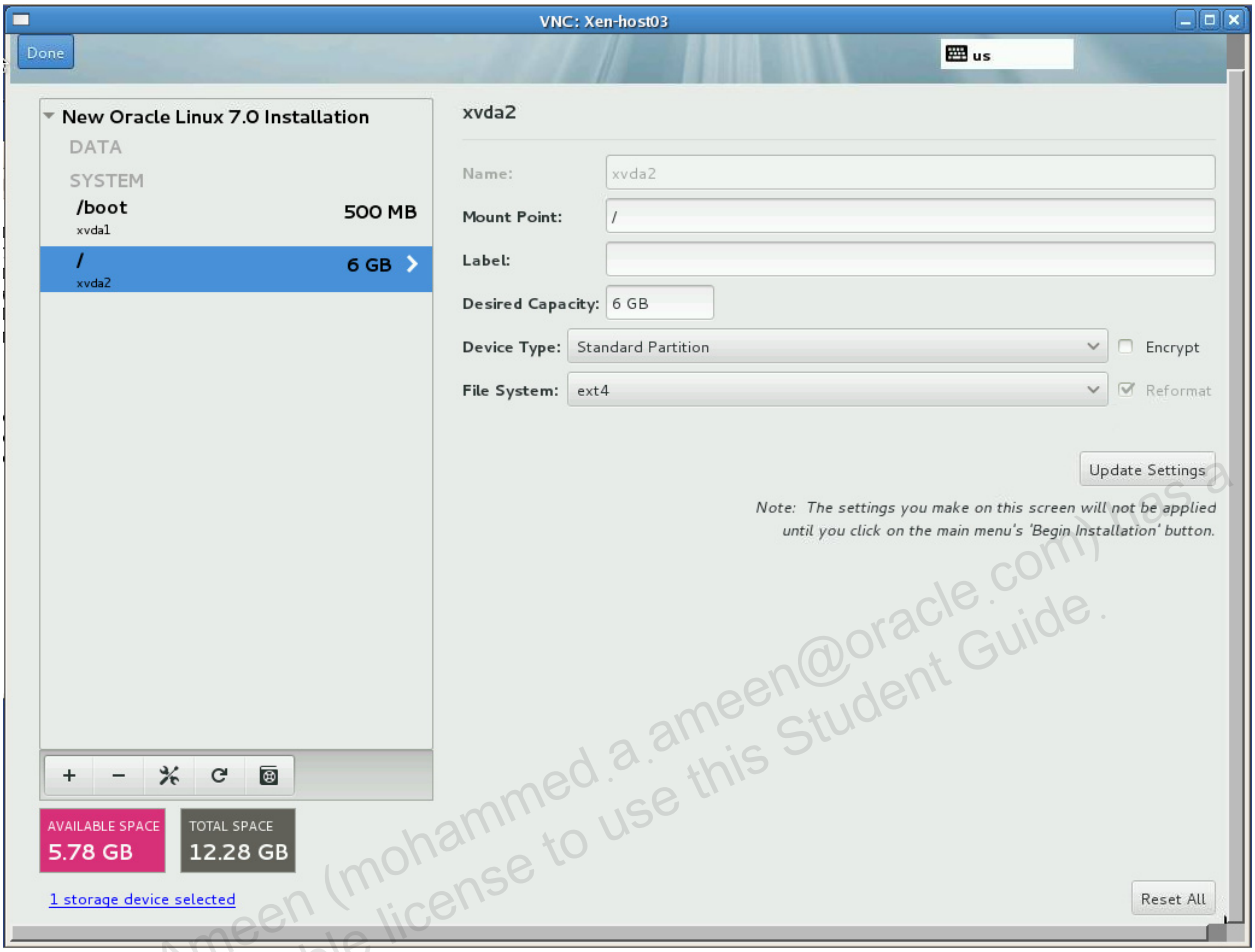
The partitioning window shows the / mount point on the **xvda2** partition.



e. With / selected, click the File System down arrow and select **ext4** as the file system type from the pop-up menu.



The partition window now shows File System as **ext4** for /.



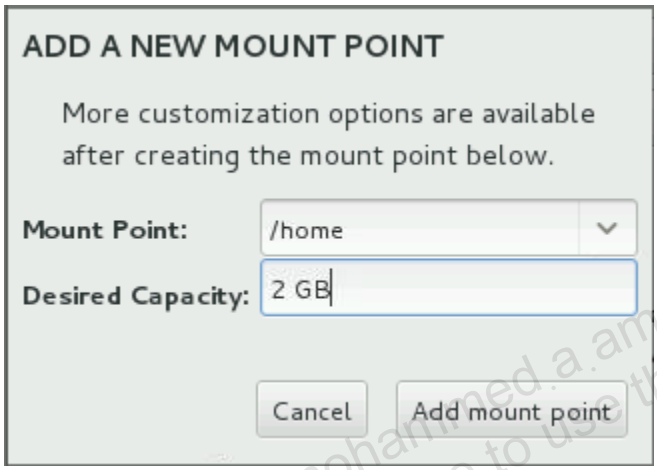
f. Click **Update Settings**.

12. Manually create the `/home` mount point with the following characteristics:
    - Mount Point: `/home`
    - File System Type: **ext4**
    - Size: **2 GB**
    - Partition: **xvda3**

    a. Click the plus (**+**) sign menu option.
    - The ADD A NEW MOUNT POINT window appears.

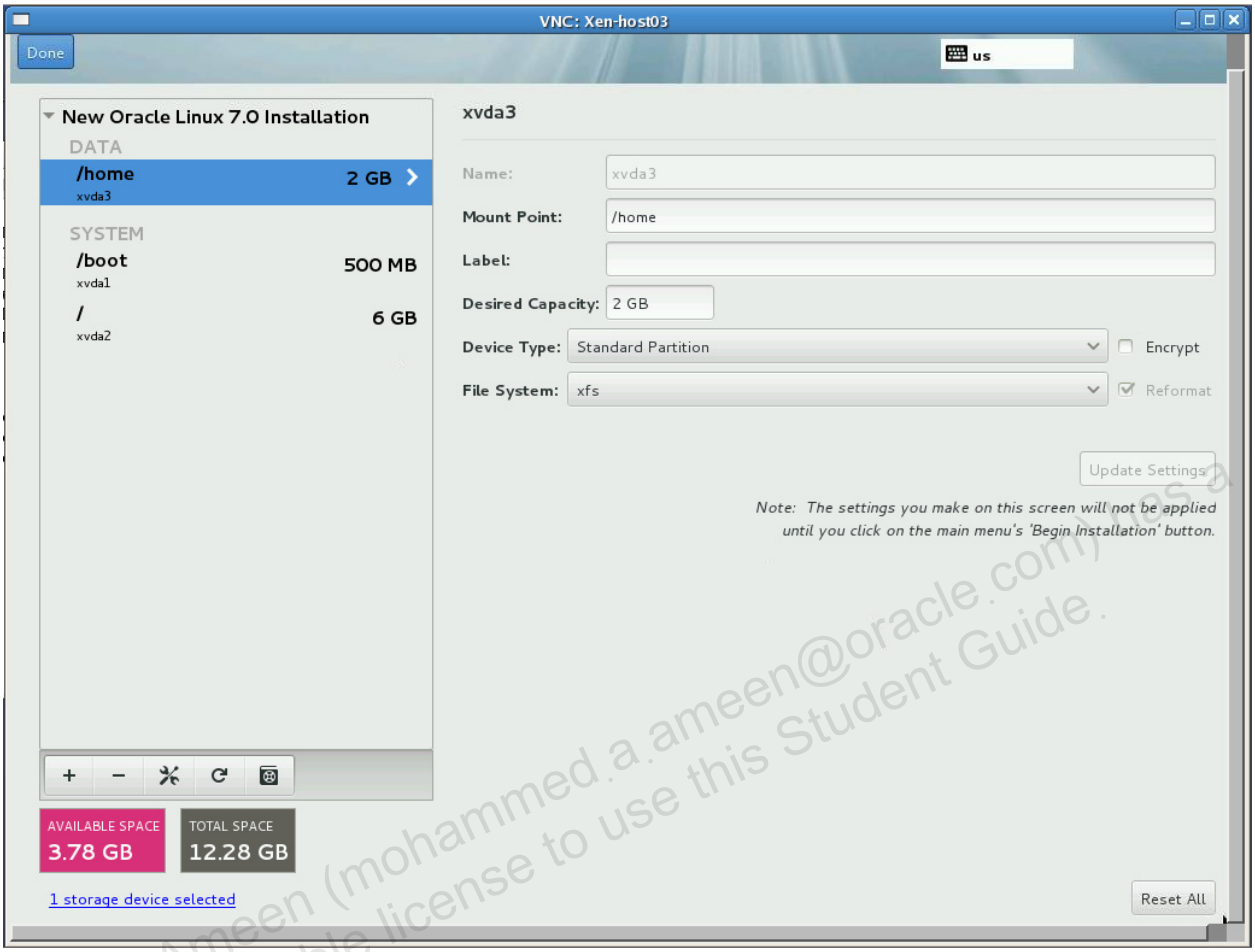
    b. Click the down arrow on the **Mount Point** field and select `/home`.

    c. Enter a **Desired Capacity** of **2 GB**.
    - Your window appears as follows:

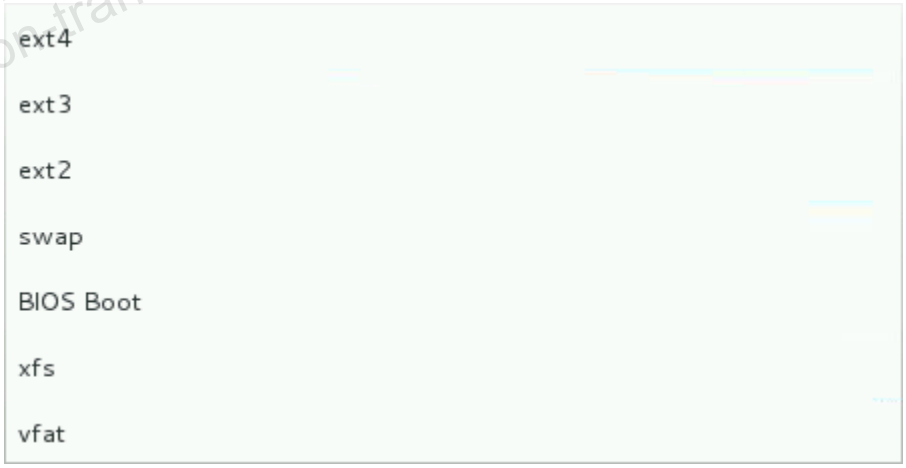


    d. Click **Add mount point**.

The partitioning window shows the /home mount point on the **xvda3** partition.

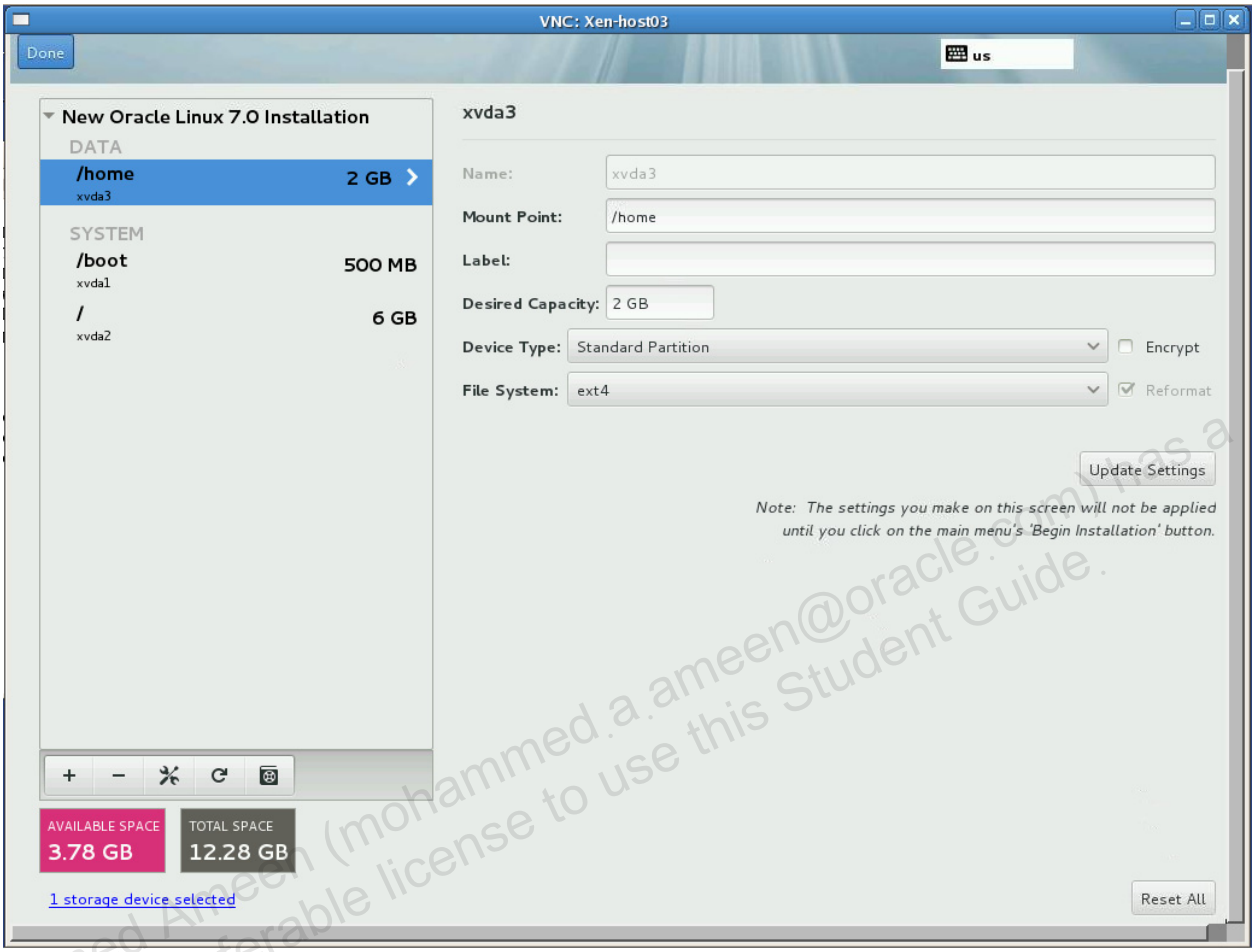


e. With /home selected, click the File System down arrow and select **ext4** as the file system type from the pop-up menu.



f. The partition window now shows File System as **ext4** for `/home`.

- Notice the AVAILABLE SPACE is 3.78 GB.



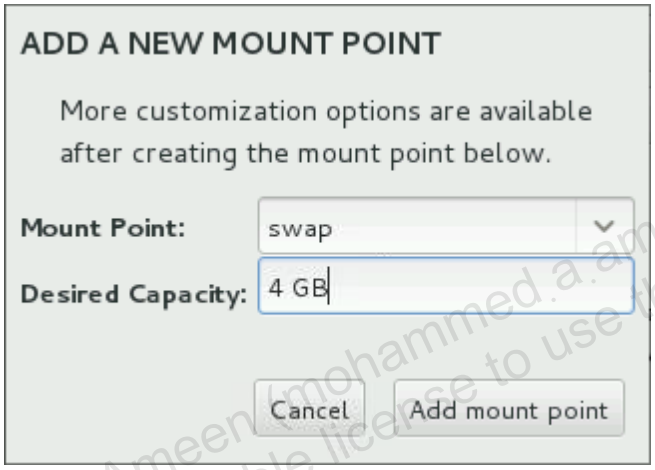
g. Click **Update Settings**.

13. Manually create the `swap` partition with the following characteristics:
    - Mount Point: `swap`
    - File System Type: **swap**
    - Size: **Remaining available space**
    - Partition: **xvda5**

    a. Click the plus (**+**) sign menu option.
    - The ADD A NEW MOUNT POINT window appears.

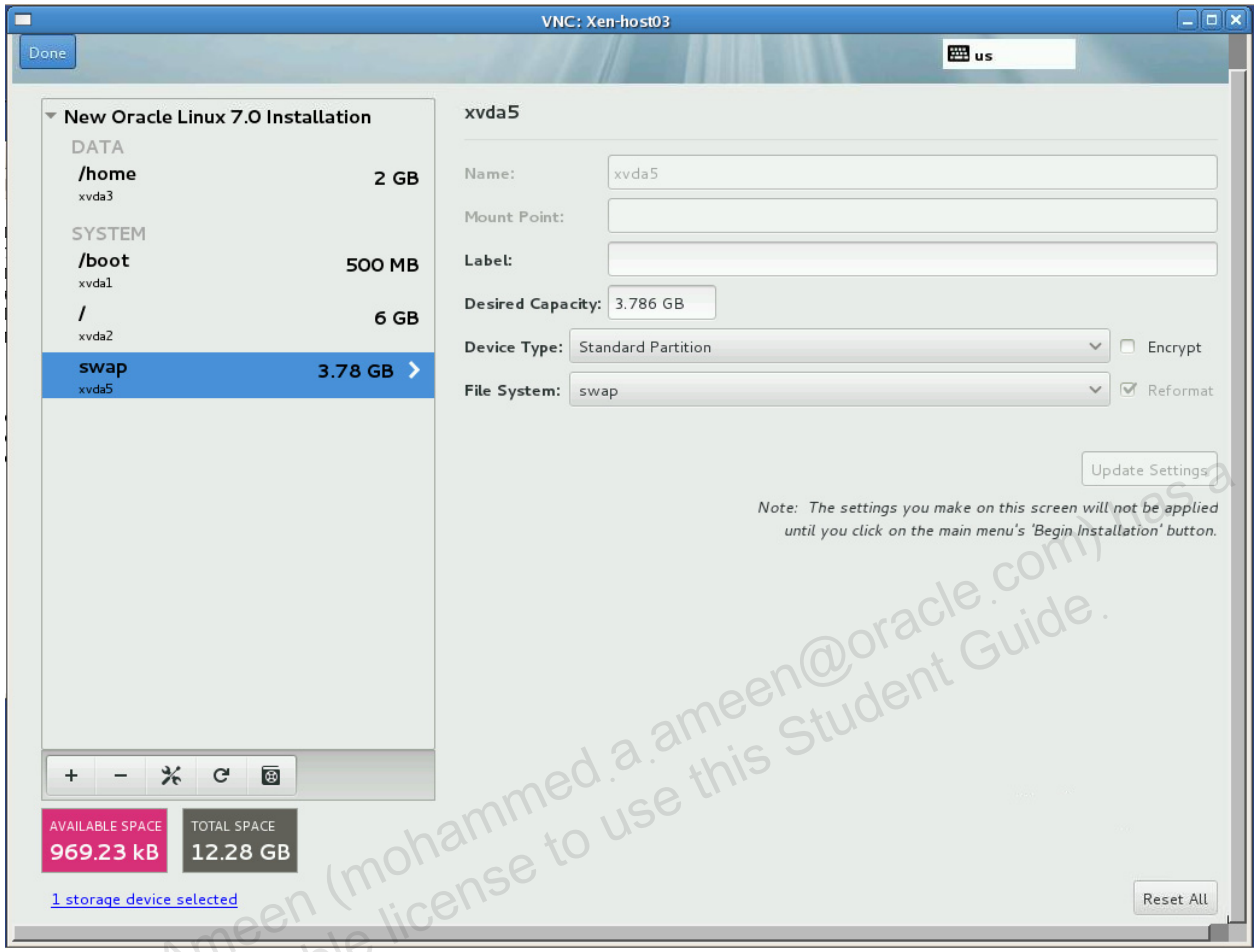
    b. Click the down arrow on the **Mount Point** field and select `swap`.

    c. Enter a **Desired Capacity** of **4 GB**.
    - The AVAILABLE SPACE is 3.78 GB.
    - Entering a size greater than the available space adjust as necessary.
    - Your window appears as follows:



    d. Click **Add mount point**.

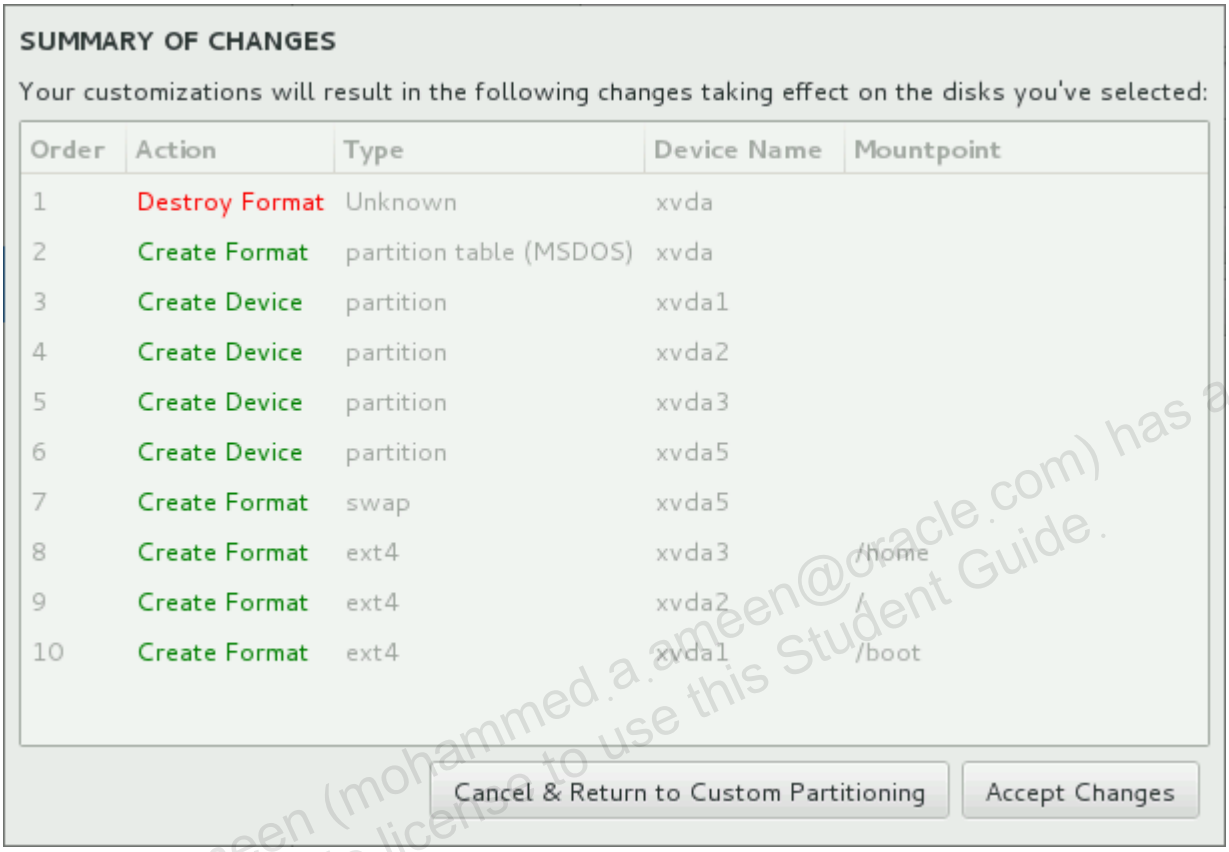
The partitioning window shows `swap` on the **xvda5** partition.



- Notice the size adjusted to the available space of 3.78 GB.

14. Complete manual partitioning.
    a. Scroll up if necessary and click **Done**.

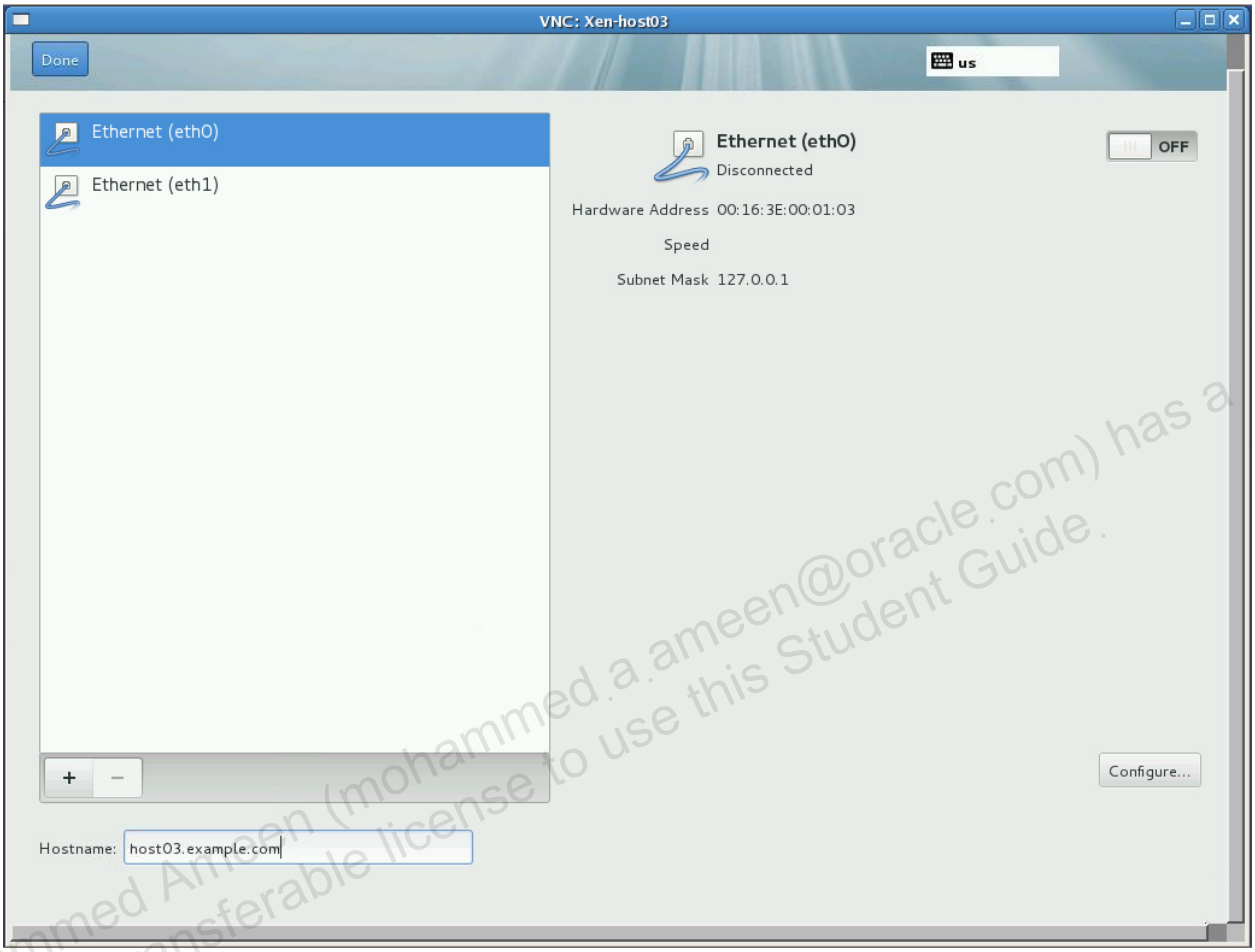
    The following SUMMARY OF CHANGES window appears:



    b. Click **Accept Changes**.
    - The Installation Summary window appears.

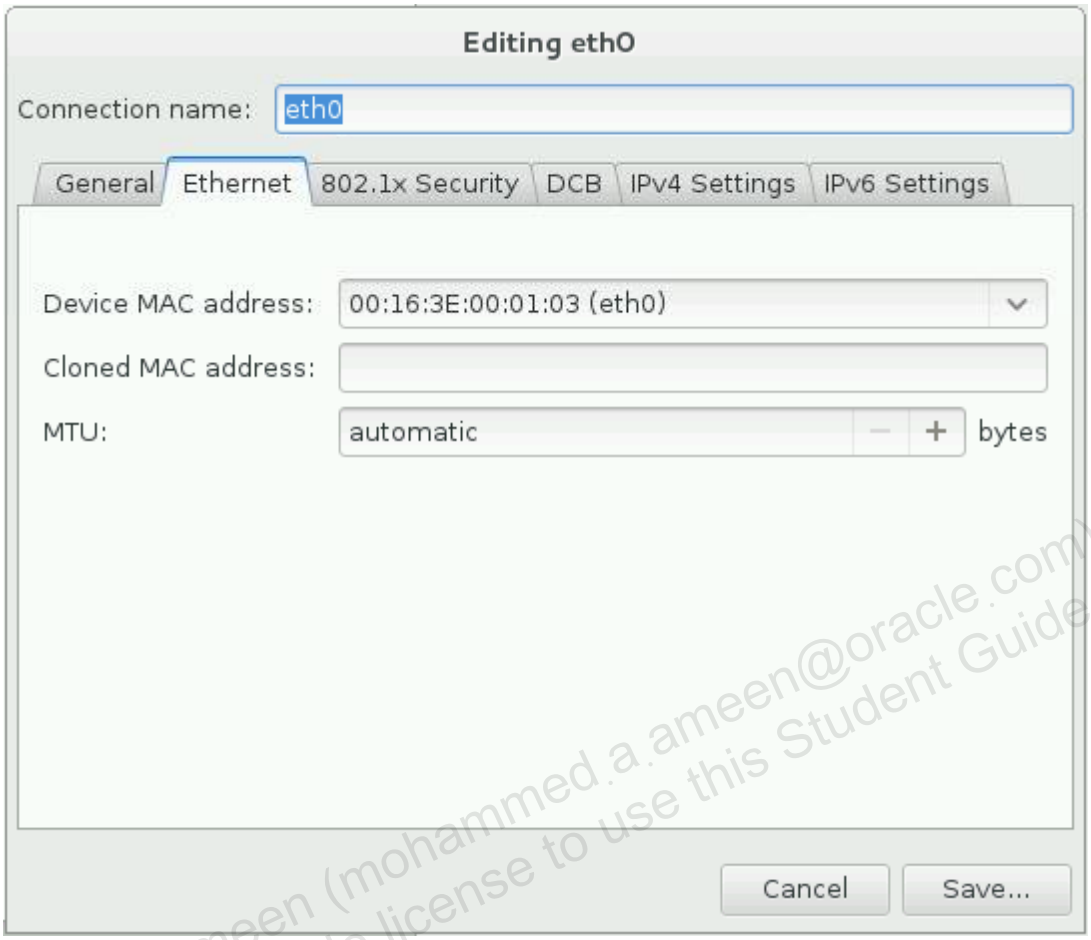
15. Configure NETWORK & HOSTNAME.
    a. Select the NETWORK & HOSTNAME menu option.

The NETWORK & HOSTNAME window appears.



    a. Enter `host03.example.com` as the **Hostname** as shown.
    b. With **Ethernet (eth0)** selected, click the OFF switch to toggle to ON.
    c. Click the **Configure…** button.

The following window is shown.



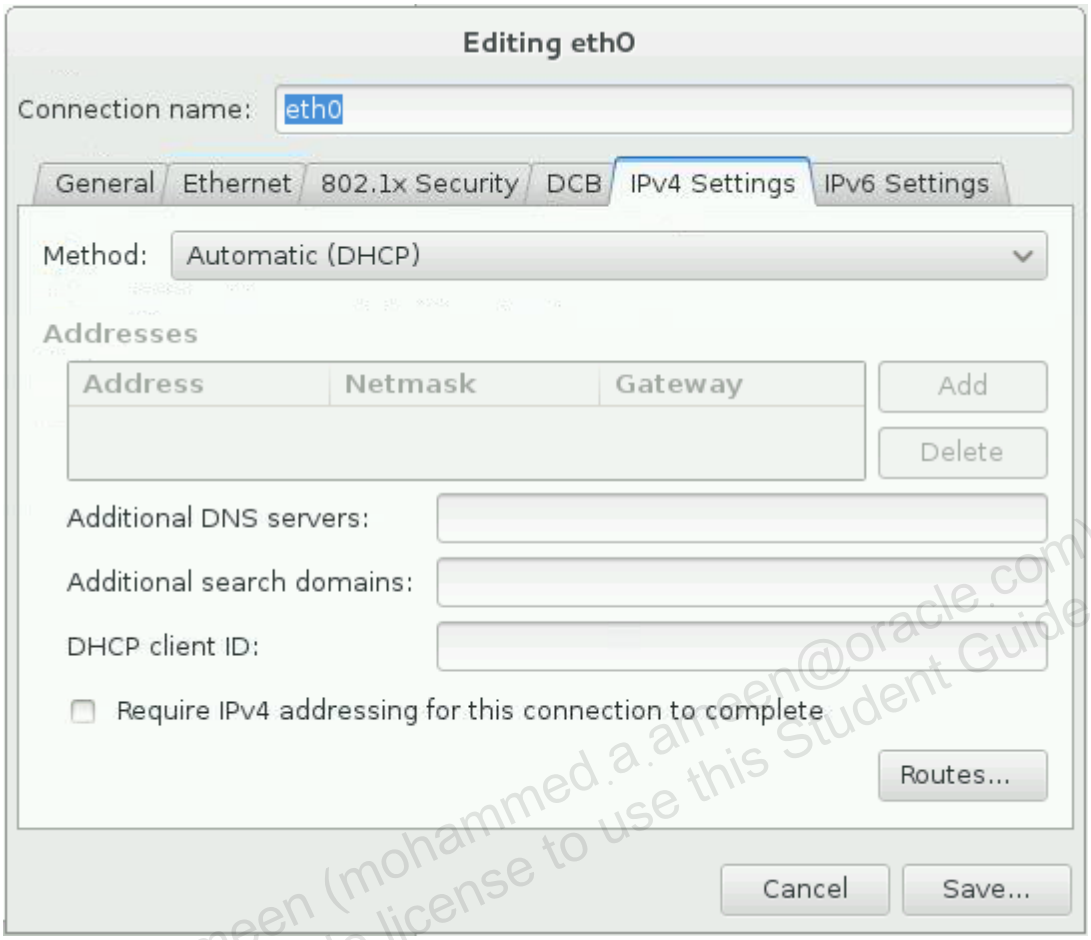
d. Click the **General** tab.

The following window is shown.

e. Click the **Automatically connect to this network when it is available** check box as shown.

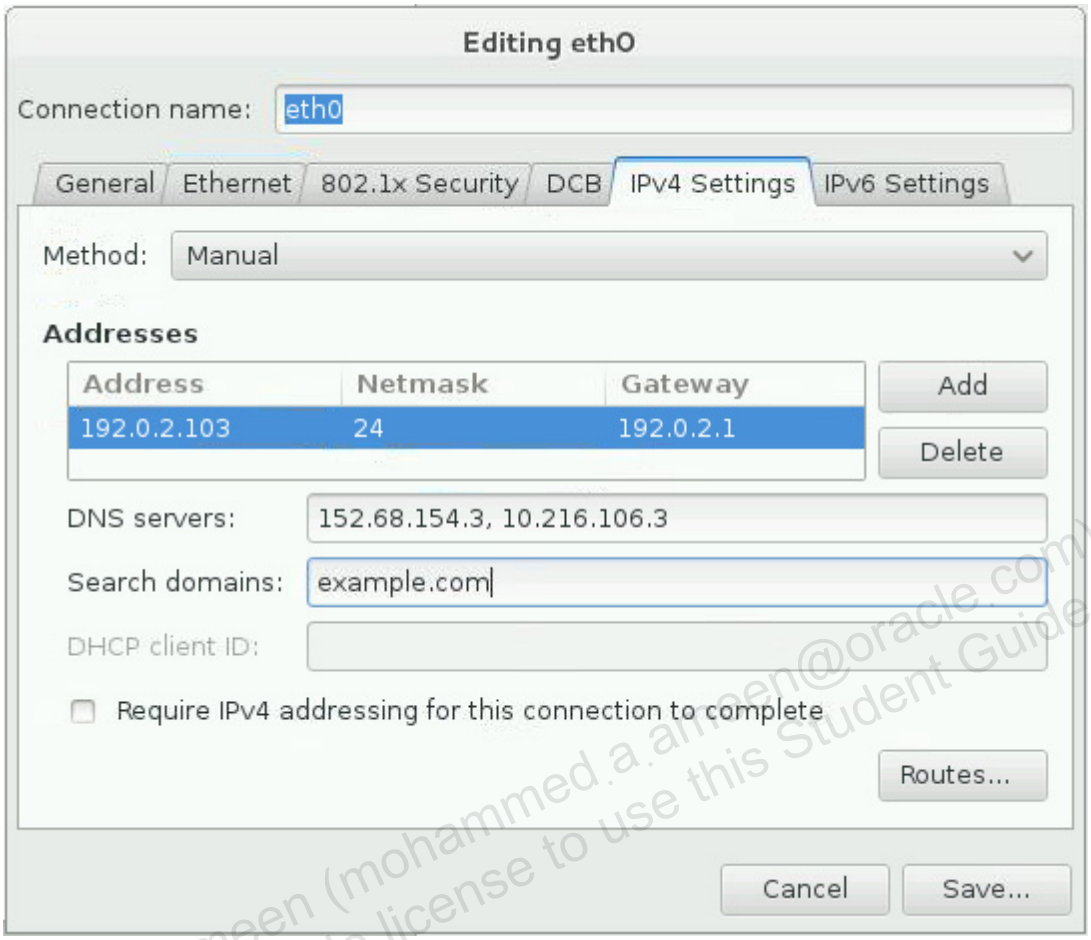
f. Click the **IPV4 Settings** tab.

The following window is shown.



g. Select **Manual** from the drop-down **Method** menu.
h. Click the **Add** button to add the following IPV4 address, Netmask, and Gateway address:
   1) **Address**=192.0.2.103
   2) **Netmask**=24
   3) **Gateway**=192.0.2.1
i. Enter **Additional DNS servers**:
   1) 152.68.154.3,10.216.106.3
j. Enter **Additional search domains**:
   1) example.com

Ensure the window appears as shown.

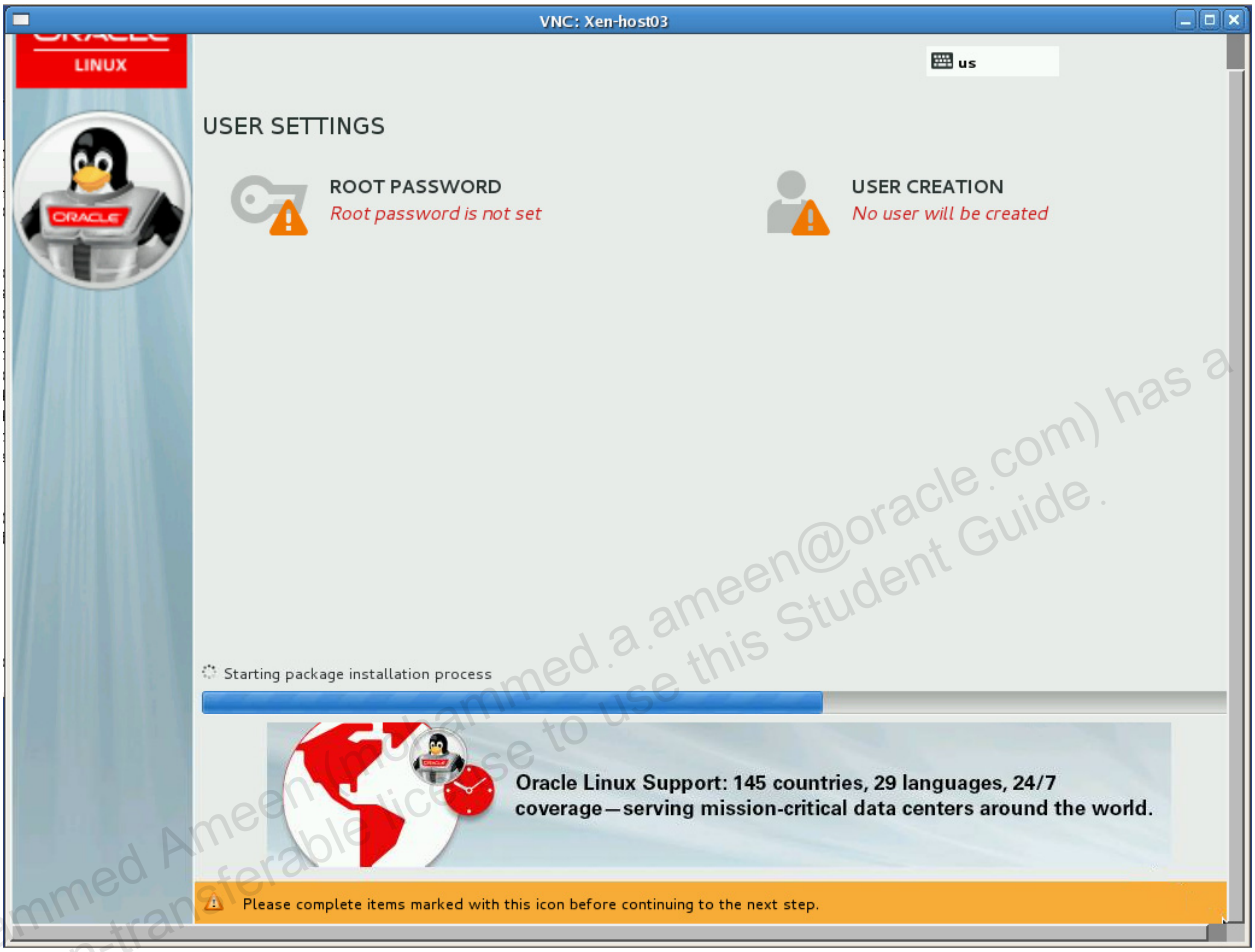


k. Click the **Save…** button.
   - The NETWORK & HOSTNAME window appears.

l. Scroll up if necessary and click **Done**.
   - The INSTALLATION SUMMARY window appears.

16. Begin the installation.
    a. From the INSTALLATION SUMMARY window, scroll down if necessary and click **Begin Installation**.

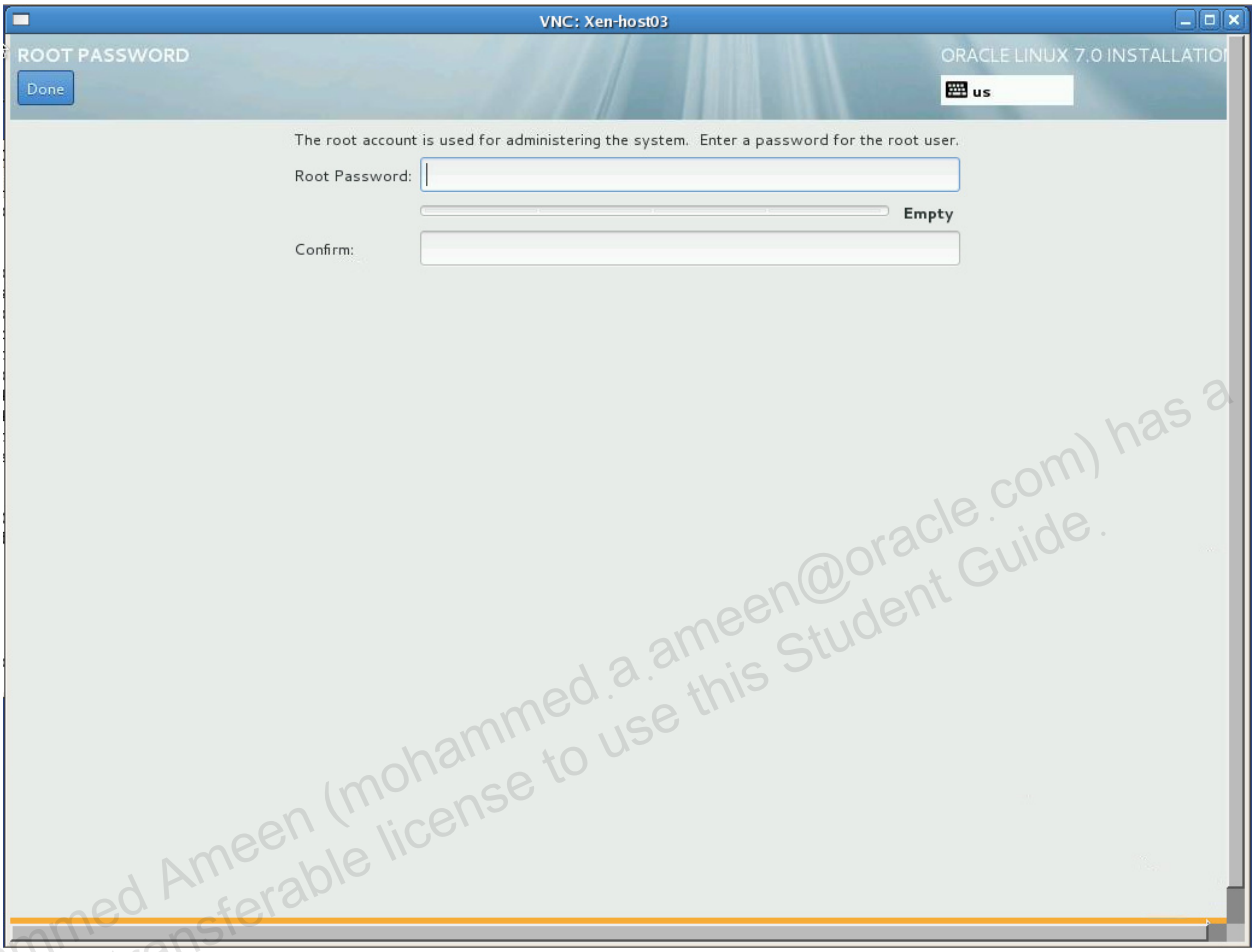
    The CONFIGURATION window appears.



    Note that a status bar indicates packages are being installed.

17. Set the `root` user password.
    a. Click ROOT PASSWORD.

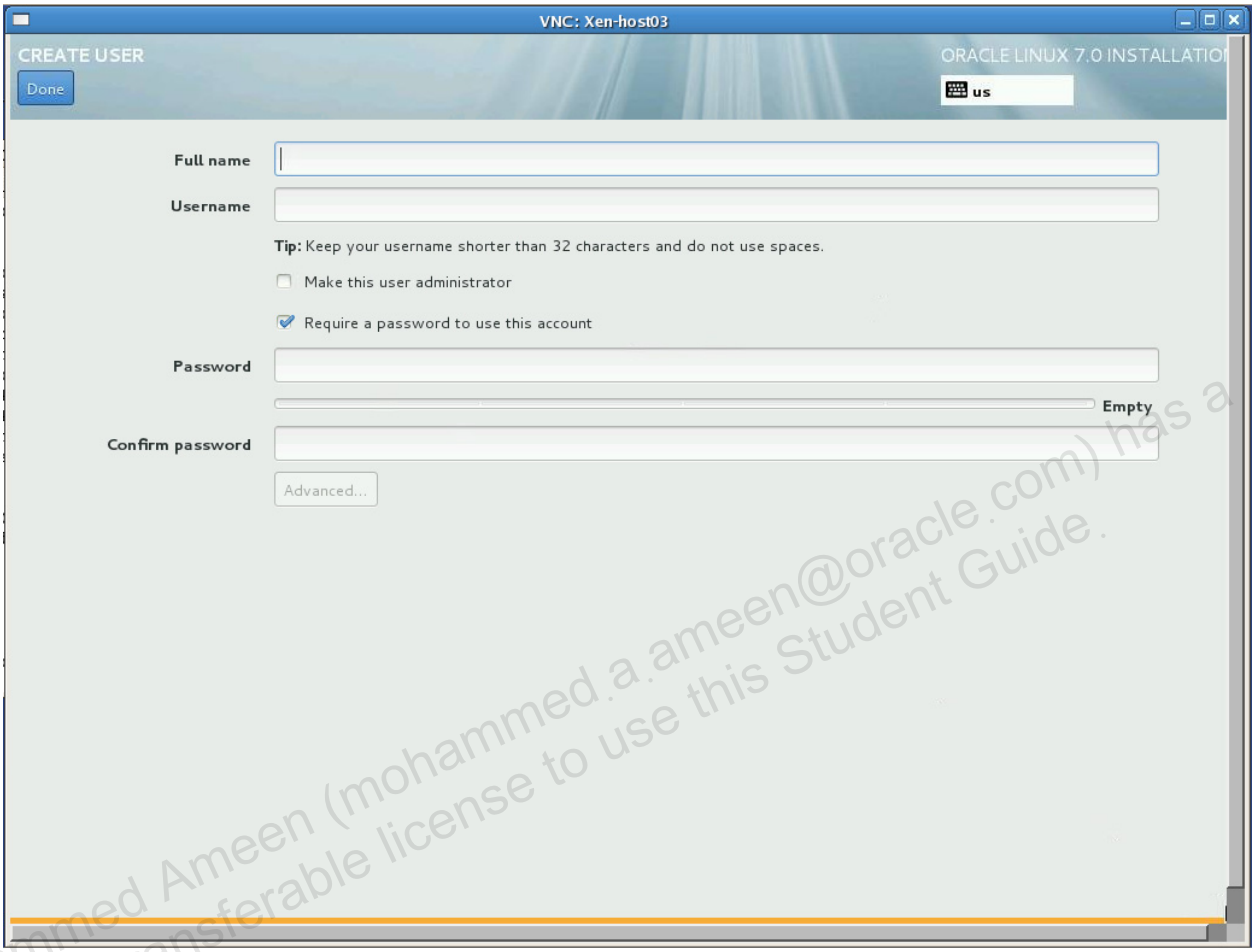
The ROOT PASSWORD window appears.



    b. Enter the Root Password `oracle` for the `root` user.
    c. Press Tab or click the **Confirm** field and re-enter the same password `oracle`.
    d. Scroll up if necessary and click **Done**.
    e. Click **Done** a second time.
        - The CONFIGURATION window appears.

18. Create a non-root user.
    a. Click USER CREATION.

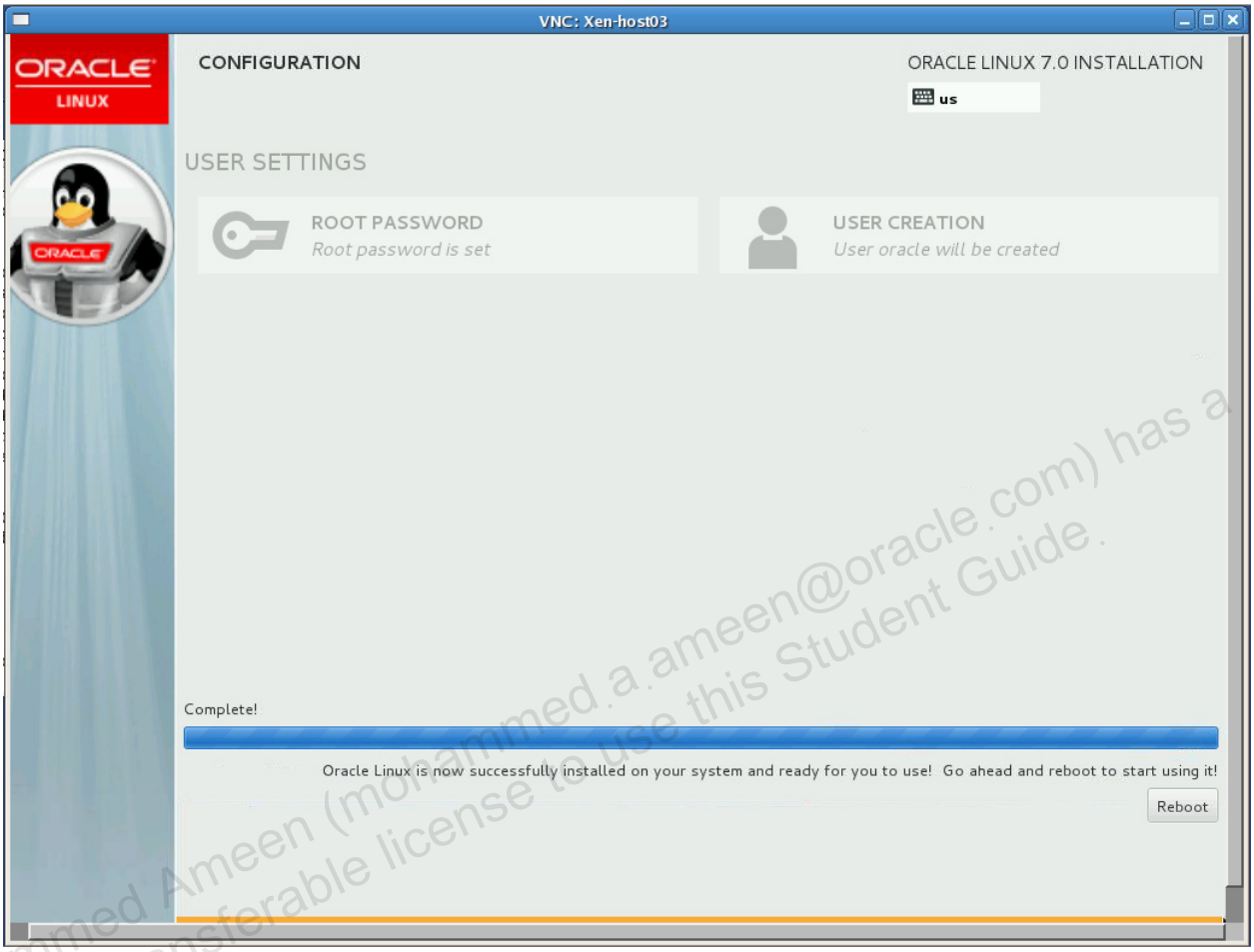
    The CREATE USER window appears.



    b. Create a user with the following attributes:
        - Full Name: `Oracle Student`
        - Username: `oracle`
        - Password: `oracle`
        - Confirm Password: `oracle`
    c. Scroll up if necessary and click **Done**.
    d. Click **Done** a second time.
        - The CONFIGURATION window appears.

19. Reboot after the installation completes.

    The installation continues for a few minutes. When complete, a Reboot option appears at the bottom of the CONFIGURATION window.



    a. Click **Reboot**.

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

Practice 3-2: Completing Initial Setup and FirstBoot

Overview

In this practice, you complete the Initial Setup and FirstBoot.

Assumptions

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

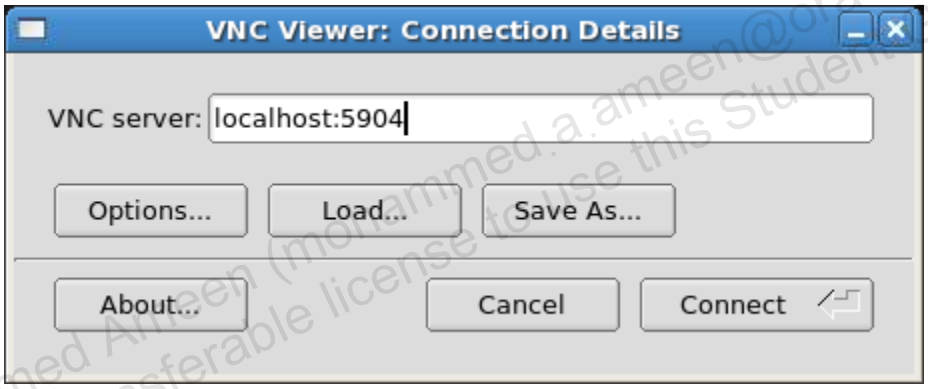
Tasks

1. Connect to the **host03** guest by using **vncviewer**.
   a. From **dom0**, run the `vncviewer&` command.

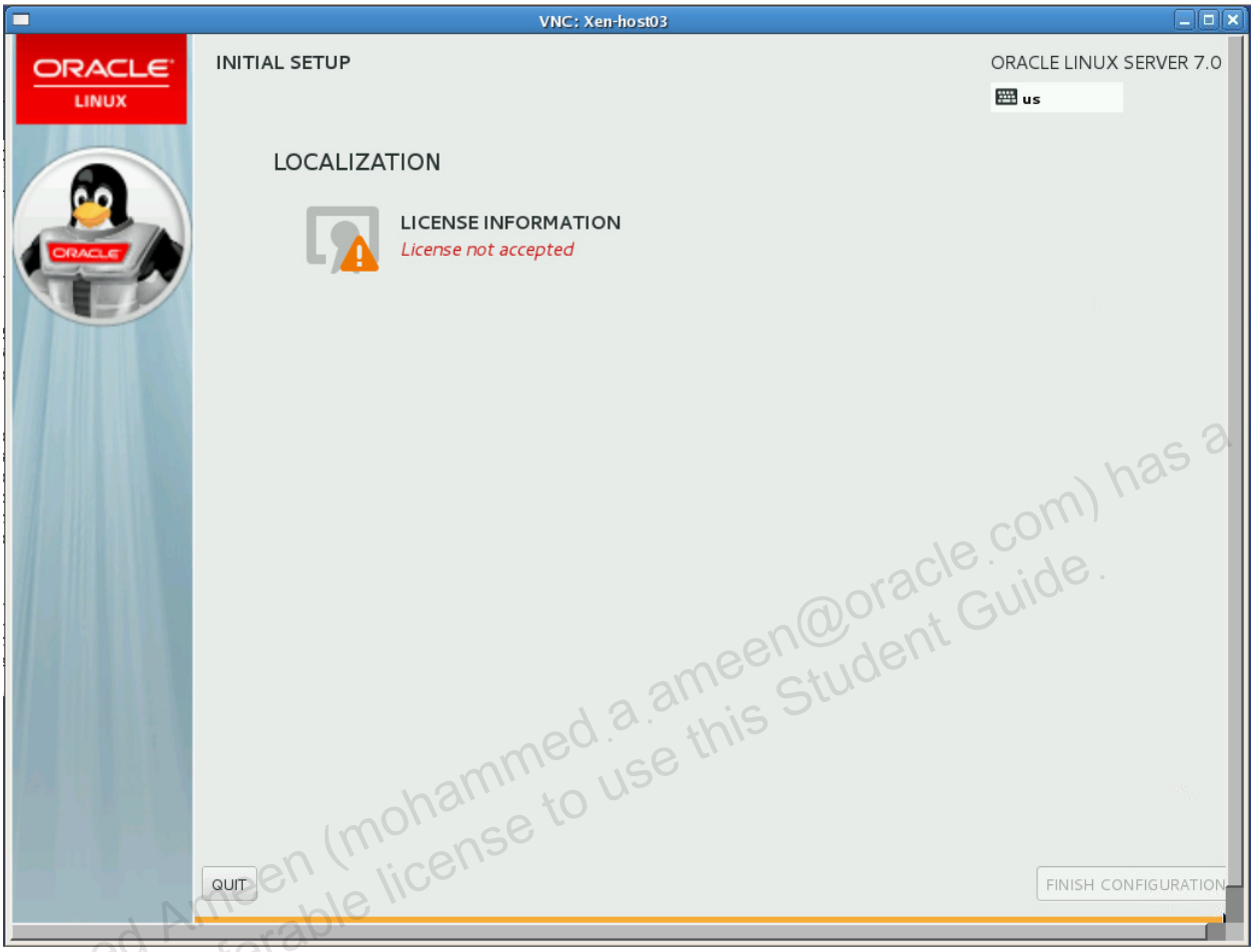
```
[dom0]# vncviewer&
```

   - 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 `5904`, enter `localhost:5904` and click **Connect** (as shown in the following screenshot):



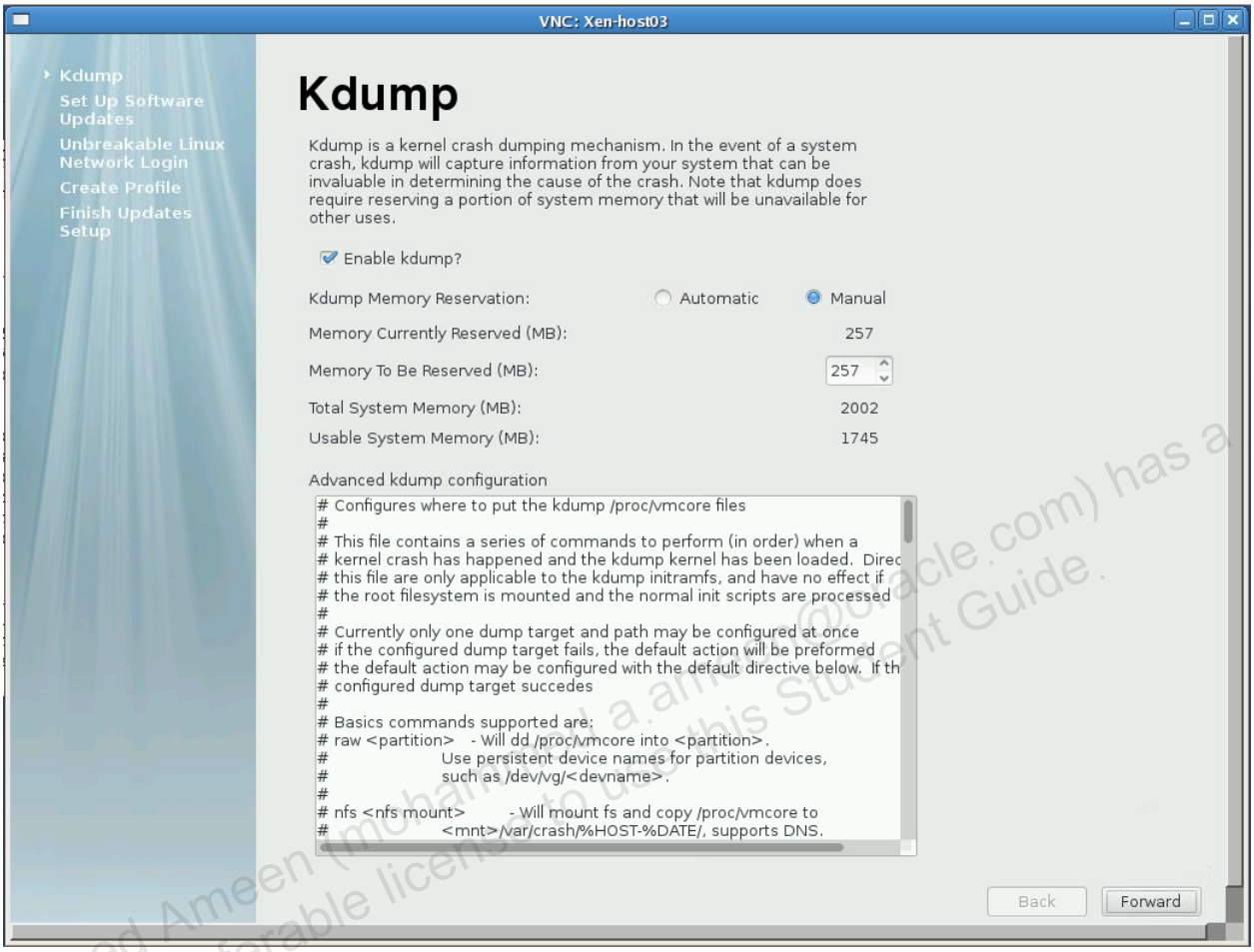
2. Complete INITIAL SETUP.
   The INITIAL SETUP window appears.



   a. Click LICENSE INFORMATION.
      - The License Agreement appears.
   b. Click **I accept the license agreement**.
   c. Scroll up if necessary and click **Done**.
      - The INITIAL SETUP window appears.
   d. Scroll down if necessary and click **FINISH CONFIGURATION**.

3. Complete FirstBoot.

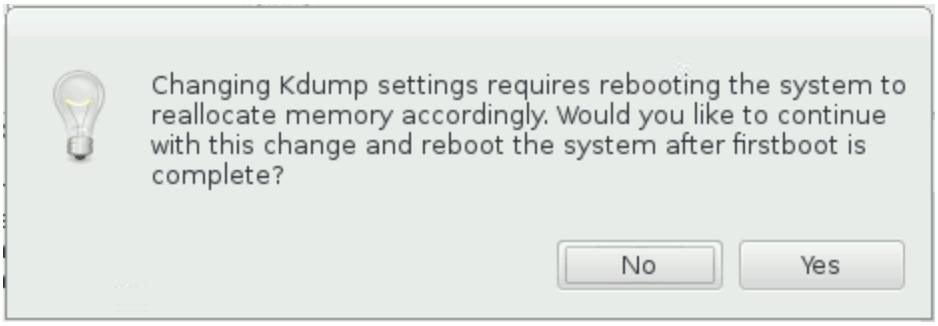
The **Kdump** window appears.



a. Deselect **Enable kdump?**

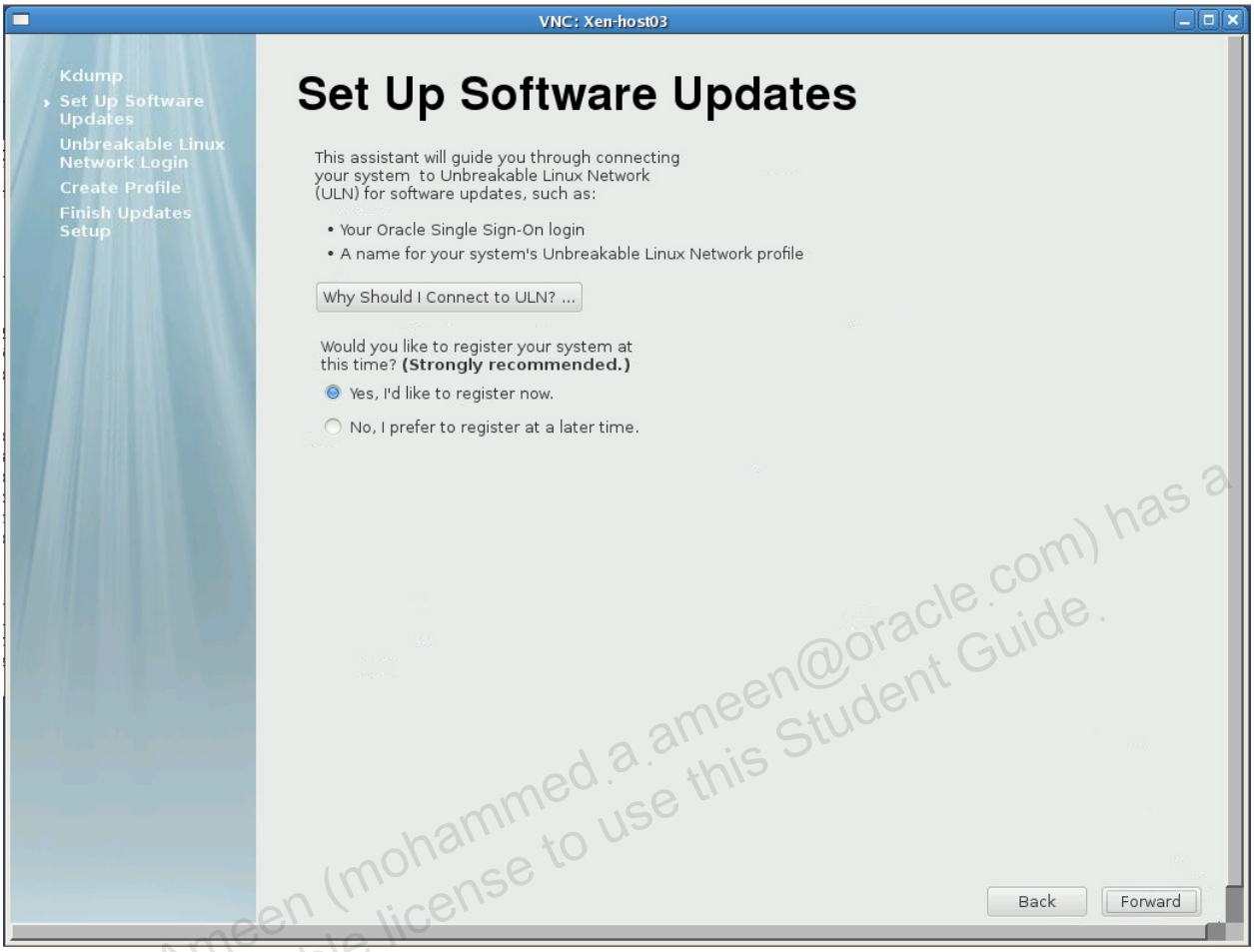
b. Scroll down if necessary and click **Forward**.

The following window appears:



c. Click **Yes**.

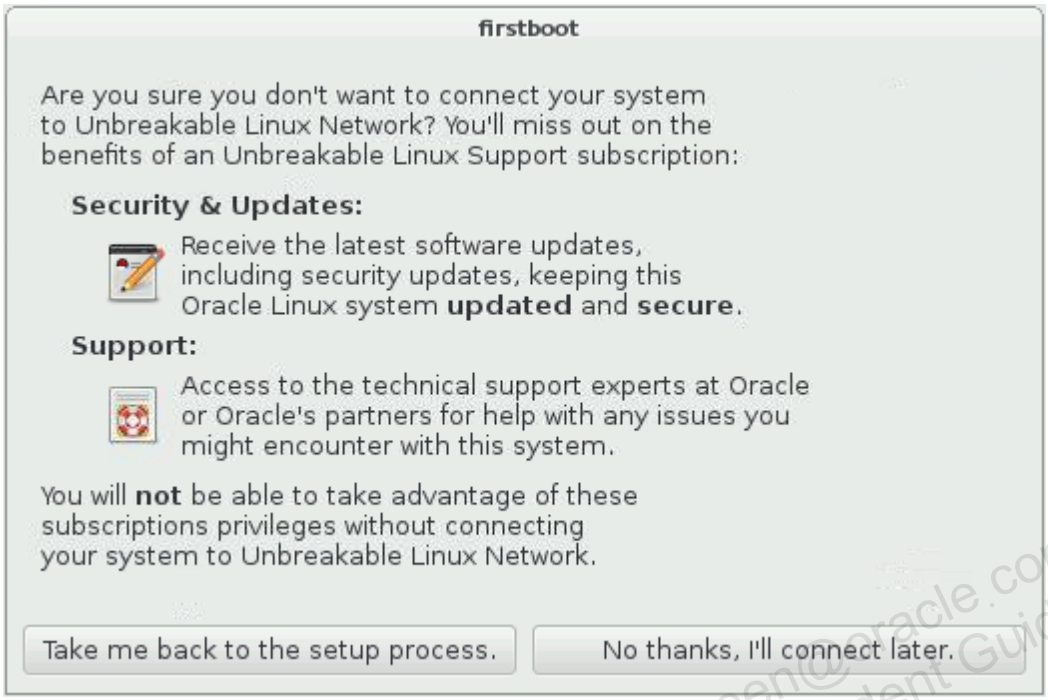
The Set Up Software Updates window appears.



d. Click the **No, I prefer to register at a later time** option button.

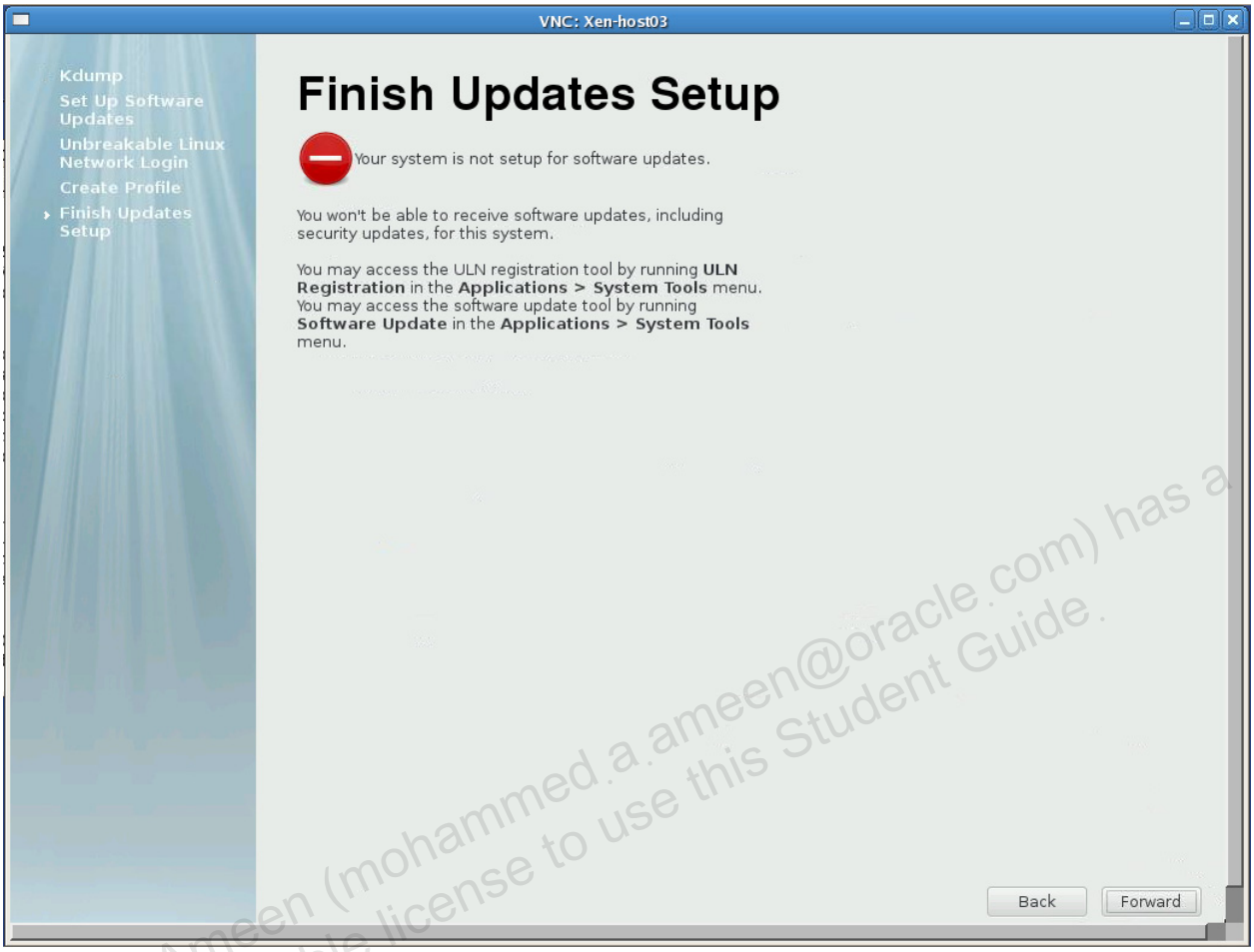
e. Scroll down if necessary and click **Forward**.

The following window appears.



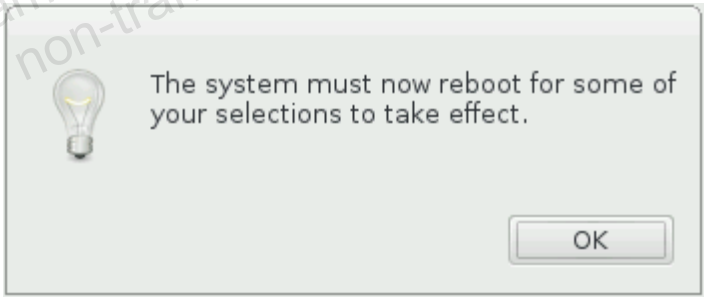
f. Click **No thanks, I'll connect later.**

The Finish Updates Setup window appears.



g. Click **Forward**.

Because you disabled Kdump, a reboot is necessary. The following window appears.



h. Click **OK**.

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

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.

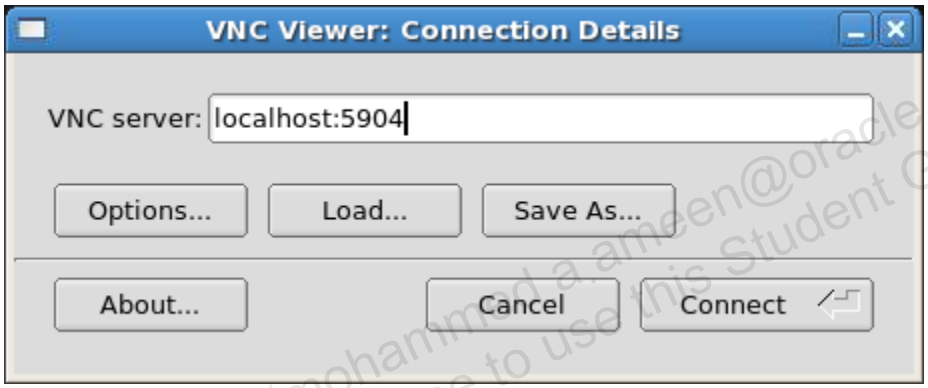
Tasks

1. Connect to the **host03** guest by using **vncviewer**.
   a. From **dom0**, run the `vncviewer&` command.

```
[dom0]# vncviewer&
```

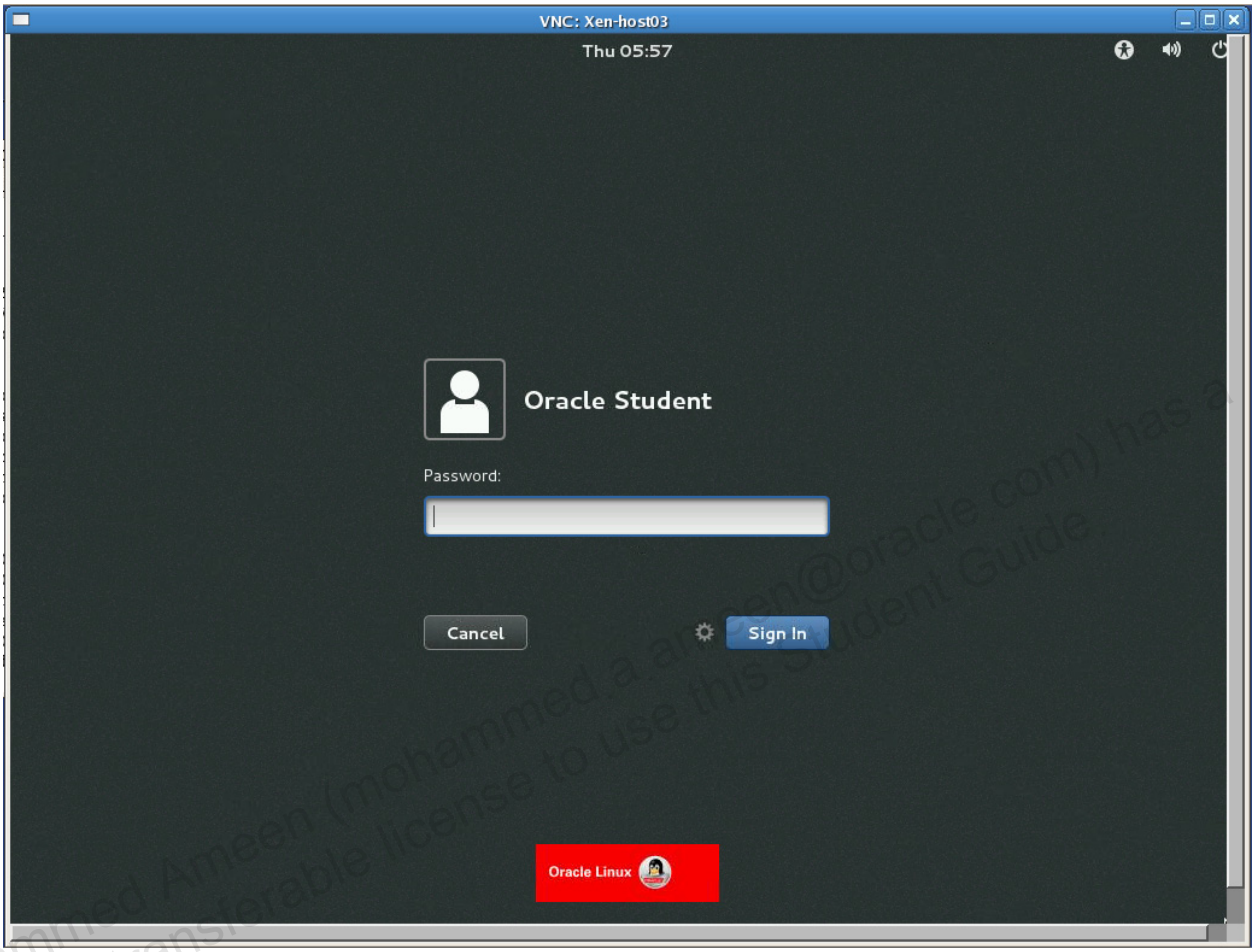
   - The VNC Viewer: Connection Details dialog box is displayed.

   b. Enter `localhost:<port_number>`, substituting the correct port number for the **host03** guest. For example, if the port number is `5904`, enter `localhost:5904` and click **Connect** (as shown in the following screenshot):



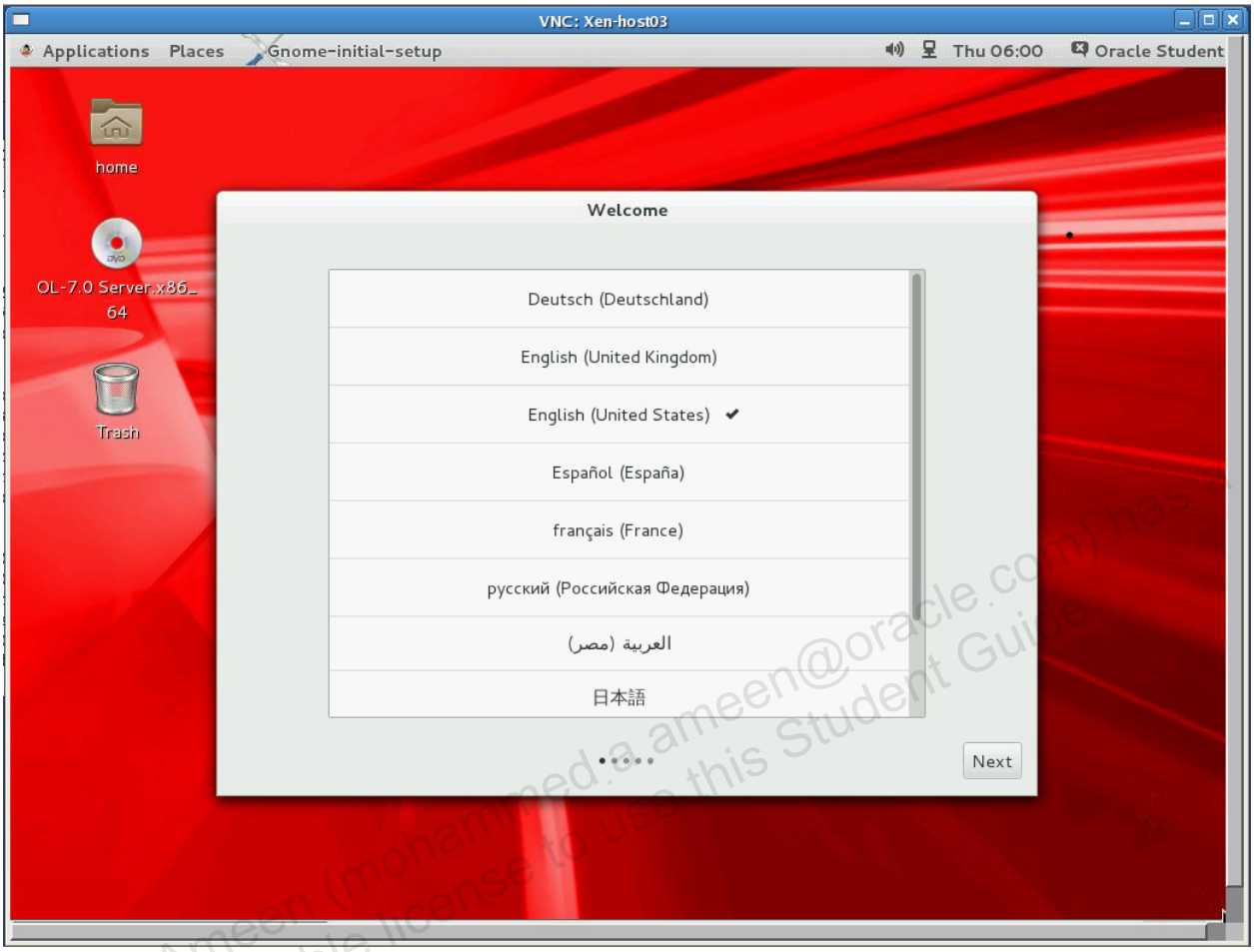
   The GNOME desktop login window appears.

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



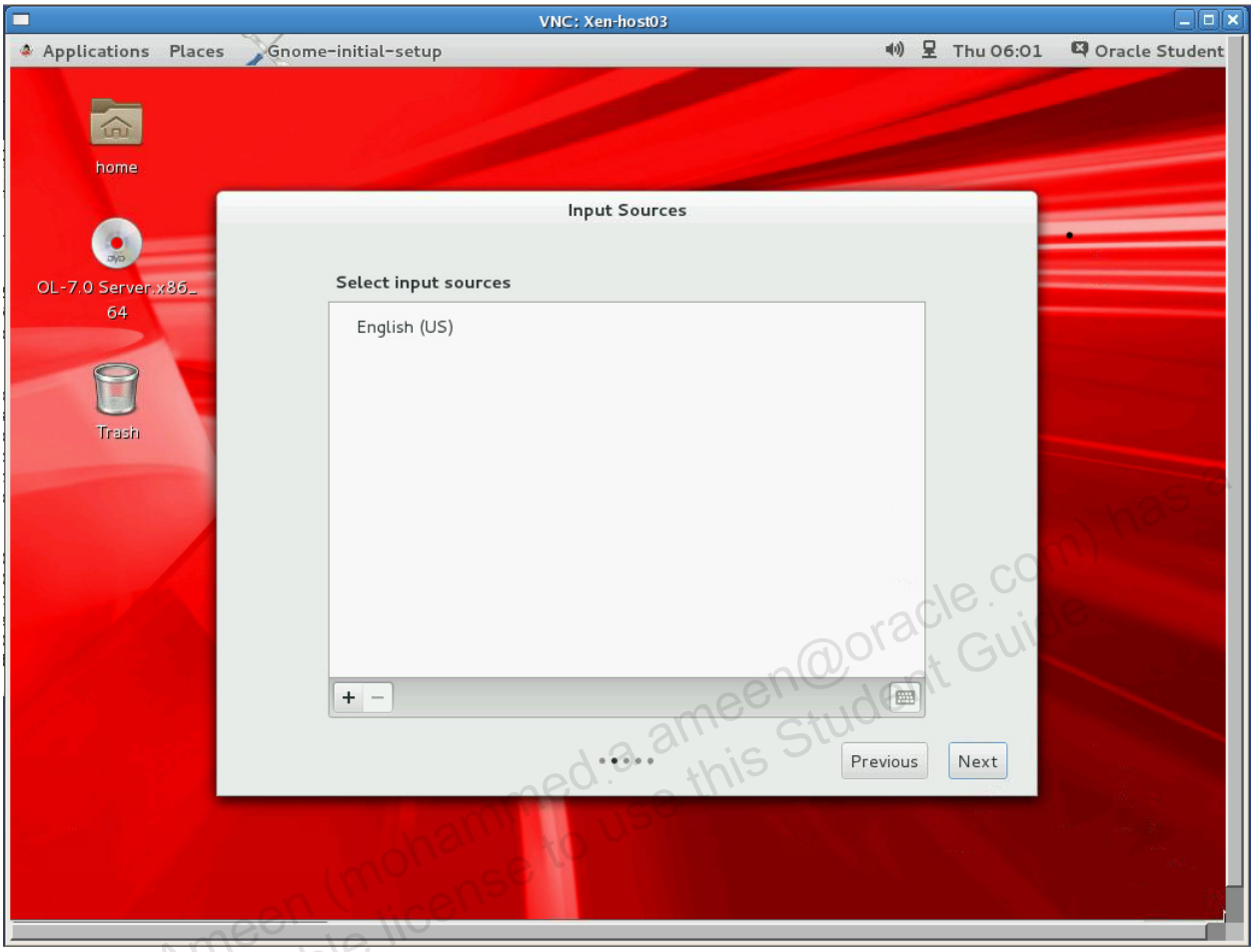
   b. Enter the password `oracle` for Oracle Student.
   c. Click **Sign In**.
      - The GNOME desktop appears.

The first time you log in, a Welcome window prompting you to select a language appears.



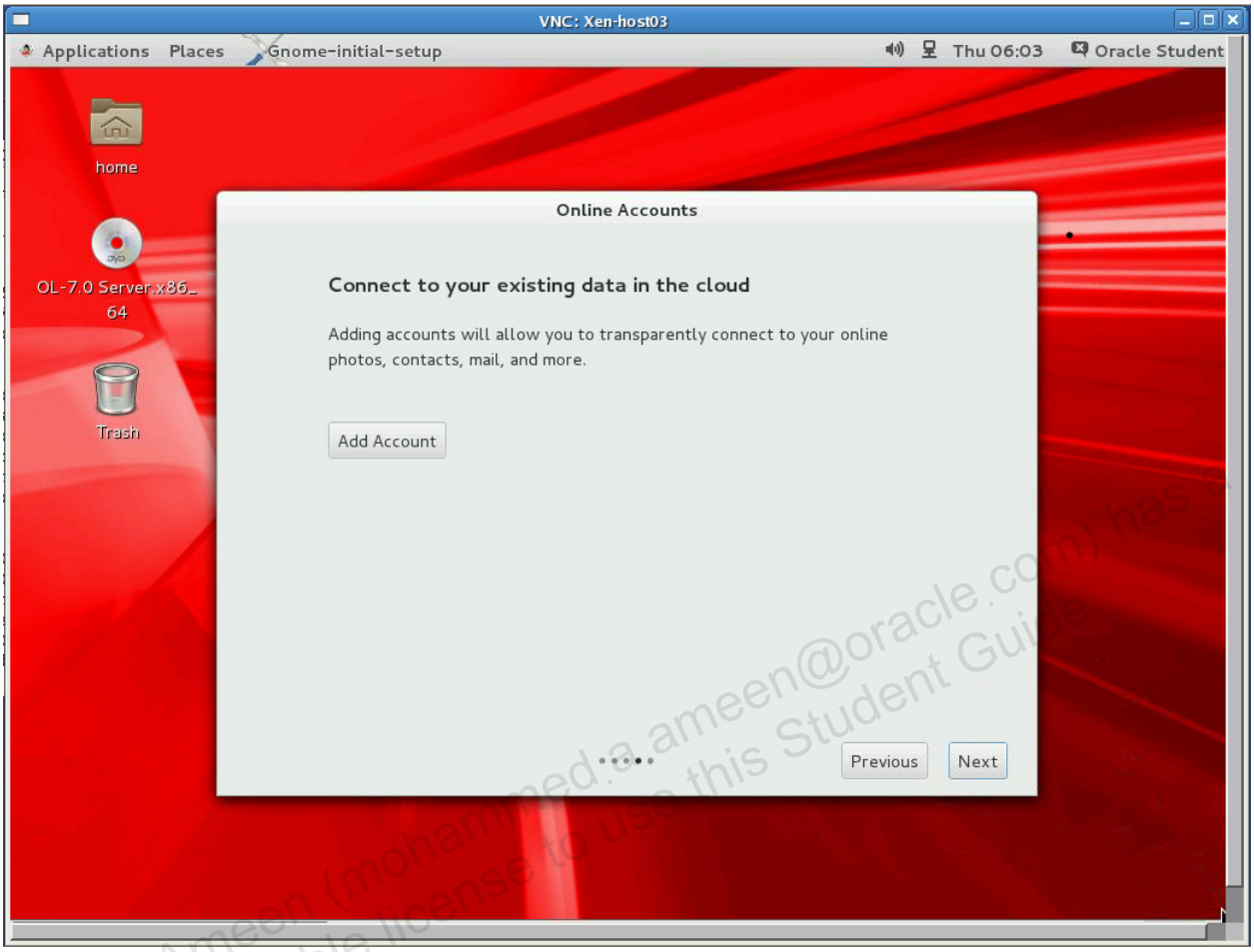
d. Click **Next**.

The following window appears:



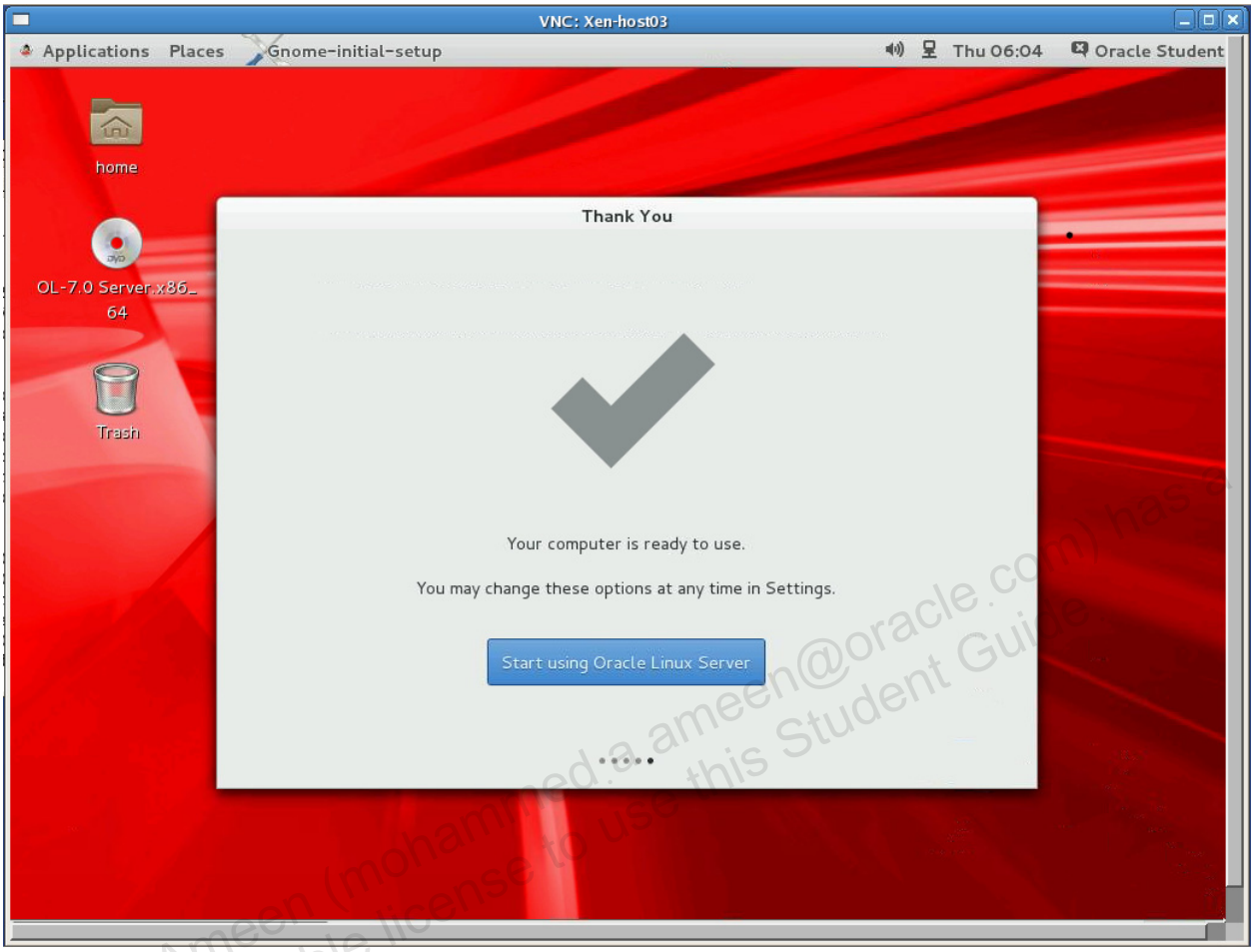
e. Click **Next**.

The following window appears:



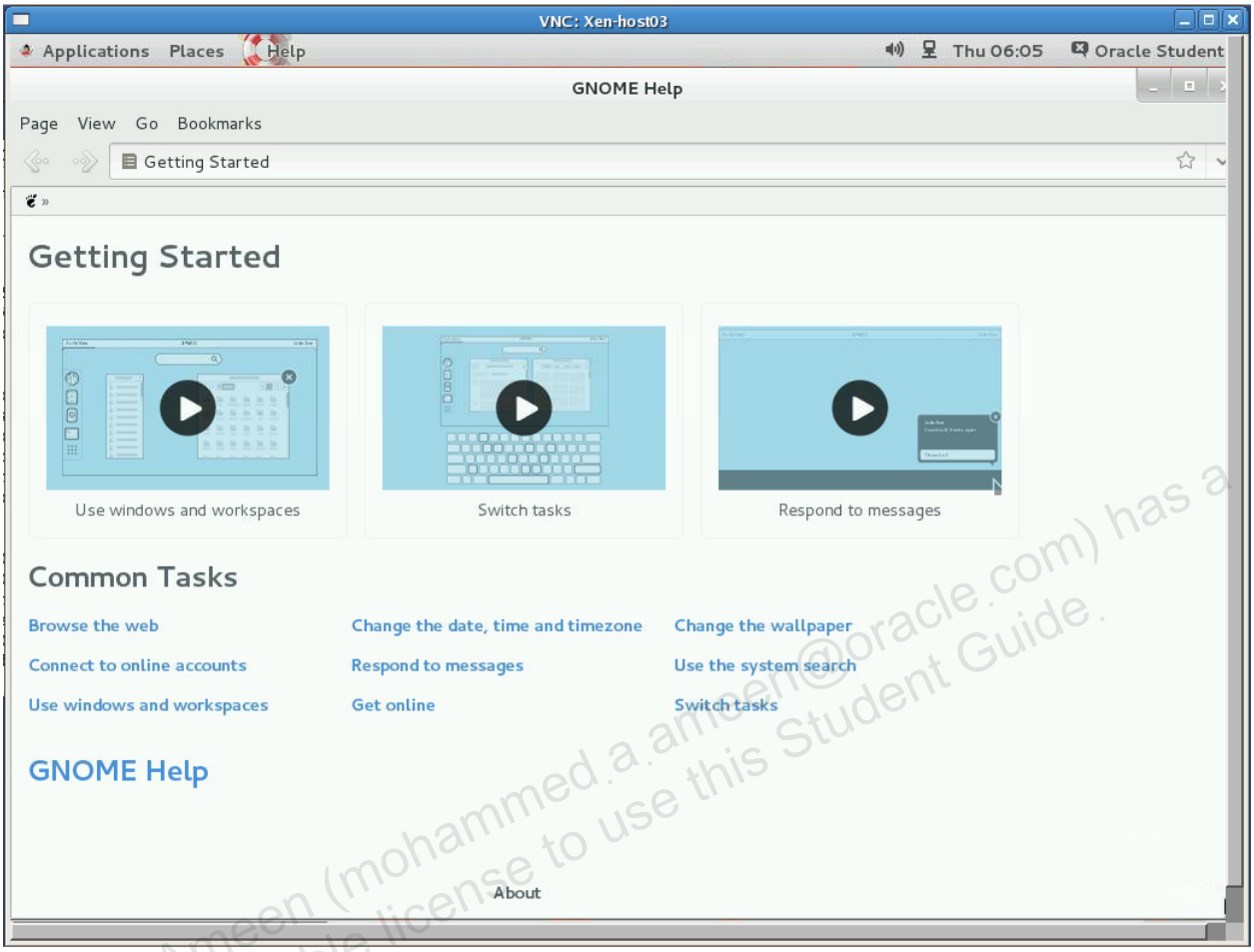
f. Click **Next**.

The following window appears:



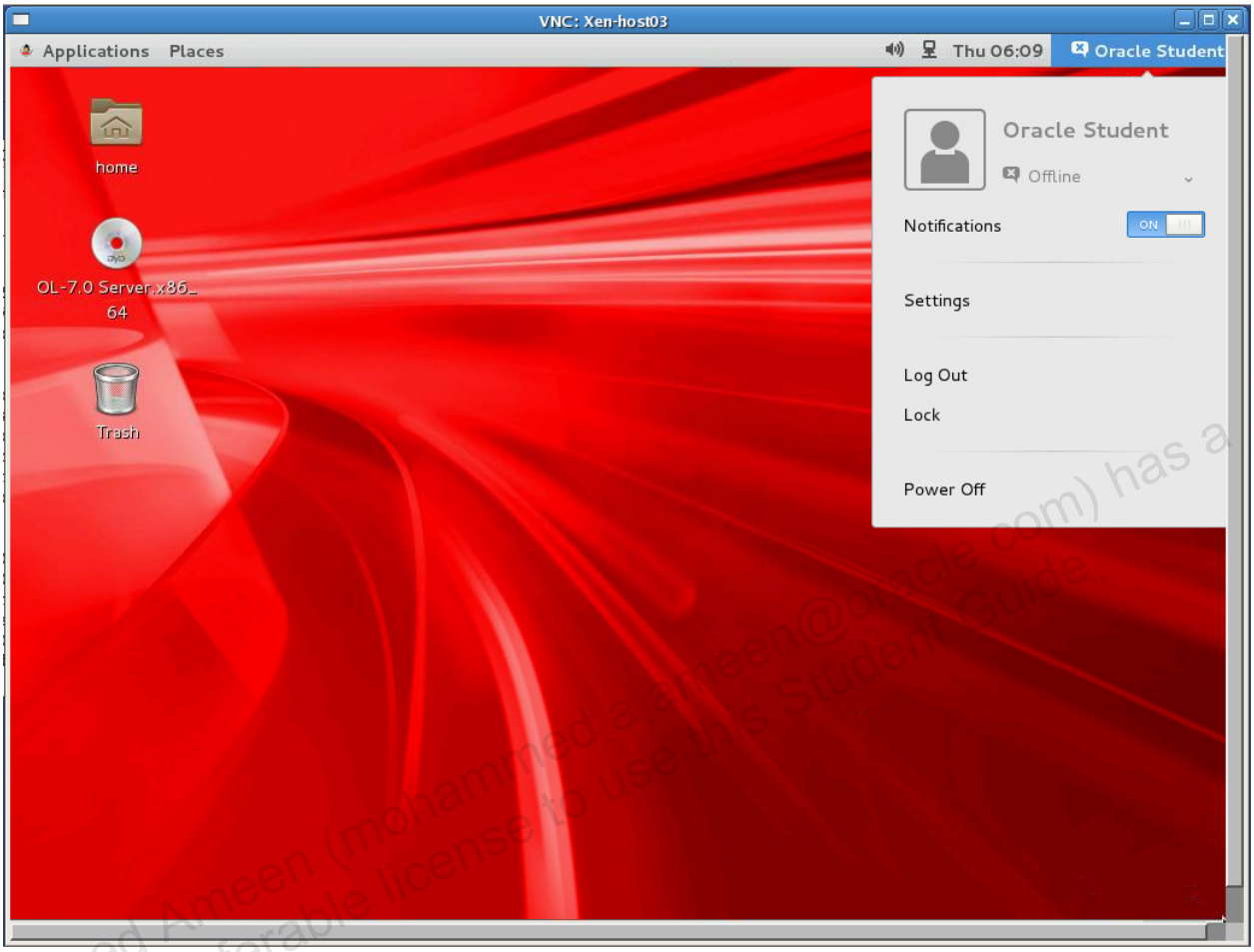
g. Click **Start using Oracle Linux Server**.

The following window appears:



h. From the GNOME help window menu bar, click **Page->Close** to close the window.

3. Shut down Oracle Linux.
   a. Click **Oracle Student** as shown:



   b. Select **Power Off** from menu.
   The following dialog box appears:

   c. Click **Power Off**.

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

Overview

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

Tasks

1. 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=...)
```

   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: Oracle Linux 7 Boot Process

Chapter 4

Practices for Lesson 4: Overview

Practices Overview

In these practices, you perform the following tasks:

- Explore the GRUB 2 boot loader.
- Edit the GRUB 2 configuration file to boot different kernels.
- Boot the rescue kernel to repair the initial RAM disk image.
- Use the GRUB 2 menu to modify kernel boot parameters.
- View `systemd` units.
- View target units.
- View the default target unit.
- Change the default target unit.
- View service units.
- Configure a service to start and stop at a specific target state.

Practice 4-1: Exploring the GRUB 2 Boot Loader

Overview

In this practice, you explore the `/boot` directory, the GRUB 2 configuration file, and kernel boot parameters. The displayed sample output might not exactly match what you see on your system. In some cases, only a 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 (Oracle Linux 7 install).
- VM **host03** is running.

If you were unable to complete the Oracle Linux 7 install on **host03** in Practice 3, substitute **host02** for **host03** in this practice and all future practices in this course.

Tasks

1. Connect to the **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 0.0.0.0:5904)
                (location 3)
```

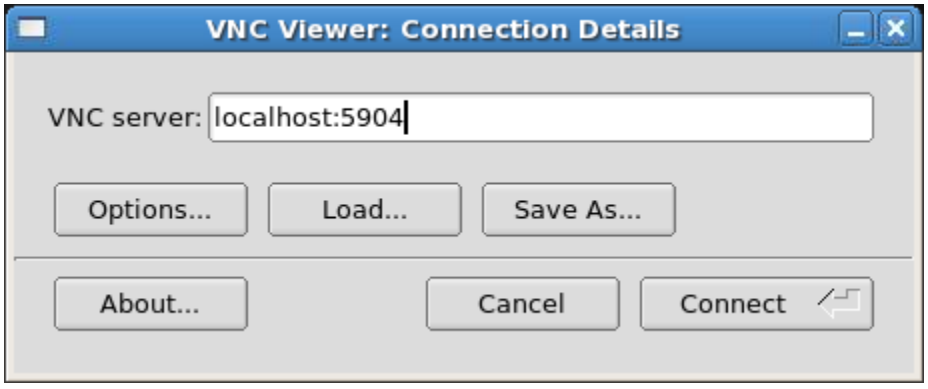
   - The sample shown indicates that the port number is **5904**. This might not be true in your case.

   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 `5904`, enter `localhost:5904`. Then click **Connect**.

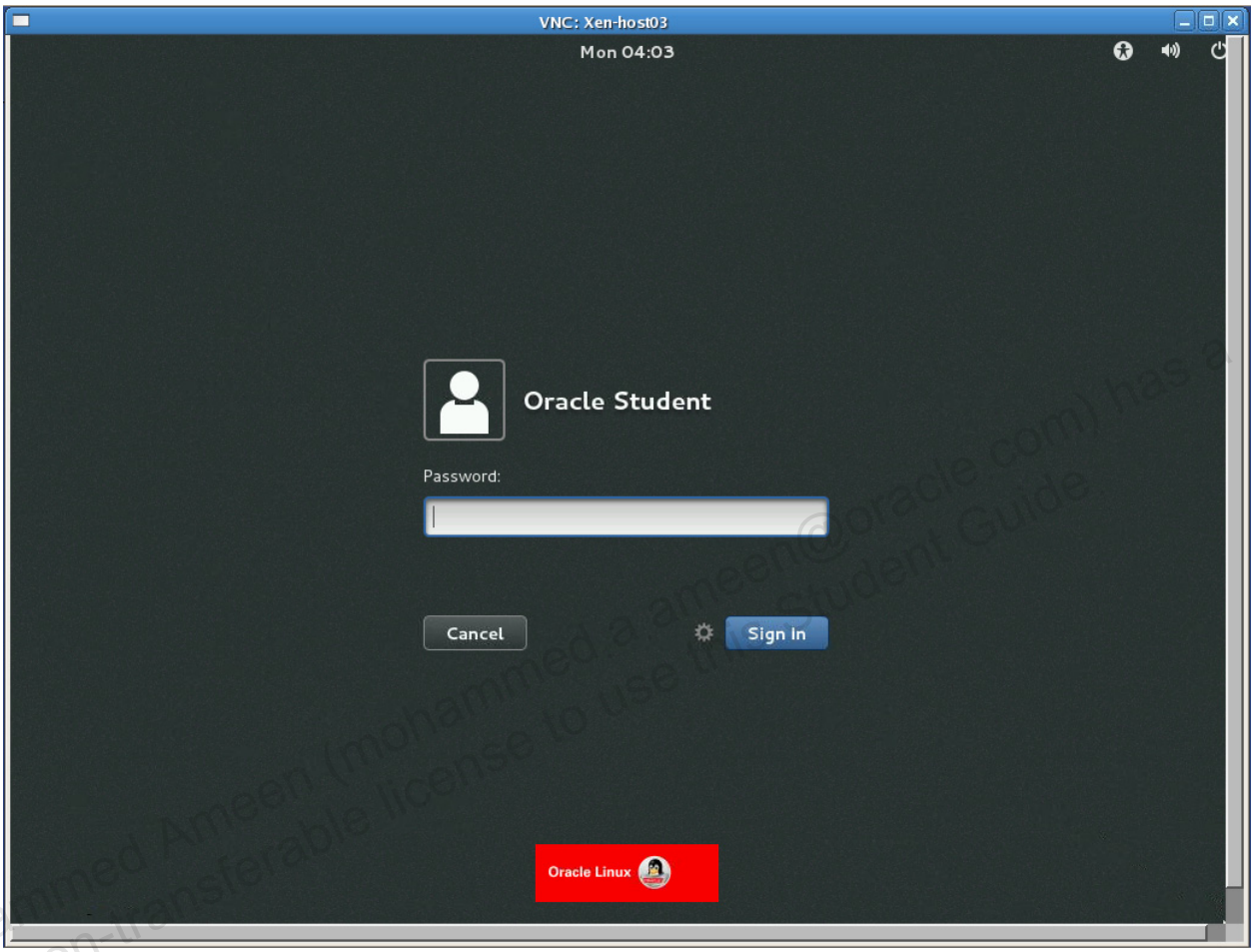


   - After you connect, the GNOME login window appears.

2. Log in from the GNOME desktop.
   - You might need to press ENTER to display the list of users.

   a. Select Oracle Student.

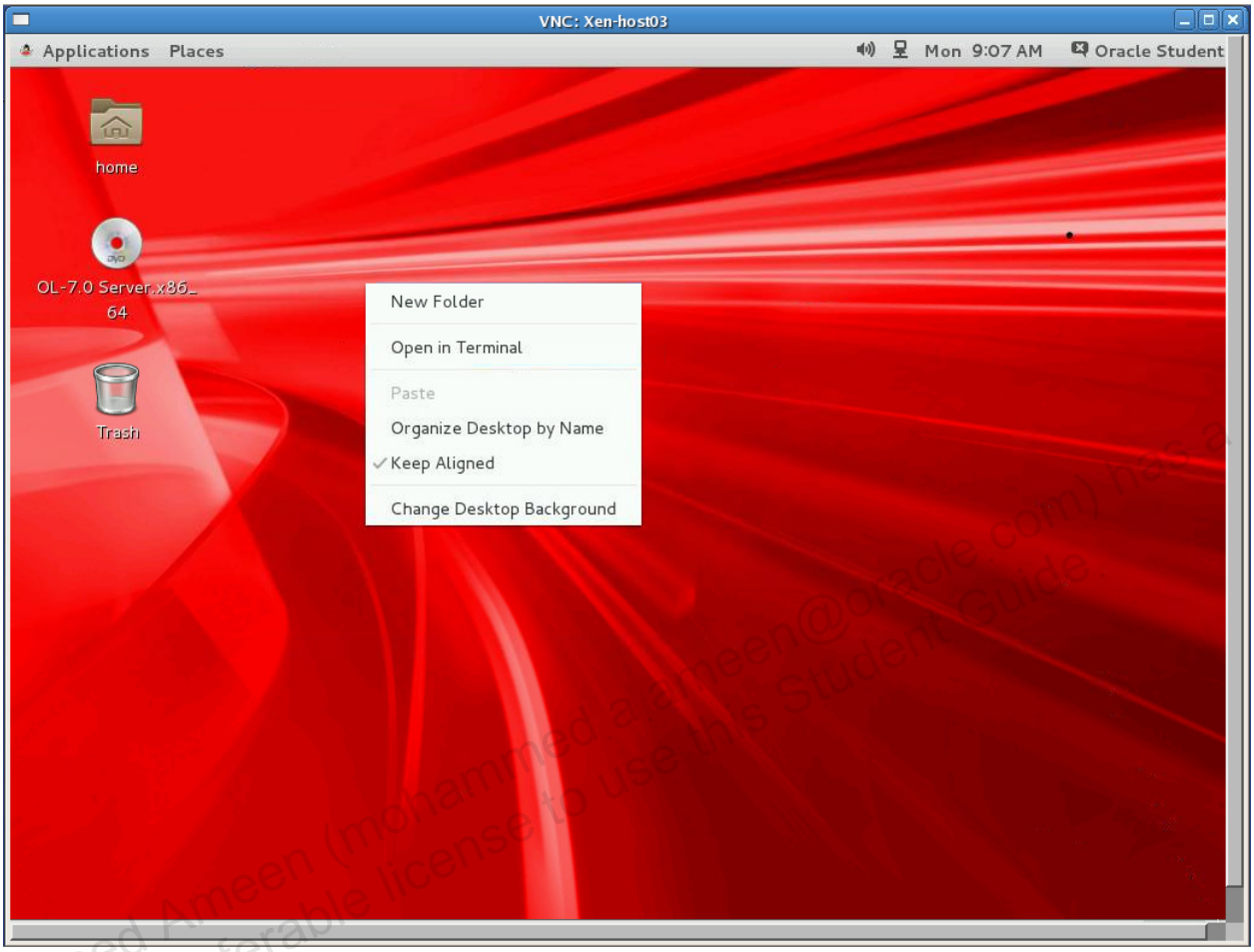
   The following window appears:



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

   c. Click **Sign 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:

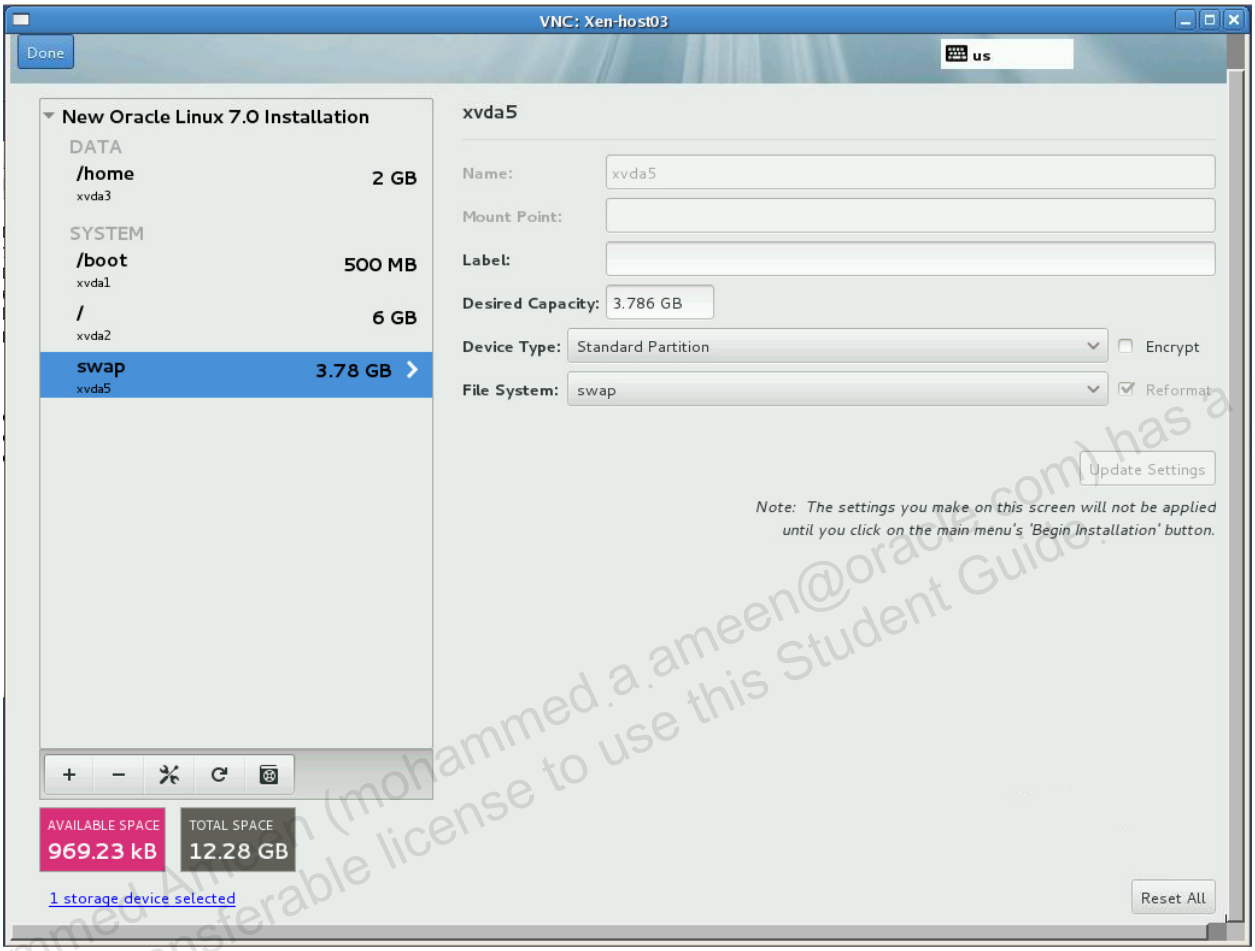


   b. Click **Open in Terminal** to display a terminal window.
   c. In the terminal window, enter the `su -` command. Then enter the `root` password `oracle` to become `root`.
      - The password is displayed in the following example but does not appear on your screen.

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

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

4. The following screen serves as a reminder of the partition layout that you created when installing Oracle Linux.
   - The partition layout that you created during the installation process is shown here:



   - The partition mounted on `/boot` contains the kernel and other files used to boot your system.

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

```
# df -h
Filesystem            Size   Used  Avail  Use%   Mounted on
/dev/xvda2            5.7G   3.1G   2.3G   58%   /
...
/dev/xvda3            1.9G    11M   1.8G    1%   /home
/dev/xvda1            477M   113M   335M   13%   /boot
...
```

   - Note that `/boot` is a separate partition mounted on `/dev/xvda1`.
   - The `/boot` partition is always located on a standard partition.
   - Placing the `/boot` partition on an LVM volume is not supported.

5. Explore the `/boot` partition.
   a. Use the `ls` command to view the `/boot/grub2` directory.

```
# ls /boot/grub2
device.map  fonts  grub.cfg  grubenv  i386-pc  locale  themes
```

   - Note the GRUB 2 configuration file, `grub.cfg`, in this directory.

   b. Use the `less` command to view the GRUB 2 configuration file `/boot/grub2/grub.cfg`.
   - Navigate through the file using the standard "`less`" navigation keys:
     - ENTER key advances one line.
     - SPACEBAR advances one screen.
     - Up and down arrows to move forward and backward.
     - Press "q" to exit the `less` command.
   - Some things to note about the `grub.cfg` file:
     - Do not edit this file. The file is generated by the `grub2-mkconfig` command using template scripts from the `/etc/grub.d` directory and settings from the `/etc/default/grub` file.
     - Each installed kernel is represented by a `menuentry` stanza, surrounded by { }.
     - Each `menuentry` stanza includes a `linux16` directive that specifies the kernel version to be booted as well as kernel boot parameters.
     - Each `menuentry` stanza includes an `initrd16` directive followed by the path to the `initramfs` image.
     - The kernel as given on the `linux16 /vmlinuz-<kernel_version>` line must match the version number of the `initramfs` image given on the `initrd16 /initramfs-<kernel_version>.img` line of each stanza.

```
# less /boot/grub2/grub.cfg
#
# DO NOT EDIT THIS FILE
#
# It is automatically generated by grub2-mkconfig using templ...
# from /etc/grub.d and settings from /etc/default/grub
#

### BEGIN /etc/grub.d/00_header ###
...
### END /etc/grub.d/00_header ###

### BEGIN /etc/grub.d/10_linux ###
menuentry 'Oracle Linux Server, with Linux 3.10.0-123.el7... {
...
        linux16 /vmlinuz-3.10.0-123.el7.x86_64 root=UUID=...
        initrd16 /initramfs-3.10.0-123.el7.x86_64.img
```

```
}
menuentry 'Oracle Linux Server, with Unbreakable Enterprise... {
...
        linux16 /vmlinuz-3.8.13-35.3.1.el7uek.x86_64 root=...
        initrd16 /initramfs-3.8.13-35.3.1.el7uek.x86_64.img
}
menuentry 'Oracle Linux Server, with Linux 0-rescue-... {
...
        linux16 /vmlinuz-0-rescue-... root=UUID=...
        initrd16 /initramfs-0-rescue-...img
}

### END /etc/grub.d/10_linux ###

### BEGIN /etc/grub.d/20_linux_xen ###

### END /etc/grub.d/20_linux_xen ###

### BEGIN /etc/grub.d/20_ppc_terminfo ###
### END /etc/grub.d/20_ppc_terminfo ###

### BEGIN /etc/grub.d/30_os-prober ###
### END /etc/grub.d/30_os-prober ###

### BEGIN /etc/grub.d/40_custom ###
...
### END /etc/grub.d/40_custom ###

### BEGIN /etc/grub.d/41_custom ###
...
### END /etc/grub.d/41_custom ###
```

c. Press "q" to exit the `less` command.

d. Use the `grep` command to list the bootable kernels in `/boot/grub2/grub.cfg`.

```
# grep '^menuentry' /boot/grub2/grub.cfg
menuentry 'Oracle Linux Server, with Linux 3.10.0-123... {
menuentry 'Oracle Linux Server, with Unbreakable Enterpri ... {
menuentry 'Oracle Linux Server, with Linux 0-rescue-... {
```

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

```
# ls /boot/vmlinuz*
vmlinuz-0-rescue-...
vmlinuz-3.10.0-123.el7.x86_64
```

```
vmlinuz-3.8.13-35.3.1.el7uek.x86_64
```

- Note that these file names match entries in the `grub.cfg` file.
- For clarification purposes, the "`el`" in the filenames is not "`e`" then the number `1`, it is "`e`" then lowercase letter `L`. The "`el`" stands for Enterprise Linux.

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

```
# ls /boot/initramfs*
initramfs-0-rescue-...img
initramfs-3.10.0-123.el7.x86_64.img
initramfs-3.8.13-35.3.1.el7uek.x86_64.img
```

- Note that these file names match entries in the `grub.cfg` file.
- You might also see several `initramfs-<`*`version`*`>kdump.img` files. These are special files created by the Kdump mechanism for kernel debugging purposes. They are not used during the boot process and can be ignored.

6. View the kernel boot parameters.

a. Use the `uname -r` command to determine the loaded kernel version.

```
# uname -r
3.8.13-35.3.1.el7uek.x86_64
```

- In this example, the UEK (`el7uek`) is loaded.

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

- Append the kernel version from the previous command to "`vmlinuz-`" in the `grep` command.

```
# grep vmlinuz-3.8.13-35.3.1.el7uek.x86_64 /boot/grub2/grub.cfg
    linux16 /vmlinuz-3.8.13-35.3.1.el7uek.x86_64 root=UUID=...
ro vconsole.keymap=us vconsole.font=latarcyrheb-sun16 rhgb quiet
```

- Kernel boot parameters are appended to the "`linux16`" entry in `grub.cfg`.
- These parameters are written to the `/proc/cmdline` file and viewable after boot.
- The UUID value differs on your system and therefore is not displayed.

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

```
# cat /proc/cmdline
BOOT_IMAGE=/vmlinuz-3.8.13-35.3.1.el7uek.x86_64 root=UUID=... ro
vconsole.keymap=us vconsole.font=latarcyrheb-sun16 rhgb quiet
```

- Note that the content is the same as the kernel boot parameters in `grub.cfg`.

Practice 4-2: Booting Different Kernels

Overview

In this practice, you change the GRUB 2 configuration file to boot different kernels. You repair the initial RAM disk image to fix a boot problem.

Assumptions

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

Tasks

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

```
# uname -r
3.8.13-35.3.1.el7uek.x86_64
```

   - The current running kernel is the UEK (`el7uek`).

   b. Use the `grep` command to list the bootable kernels in `/boot/grub2/grub.cfg`.

```
# grep '^menuentry' /boot/grub2/grub.cfg
menuentry 'Oracle Linux Server, with Linux 3.10.0-123... {
menuentry 'Oracle Linux Server, with Unbreakable Linux ... {
menuentry 'Oracle Linux Server, with Linux 0-rescue-... {
```

   - Note that the UEK is listed second in `grub.cfg`.
   - You want to boot the kernel listed first, which is the RHCK.

   c. Use the `vi` editor to change the `GRUB_DEFAULT` entry in `/etc/default/grub` from `saved` to `0`.

   - Use `0` to reference the 1st kernel entry, use `1` to reference the 2nd kernel entry, etc.
   - Use the `cp` command to make a backup copy of `/etc/default/grub` before editing.

```
# cp /etc/default/grub /etc/default/grub_SAV
# vi /etc/default/grub
...
GRUB_DEFAULT=saved                                (old entry)
GRUB_DEFAULT=0                                    (new entry)
```

   d. Use the `grub2-mkconfig` command to generate `grub.cfg`.

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
Generating grub configuration file ...
...
done
```

e. Use the `systemctl reboot` command to reboot your system.

- It might take a couple minutes to complete the reboot.

```
# systemctl reboot
...
```

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

f. 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 5904, enter the following and click **Connect**:

```
localhost:5904
```

g. Select `Oracle Student` from the GNOME login window, password is `oracle`.

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

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

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

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

```
# uname -r
3.10.0-123.el7.x86_64
```

- With **`GRUB_DEFAULT=0`**, the kernel associated with the first `menuentry` in `grub.cfg` is booted.
- The first bootable kernel is the RHCK (`el7`).

2. Cause the boot process to fail.

– In this practice, you move the `initramfs` file associated with the RHCK and cause the boot process to fail.

– You then re-create the `initramfs` file by using the `dracut` utility to fix your boot problem.

– The first thing you do is increase the GRUB 2 timeout variable to 20 so that the menu displays for a longer period of time.

a. Use the `vi` editor to change the `GRUB_TIMEOUT` entry in `/etc/default/grub` from `5` to `20`.

- The GRUB kernel selection menu displays for 20 seconds instead of 5 seconds.

```
# vi /etc/default/grub
...
GRUB_TIMEOUT=5                                  (old entry)
GRUB_TIMEOUT=20                                 (new entry)
```

b. Use the `grub2-mkconfig` command to generate `grub.cfg`.

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
Generating grub configuration file ...
...
done
```

c. Use the `mv` command to move the `initramfs` image for the RHCK to the `root` user's home directory.

- You can use the `~` character to represent the `root` user's home directory.

```
# mv /boot/initramfs-3.10.0-123.el7.x86_64.img ~
```

d. Use the `systemctl reboot` command to reboot your system.

```
# systemctl reboot
...
```

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

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

1) Run the `vncviewer&` command.

```
# vncviewer&
```

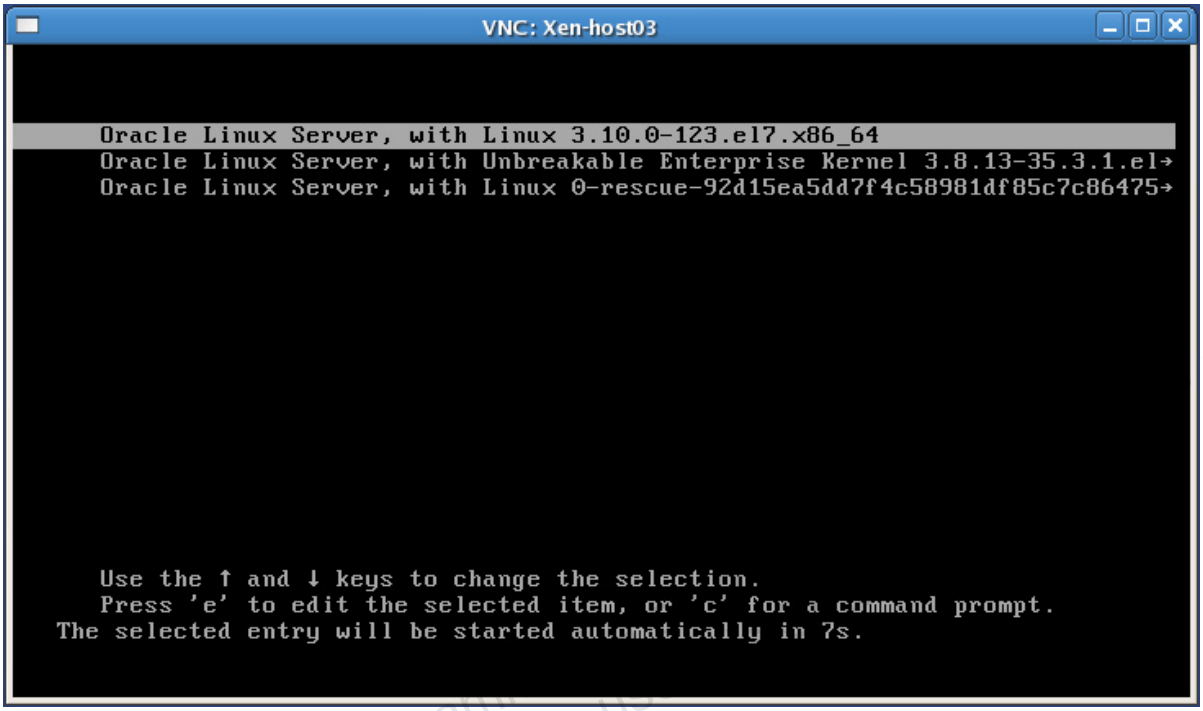
- The VNC Viewer: Connection Details dialog box is displayed.

2) Enter the command `localhost:<port_number>`, substituting the correct port number for the **host03** guest. For example, if the port number is 5904, enter the following and click **Connect**:

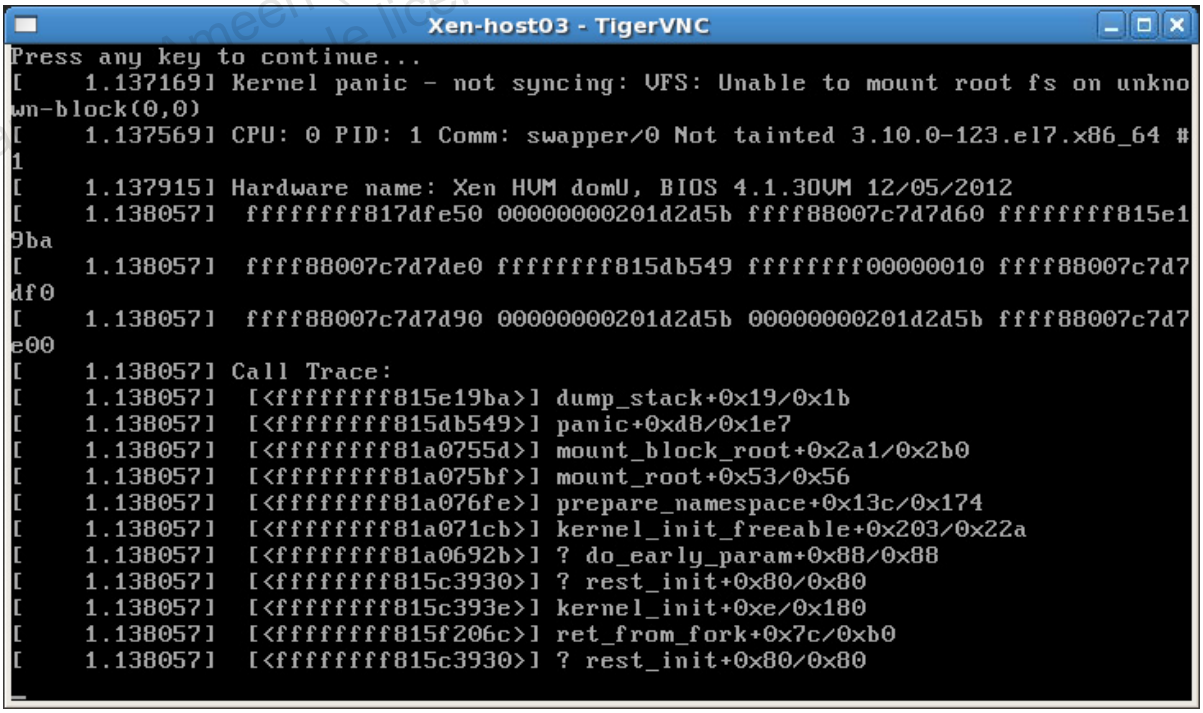
```
localhost:5904
```

The GRUB 2 menu appears with the RHCK (`3.10.0-123.el7.x86_64`) selected by default.

f. Either press **Enter** to select the RHCK or wait for the remainder of the 20 second timeout period to expire. The RHCK is automatically selected after 20 seconds.



Because the `initramfs` disk image for the RHCK is missing, the system fails to boot the RHCK. A screen similar to the following appears:



g. Click the **X** in the top-right corner to close the window.

3. Boot the rescue kernel and repair an initial RAM disk image.

a. Use the `xm destroy` command to destroy the **host03** VM.

```
# xm destroy host03
```

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 to create the **host03** VM.

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

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

1) Run the `vncviewer&` command.

```
# vncviewer&
```

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

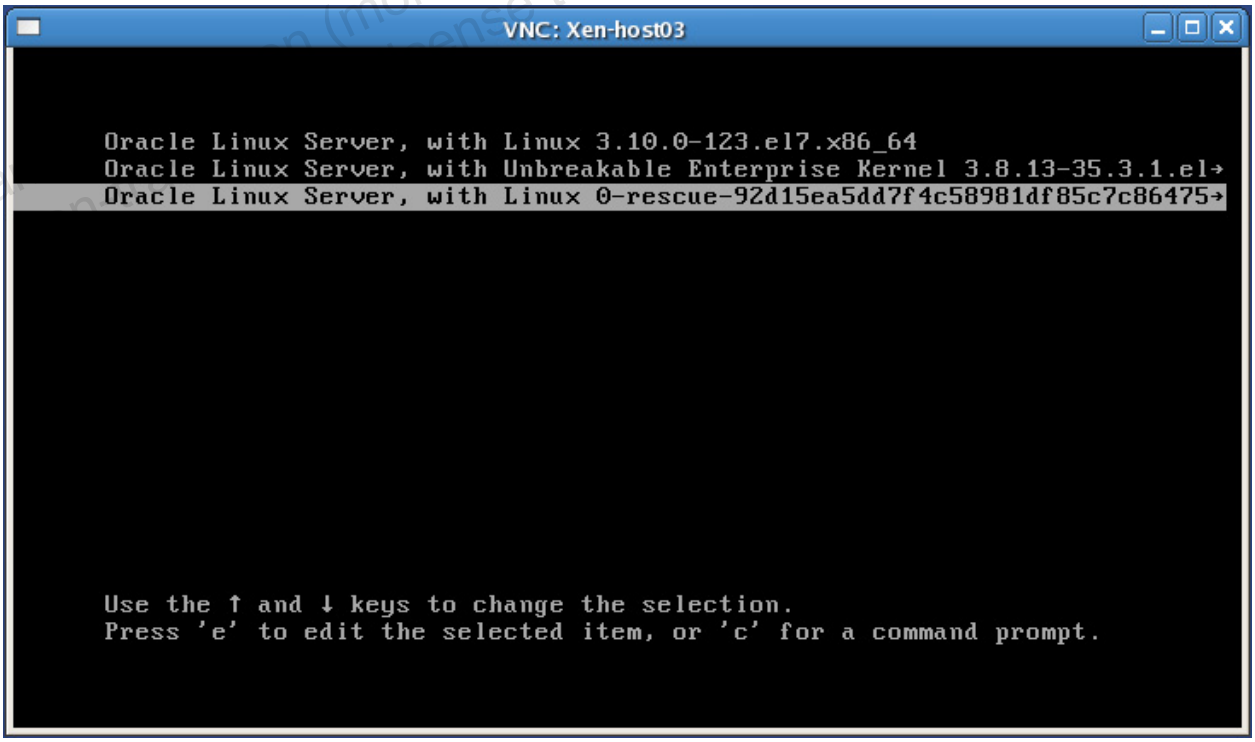
2) Enter the command `localhost:<port_number>`, substituting the correct port number for the **host03** guest. For example, if the port number is 5904, enter the following and click **Connect**:

```
localhost:5904
```

The GRUB 2 menu appears with the RHCK (`3.10.0-123.el7.x86_64`) selected by default.

e. Use the down arrow to highlight the rescue kernel as shown. Press **Enter** to select the rescue kernel.

- It might take a couple minutes to complete the boot process.



f. Select `Oracle Student` from the GNOME login window, password is `oracle`.

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

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

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

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

```
# uname -r
3.10.0-123.el7.x86_64
```

- Note the rescue kernel boots the RHCK with a known good `initramfs`.

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

```
# ls /boot/initramfs*
initramfs-0-rescue-...img
initramfs-3.8.13-35.3.1.el7uek.x86_64.img
```

- Note that the `initramfs-3.10.0-123.el7.x86_64.img` file for the RHCK is missing.

k. Use the `cd` command to change to the `/boot` directory.

- The `/boot` directory is where you want to create the `initramfs` file.

```
# cd /boot
```

l. Use the `dracut` utility to create the `initramfs-3.10.0-123.el7.x86_64.img` file.

- This command takes a few seconds to complete.
- For clarification purposes, the "`el`" in the filename is not "`e`" then the number `1`, it is "`e`" then lowercase letter `L`. The "`el`" stands for Enterprise Linux.

```
# dracut initramfs-3.10.0-123.el7.x86_64.img
```

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

```
# ls /boot/initramfs*
initramfs-0-rescue-...img
initramfs-3.10.0-123.el7.x86_64.img
initramfs-3.8.13-35.3.1.el7uek.x86_64.img
```

- Note that the `initramfs-3.10.0-123.el7.x86_64.img` file now exists.

n. Use the `lsinitrd` command to view the contents of the image created by `dracut`.

- You can pipe the output of the command to `less` to view one screen at a time.

```
# lsinitrd initramfs-3.10.0-123.el7.x86_64.img | less
...
```

o. Use the `systemctl reboot` command to reboot your system.

- It might take a couple minutes to complete the reboot.

```
# systemctl reboot
...
```

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

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

1) Run the `vncviewer&` command.

```
# vncviewer&
```

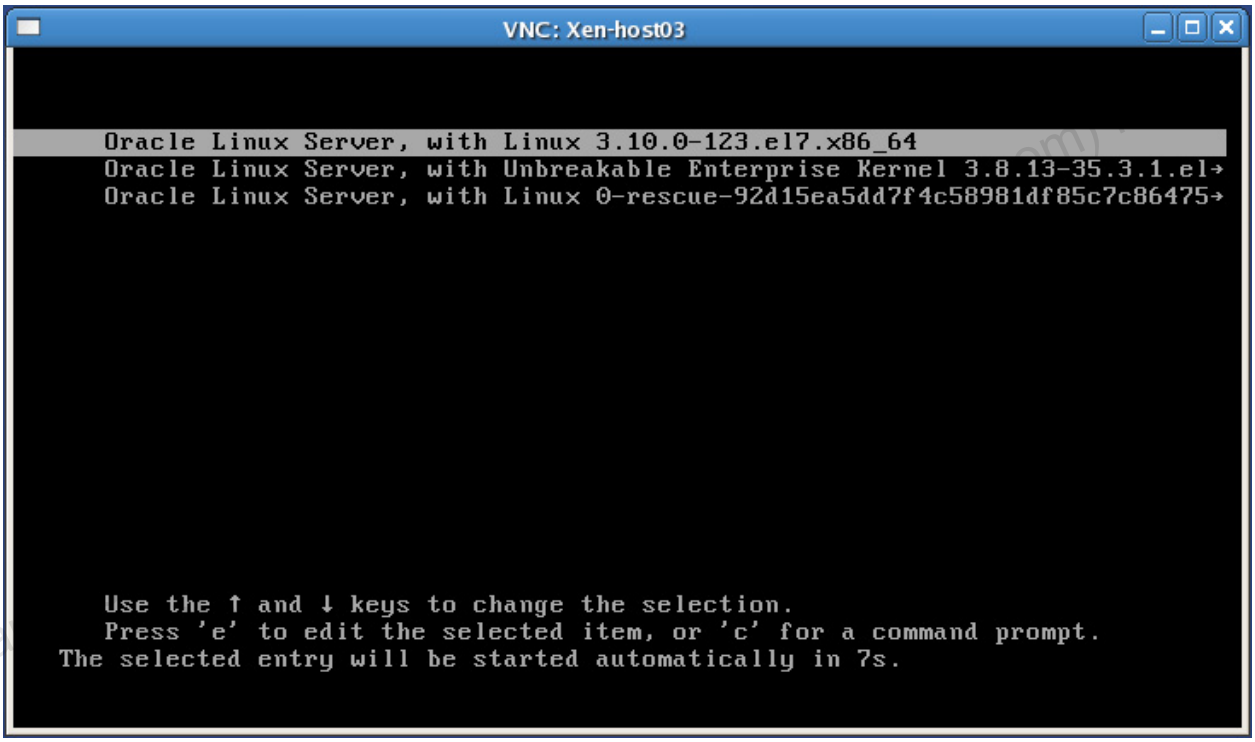
- The VNC Viewer: Connection Details dialog box is displayed.

2) Enter the command `localhost:<port_number>`, substituting the correct port number for the **host03** guest. For example, if the port number is 5904, enter the following and click **Connect**:

```
localhost:5904
```

The GRUB 2 menu appears with the RHCK (`3.10.0-123.el7.x86_64`) selected by default.

q. Either press **Enter** to select the RHCK or wait for the remainder of the 20 second timeout period to expire. The RHCK is automatically selected after 20 seconds.



r. Select `Oracle Student` from the GNOME login window, password is `oracle`.

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

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

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

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

```
# uname -r
3.10.0-123.el7.x86_64
```

- Note the RHCK boots successfully.

4. Boot the UEK.
   a. Use the `cp` command to restore `grub.cfg` from the backup copy, `grub_SAV`.

```
# cp /etc/default/grub_SAV /etc/default/grub
cp: overwrite '/etc/default/grub'? y
```

   b. Use the `grub2-mkconfig` command to generate `grub.cfg`.

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
Generating grub configuration file ...
...
done
```

   c. Use the `systemctl reboot` command to reboot your system.
      - It might take a couple minutes to complete the reboot.

```
# systemctl reboot
...
```

      - After you reboot your system, your VNC session closes.
   d. 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:5904`).
   e. Select `Oracle Student` from the GNOME login window. The password is `oracle`.
   f. Right-click the GNOME desktop and open a terminal window.
   g. In the terminal window, become the `root` user by entering the `su -` command followed by the `root` password `oracle`.

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

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

```
# uname -r
3.8.13-35.3.1.el7uek.x86_64
```

      - Notice the UEK (`el7uek`) is now loaded.

Practice 4-3: Using the GRUB 2 Menu

Overview

In this practice, you access the GRUB 2 menu and modify the kernel boot parameter to boot into rescue (single-user) mode. You also change into rescue mode from the command line.

Assumptions

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

Tasks

1. Display the default system-state target.
   a. Use the `systemctl get-default` command to view the default system-state target.

```
# systemctl get-default
graphical.target
```

   - The `graphical.target` unit corresponds to run level 5 on a SysV init system.

   b. Use the `runlevel` command to display the current run level.

```
# runlevel
N 5
```

   - The `runlevel` command still exists in Oracle Linux 7 but is included only for compatibility reasons.

2. Increase the time for which the boot menu is displayed.
   a. Use the `vi` editor to change the `GRUB_TIMEOUT` entry in `/etc/default/grub` from `5` to `60`.
   - This change causes the boot menu to display for `60` seconds before booting the default `menuentry`.

```
# vi /etc/default/grub
...
GRUB_TIMEOUT=5                                    (old entry)
GRUB_TIMEOUT=60                                   (new entry)
```

   b. Use the `grub2-mkconfig` command to generate `grub.cfg`.
   - Remember that any time a change is made to `/etc/default/grub`, you need to re-generate the `grub.cfg` file.

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
Generating grub configuration file ...
...
done
```

3. Reboot your system and display the GRUB 2 menu.
   a. Reboot your VM by using the `systemctl reboot` command.

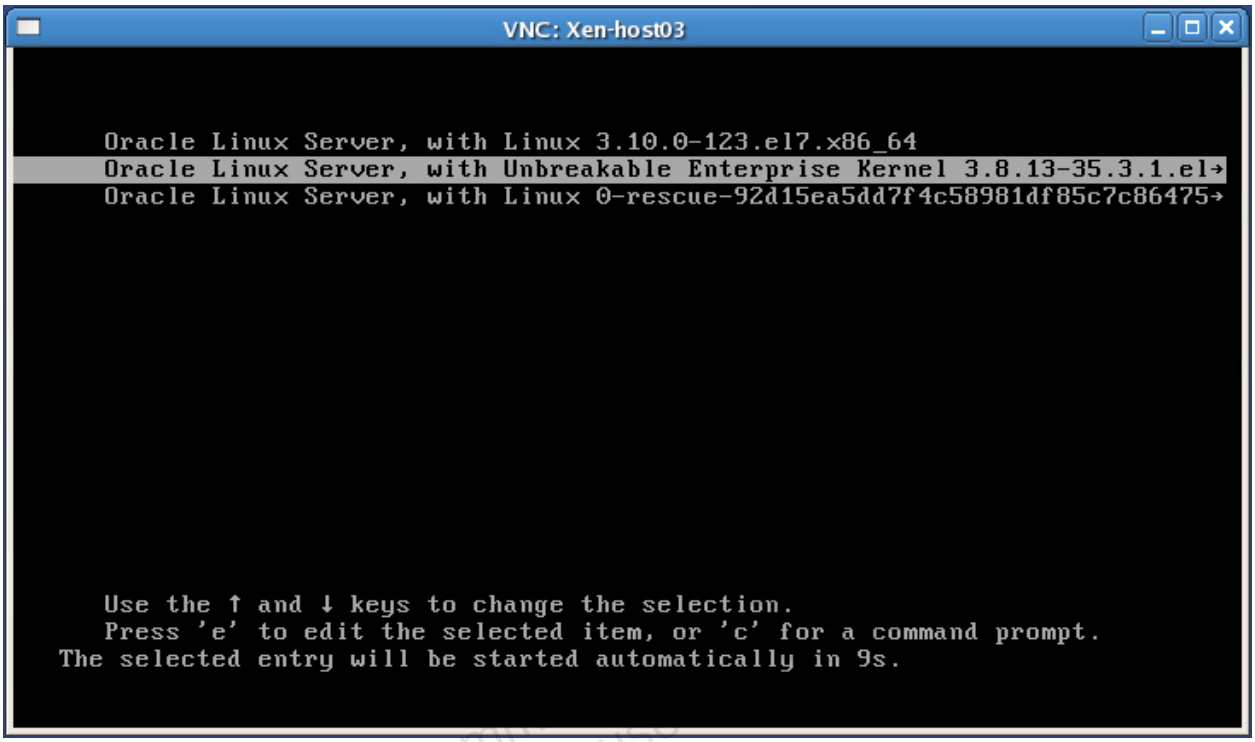
```
# systemctl reboot
```

   - 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:5904`).

After a few seconds, the GRUB 2 menu appears as shown:



- Each entry in the GRUB 2 menu is associated with a `menuentry` entry in the `grub.cfg` file.

4. Edit a bootable kernel entry to boot into rescue mode.

– Rescue mode is the same as single-user mode.

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

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

- The code associated with the UEK from `grub.cfg` is displayed as shown:

VNC: Xen-host03

```
setparams 'Oracle Linux Server, with Unbreakable Enterprise Kernel 3.8.13-35.3\
.1.el7uek.x86_64'

        load_video
        set gfxpayload=keep
        insmod gzio
        insmod part_msdos
        insmod ext2
        set root='hd0,msdos1'
        if [ x$feature_platform_search_hint = xy ]; then
          search --no-floppy --fs-uuid --set=root --hint='hd0,msdos1'  1e0af6c\
4-bf07-4f50-bfde-ac2fc3a21807
        else
          search --no-floppy --fs-uuid --set=root 1e0af6c4-bf07-4f50-bfde-ac2f\
c3a21807                                                                     ↓

      Press Ctrl-x to start, Ctrl-c for a command prompt or Escape to
      discard edits and return to the menu. Pressing Tab lists
      possible completions.
```

c. Use the arrow keys to position your cursor at the end of the `linux16 /vmlinuz-3.8.13-35.3.1.el7uek.x86_64` line as shown:

```
        linux16 /vmlinuz-3.8.13-35.3.1.el7uek.x86_64 root=UUID=161beb9e-4942-4\
32f-83ef-23855602e694 ro vconsole.keymap=us crashkernel=auto  vconsole.font=la\
tarcyrheb-sun16 rhgb quiet _
```

The following table lists the options for booting to a specific `systemd` system-state target.

- The numeric options are for legacy purposes.

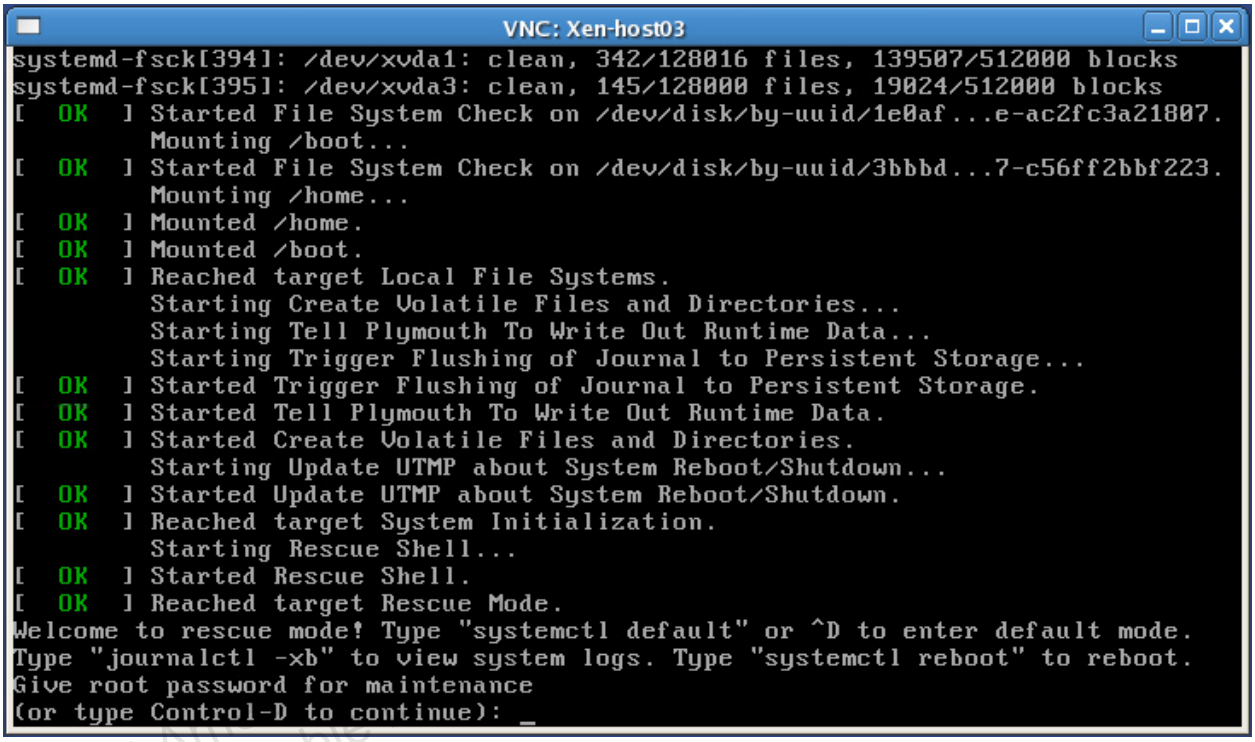
Option	**Description**
`0,1,2,3,4,5,6` or `systemd.unit=runlevelN.target`	Specifies the nearest `systemd`-equivalent system-state target to an Oracle Linux 6 run level. *N* can take an integer value between `0` and `6`.
`1,s,S,single` or `system.unit=rescue.target`	Specifies the rescue shell. The system boots to single-user mode and does not prompt for the `root` password.
`2,3,4` or `system.unit=multi-user.target`	Specifies the `systemd` target for multi-user, non-graphical login.
`5` or `systemd.unit=graphical.target`	Specifies the `systemd` target for multi-user, graphical login.
`-b`, `emergency`, or `systemd.unit=emergency.target`	Specifies emergency mode.

d. At the end of the `linux16 /vmlinuz-3.8.13-35.3.1.el7uek.x86_64` line, append a kernel boot parameter to specify the rescue shell. The following example appends the word `single`, but you could also use `1,s,S` or `system.unit=rescue.target`.

```
        linux16 /vmlinuz-3.8.13-35.3.1.el7uek.x86_64 root=UUID=161beb9e-4942-4\
32f-83ef-23855602e694 ro vconsole.keymap=us crashkernel=auto  vconsole.font=la\
tarcyrheb-sun16 rhgb quiet single_
```

e. Press Ctrl + x to accept your change and continue the boot process.

- It might take a couple minutes to complete the reboot.
- The boot process proceeds until the following “Welcome to rescue mode!” screen appears:



- View the different options available for you to type from rescue mode:
  - `systemctl default`
  - `Ctrl-D (^D)`
  - `journalctl -xb`
  - `systemctl reboot`

f. Enter the `root` password for maintenance. The `root` password is `oracle`.

```
Give root password for maintenance
(or type Control-D to continue): oracle
```

g. Use the `runlevel` command to display the current run level.

```
# runlevel
N 1
```

- Note that you are in single-user mode (run level is `1`).

h. Use the `systemctl status network` command to check if the network service is enabled.

```
# systemctl status network
Network.service - LSB: Bring up/down networking
  Loaded: loaded (/etc/rc.d/init.d/network)
```

```
   Active: inactive (dead)
```

- Note that the network service is loaded but is not started – inactive (dead) – in rescue mode.

i. Use the `systemctl get-default` command to view the default system-state target.

```
# systemctl get-default
graphical.target
```

- Note that the default system-state target remains set to `graphical.target`, even though you are in rescue (single-user) mode.

5. Bring the system from rescue mode to the default system-state target.
   a. Press Ctrl + d to exit rescue mode and continue the boot process.
   b. Select `Oracle Student` from the GNOME login window. The password is `oracle`.
   c. Right-click the GNOME desktop and open a terminal window.
   d. In the terminal window, become the `root` user by entering the `su -` command followed by the `root` password `oracle`.

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

   e. Use the `runlevel` command to display the current run level.

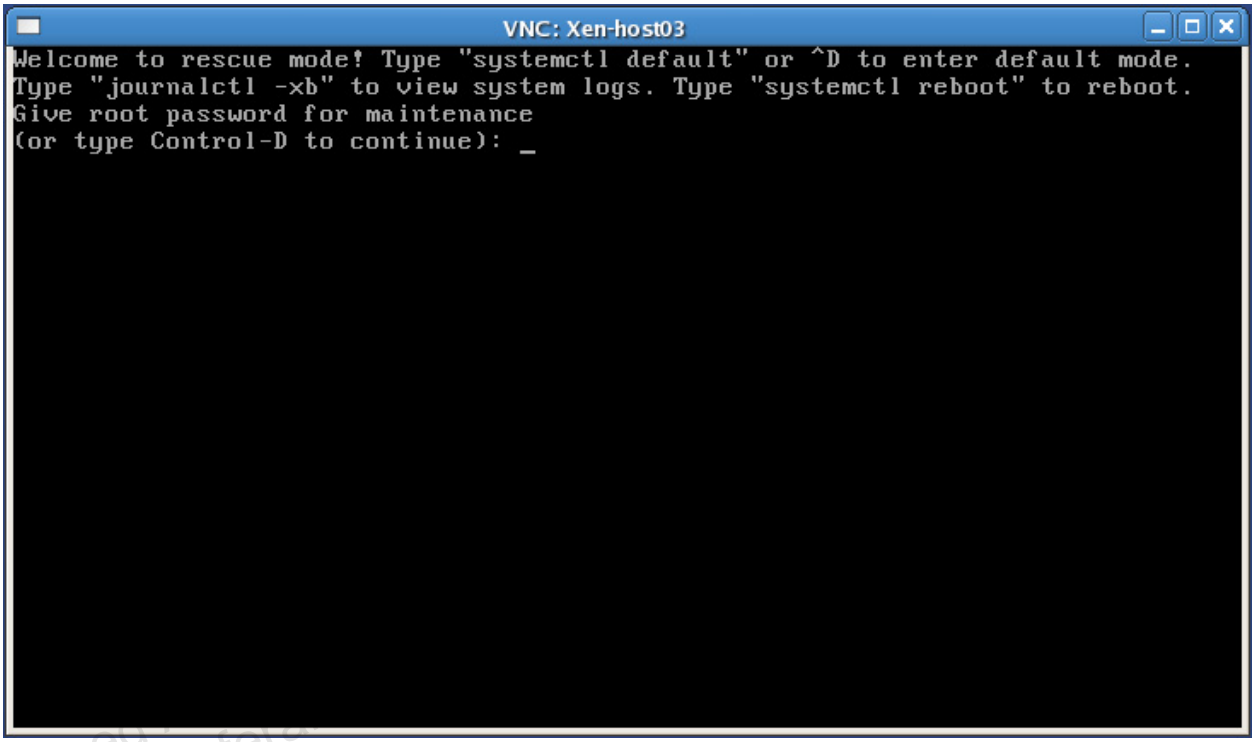
```
# runlevel
1 5
```

- Note that you are at run level `5`, which corresponds to the system `graphical.target` target unit.

6. Enter rescue mode from the shell.
   – You can enter rescue mode after you are logged in.
   a. Use the `systemctl rescue` command to enter rescue mode.
      - This command sends a message to all users currently logged on that the system is going down.

```
# systemctl rescue
```

      - The following screen appears almost immediately:



   b. Press Ctrl + d to exit rescue mode and continue the boot process.
   c. Select `Oracle Student` from the GNOME login window. The password is `oracle`.
   d. Right-click the GNOME desktop and open a terminal window.
   e. In the terminal window, become the `root` user by entering the `su -` command followed by the `root` password `oracle`.

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

Practice 4-4: Exploring `systemd` Units

Overview

In this practice, you view `systemd` units.

Assumptions

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

Tasks

1. Explore all `systemd` unit files.

   Run the `systemctl list-unit-files` command to list all installed unit files.

```
# systemctl list-unit-files
UNIT FILE                                   STATE
proc-sys-fs-binfmt_misc.automount           static
dev-hugepages.mount                         static
proc-fs-nfsd.mount                          static
...
brandbot.path                               disabled
cups.path                                   enabled
...
session-2.scope                             static
abrt-ccpp.service                           enabled
abrt-oops.service                           enabled
...
machine.slice                               static
system.slice                                static
...
avahi-daemon.socket                         enabled
cups.socket                                 enabled
...
anaconda.target                             static
basic.target                                static
...
system-readahead-done.timer                 static
...
```

   Note the different types of `systemd` units:

   - `automount`: Provide automount capabilities for on-demand mounting of file systems as well as parallelized boot-up.
   - `mount`: Mount units control mount points in the file system.
   - `scope`: Similar to service units but manage foreign processes instead of starting them as well.
   - `service`: Start and control daemons and the processes they consist of.

- `slice`: Use to group units that manage system processes, such as service units and scope units, in a hierarchical tree for resource management purposes.
- `socket`: Encapsulate local inter-process communication (IPC) or network sockets in the system, which are useful for socket-based activation.
- `target`: Use to group units or to provide well-known synchronization points during boot-up.
- `timer`: Use to trigger activation of other units based on timers.
- `device`: Expose kernel devices in `systemd` and can also be used to implement device-based activation.
- `snapshot`: Can be used to temporarily save the state of the set of `systemd` units, which can later be restored by activating the saved snapshot unit.
- `swap`: Encapsulate memory swap partitions or swap files.

2. Explore `automount` units.
   a. Use the `systemctl list-units --type automount` command to list the currently active automount units on your system.

```
# systemctl list-units --type automount
UNIT                LOAD   ACTIVE SUB     DESCRIPTION
proc-sys-...        loaded active running Arbitrary Executab...

LOAD   = Reflects whether the unit definition was properly ...
ACTIVE = The high-level unit activation state, i.e. general...
SUB    = The low-level unit activation state, values depend ...

1 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

   b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type automount --all
...
1 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

3. Explore `mount` units.
   a. Use the `systemctl list-units --type mount` command to list the currently active mount units on your system.

```
# systemctl list-units --type mount
UNIT                LOAD   ACTIVE SUB       DESCRIPTION
-.mount             loaded active mounted   /
boot.mount          loaded active mounted   /boot
dev-hugepages.mount loaded active mounted   Huge Pages File ...
...

12 loaded units listed. Pass --all to see loaded but inactiv...
```

```
To show all installed unit files use 'systemctl list-unit-files'
```

b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type mount --all
...
13 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

4. Explore `scope` units.

a. Use the `systemctl list-units --type scope` command to list the currently active scope units on your system.

```
# systemctl list-units --type scope
UNIT                LOAD   ACTIVE SUB     DESCRIPTION
session-4.scope     loaded active running Session 4 of user ...
...

1 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type scope --all
...
1 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

5. Explore `slice` units.

a. Use the `systemctl list-units --type slice` command to list the currently active slice units on your system.

```
# systemctl list-units --type slice
UNIT                LOAD   ACTIVE SUB     DESCRIPTION
-.slice             loaded active active  Root Slice
system-getty.slice  loaded active active  system-getty.slice
...

7 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type slice --all
...
9 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

6. Explore `socket` units.

a. Use the `systemctl list-units --type socket` command to list the currently active socket units on your system.

```
# systemctl list-units --type socket
UNIT                LOAD   ACTIVE SUB       DESCRIPTION
avahi-daemon.socket loaded active listening Avahi mDNS...
cups.socket         loaded active running   CUPS Printing ...
...

13 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type socket --all
...
14 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

7. Explore `timer` units.

a. Use the `systemctl list-units --type timer` command to list the currently active timer units on your system.

```
# systemctl list-units --type timer
UNIT                LOAD   ACTIVE SUB      DESCRIPTION
systemd-tmpfiles... loaded active waiting  Daily Cleanup ...
...

1 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type timer --all
...
2 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

8. Explore `device` units.

a. Use the `systemctl list-units --type device` command to list the currently active device units on your system.

```
# systemctl list-units --type device
UNIT                   LOAD   ACTIVE SUB     DESCRIPTION
sys-device...block... loaded active plugged /sys/devices/plat...
...

27 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type device --all
...
62 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

9. Explore `snapshot` units.
   a. Use the `systemctl list-units --type snapshot` command to list the currently active snapshot units on your system.

```
# systemctl list-units --type snapshot
0 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

   b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type snapshot --all
0 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

10. Explore `swap` units.
    a. Use the `systemctl list-units --type swap` command to list the currently active swap units on your system.

```
# systemctl list-units --type swap
UNIT                LOAD   ACTIVE SUB    DESCRIPTION
dev-xvda5.swap      loaded active active  /dev/xvda5

1 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

    b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type swap --all
...
2 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

Practice 4-5: Working with `systemd` Target and Service Units

Overview

In this practice, you view target units, view the default target unit, change the default target unit, view service units, and configure a service to start and stop at a specific target state.

Assumptions

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

Tasks

1. Explore target units.
   a. Use the `systemctl list-units --type target` command to list the currently active targets on a system.
      - Sample output is shown. Output on your system might be different.

```
# systemctl list-units --type target
UNIT                LOAD   ACTIVE SUB    DESCRIPTION
basic.target        loaded active active Basic System
cryptsetup.target   loaded active active Encrypted Volumes
getty.target        loaded active active Login Prompts
graphical.target    loaded active active Graphical Interface
local-fs-pre.target loaded active active Local File Systems ...
local-fs.target     loaded active active Local File Systems
multi-user.target   loaded active active Multi-User System
network.target      loaded active active Network
nfs.target          loaded active active Network File System ...
paths.target        loaded active active Paths
remote-fs.target    loaded active active Remote File Systems
slices.target       loaded active active Slices
sockets.target      loaded active active Sockets
swap.target         loaded active active Swap
sysinit.target      loaded active active System Initialization
timer.target        loaded active active Timers

LOAD   = Reflects whether the unit definition was properly ...
ACTIVE = The high-level unit activation state, i.e. general...
SUB    = The low-level unit activation state, values depend ...

16 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type target --all
...
27 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

2. View and change the default target unit.
   a. Use the `systemctl get-default` command to view which target unit is used by default.

```
# systemctl get-default
graphical.target
```

   - The default target unit is represented by the `/etc/systemd/system/default.target` file.

   b. Use the `ls -l` command to list information about the `/etc/systemd/system/default.target` file.

```
# ls -l /etc/systemd/system/default.target
lrwxrwxrwx ... /etc/systemd/system/default.target ->
/lib/system/system/graphical.target
```

   - Note that the `default.target` file is a symbolic link to the current default target unit file, `graphical.target`.

   c. Use the `runlevel` command to view the SysV runlevel.

```
# runlevel
N 5
```

   - The `graphical.target` unit corresponds to runlevel `5` on a SysV init system.

   d. Use the `systemctl set-default` command to change the default target unit to the `multi-user.target` unit.

```
# systemctl set-default multi-user.target
rm '/etc/systemd/system/default.target'
ln -s '/usr/lib/systemd/system/multi-user.target'
'/etc/systemd/system/default.target
```

   - Note that changing the default target unit removes the existing `default.target` symbolic link and re-creates the symbolic link, which points to the new default target unit file.

   e. Use the `ls -l` command to confirm that the `default.target` file is now a symbolic link to the `multi-user.target` file.

```
# ls -l /etc/systemd/system/default.target
lrwxrwxrwx ... /etc/systemd/system/default.target ->
/lib/system/system/multi-user.target
```

   - The `systemctl set-default` command does not change the state of the system.

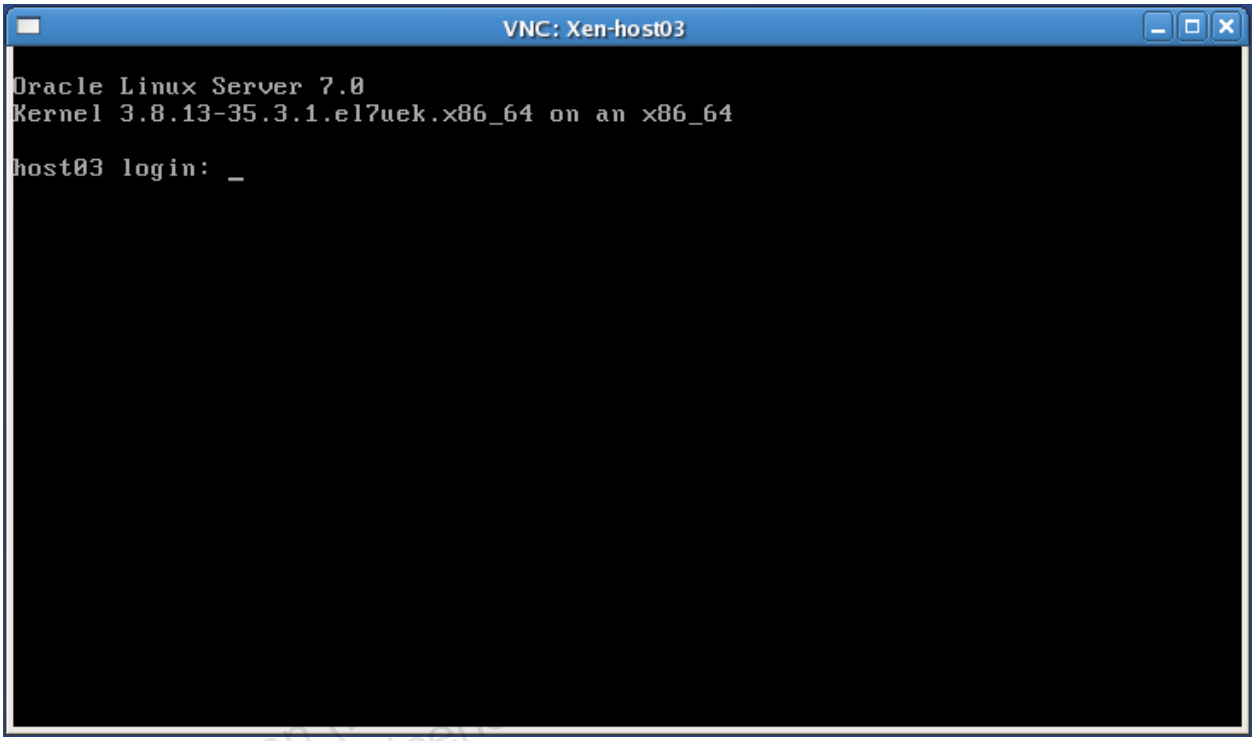
   f. Use the `runlevel` command to verify that the state of the system has not changed.

```
# runlevel
N 5
```

g. Use the `systemctl isolate` command to change the currently active system target to `multi-user.target`.

```
# systemctl isolate multi-user.target
```

The GNOME desktop closed and you now have a login prompt. The `multi-user.target` does not start the X Window services.



h. Log in as the `root` user. Password is `oracle`.

```
host03 login: root
Password: oracle
```

i. Use the `systemctl get-default` command to view the default target unit.

```
# systemctl get-default
multi-user.target
```

j. Use the `runlevel` command to view the SysV runlevel.

```
# runlevel
5 3
```

- Note that the runlevel changed from `5` to `3`.
- The `multi-user.target` unit corresponds to runlevel `3` on a SysV init system.

k. Use the `systemctl isolate` command to change the currently active system target back to `graphical.target`.

```
# systemctl isolate graphical.target
```

- Note that the GNOME desktop login window appears.

l. Select `Oracle Student` from the GNOME login window, password is `oracle`.

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

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

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

o. Use the `systemctl get-default` command to view the default target unit.

```
# systemctl get-default
multi-user.target
```

- Note that the default target unit is still `multi-user.target`.
- Running the `systemctl isolate` command does not change the default, but does change the state of the system.

p. Use the `runlevel` command to view the SysV runlevel.

```
# runlevel
3 5
```

- Notice the runlevel changed from `3` to `5`.
- If you reboot your system, it boots to SysV runlevel `3` because the default target unit is `multi-user.target`.

q. Use the `systemctl set-default` command to change the default target unit to the `graphical.target` unit.

```
# systemctl set-default graphical.target
rm '/etc/systemd/system/default.target'
ln -s '/usr/lib/systemd/system/graphical.target'
'/etc/systemd/system/default.target
```

3. Explore service units.

a. Use the `systemctl list-units --type service` command to list the currently active services on a system.

- Partial output is shown.

```
# systemctl list-units --type service
UNIT                LOAD   ACTIVE SUB     DESCRIPTION
abrt-ccpp.service   loaded active exited  Install ABRT core...
abrt-oops.service   loaded active running ABRT kernel log ...
...

LOAD   = Reflects whether the unit definition was properly ...
ACTIVE = The high-level unit activation state, i.e. general...
SUB    = The low-level unit activation state, values depend ...

64 loaded units listed. Pass --all to see loaded but inactiv...
To show all installed unit files use 'systemctl list-unit-files'
```

b. Run the same command again, but include the `--all` option to see all loaded but inactive units also.

```
# systemctl list-units --type target --all
...
134 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'
```

c. Use the `systemctl status` command to view detailed information about the `atd.service` unit.

- You can omit the `.service` extension.

```
# systemctl status atd
atd.service - Job spooling tools
Loaded: loaded (/usr/lib/systemd/system/atd.service; enabled)
Active: active (running) since <date_time; time> ago
Main PID: 518 (atd)
CGroup: /system.slice/atd.service
        |-518 /usr/sbin/atd -f
<date_time> host03.example.com systemd[1]: Started Job spool...
```

- In this example, the `atd` service is loaded and running.
- The most recent log entries are displayed at the end of the output.

d. Use the `systemctl is-active` command to check if the `iscsid` service is running (active) or not running (inactive).

```
# systemctl is-active iscsid
inactive
```

e. Use the `systemctl is-enabled` command to check if the `iscsid` service is enabled or disabled.

```
# systemctl is-enabled iscsid
disabled
```

f. Use the `systemctl status` command to view detailed information about the `iscsid` service.

```
# systemctl status iscsid
iscsid.service - Open-iSCSI
Loaded: loaded (/usr/lib/systemd/system/iscsid.service;disabled)
Active: inactive (dead)
  Docs: man:iscsid(8)
        man:iscsiadm(8)
```

- This confirms that the `iscsid` service is not running (inactive) and is disabled.
- The output of this command also includes references to man pages for `iscsid` and `iscsiadm`.

4. Start, stop, enable, and disable a service.
   a. Use the `systemctl start` command to start the `iscsid` service.

```
# systemctl start iscsid
```

   b. Use the `systemctl status` command to view detailed information about the `iscsid` service.

```
# systemctl status iscsid
iscsid.service – Open-iSCSI
Loaded: loaded (/usr/lib/systemd/system/iscsid.service;disabled)
Active: active (running) since <date_time; time> ago
...
```

      - Notice the `iscsid` service is active (running) but is still disabled.
      - This means that the `iscsid` service does not automatically start on a system reboot.

   c. Use the `systemctl enable` command to enable the `iscsid` service.

```
# systemctl enable iscsid
ln -s '/usr/lib/systemd/system/iscsid.service'
'/etc/systemd/system/multi-user.target.wants/iscsid.service'
```

      - Notice that the `systemctl enable` command enables a service by creating a symbolic link for the lowest-level system-state target at which the service starts.
      - In this example, the command creates the symbolic link `iscsid.service` for the `multi-user` target.

   d. Use the `systemctl status` command to view detailed information about the `iscsid` service.

```
# systemctl status iscsid
iscsid.service – Open-iSCSI
Loaded: loaded (/usr/lib/systemd/system/iscsid.service; enabled)
Active: active (running) since <date_time; time> ago
...
```

      - Note that the `iscsid` service is enabled.

   e. Use the `ls -l` command to view the contents of the `/etc/systemd/system/multi-user.target.wants` directory.
      - Pipe the output of the command to `less` to view one screen at a time.

```
# ls –l /etc/systemd/system/multi-user.target.wants | less
lrwxrwxrwx ... abrt-ccpp.service ->
/usr/lib/systemd/system/abrt-ccpp.service
lrwxrwxrwx ... abrtd.service ->
/usr/lib/systemd/system/abrtd.service
...
lrwxrwxrwx ... iscsid.service ->
/usr/lib/systemd/system/iscsid.service
...
```

      - Notice the symbolic link `iscsid.service` file in this directory.
      - With the service enabled, the service starts on a system reboot.

f. Use the `systemctl disable` command to disable the `iscsid` service.

```
# systemctl disable iscsid
rm '/etc/systemd/system/multi-user.target.wants/iscsid.service'
```

- Notice that the `systemctl disable` command deletes the symbolic link for the service.

g. Use the `systemctl stop` command to stop the `iscsid` service.

```
# systemctl stop iscsid
```

h. Use the `systemctl status` command to view detailed information about the `iscsid` service.

```
# systemctl status iscsid
iscsid.service - Open-iSCSI
Loaded: loaded (/usr/lib/systemd/system/iscsid.service;disabled)
Active: inactive (dead)
  Docs: man:iscsid(8)
        man:iscsiadm(8)
```

- This confirms that the `iscsid` service is not running (inactive) and is disabled.

Practices for Lesson 5: System Configuration

Chapter 5

Practices for Lesson 5: Overview

Practices Overview

In these practices, you configure date and time, enable NTP on **dom0** and **host03**, enable chrony on **host03**, modify the system configuration files, view and modify kernel settings, and view hardware device and device driver attributes.

Practice 5-1: Configuring Date and Time

Overview

In this practice, you use the `date`, `hwclock`, and `timdatectl` utilities to configure and display date and time information.

Assumptions

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

Tasks

1. Use the `date` command.

   Sample output is displayed.

   a. Run the `date` command with no arguments to display the current date and time.

   ```
   # date
   Thu Sep 18 14:01:15 MDT 2014
   ```

   b. Run the `date` command with the argument to view the weekday name.

   ```
   # date +%A
   Thursday
   ```

   c. Run the `date` command with the argument to view the full month name.

   ```
   # date +%B
   September
   ```

   d. Use the `date` command to change the date to another date, for example November 19, 2015.

   - You can substitute any date. The example changes the date to `11/19/15`.

   ```
   # date +%D -s 2015-11-19
   11/19/15
   ```

   e. Use the `date` command to change the date to the correct date.

   - Substitute today's correct date. The example changes the date to `9/18/14`.

   ```
   # date +%D -s 2014-9-18
   9/18/14
   ```

   f. Use the `date` command to change the time to the correct time.

   - Substitute the current local time. The example changes the time to `2:21 PM`.

   ```
   # date +%T%p -s 2:21:00PM
   14:21:00PM
   ```

2. Use the `hwclock` command.

   Sample output is displayed.

   a. Run the `hwclock` command with no arguments to display the hardware clock's current date and time.

   ```
   # hwclock
   Thu 18 Sep 2014 02:28:45 PM MDT  -0.988236 seconds
   ```

b. Use the `hwclock` command with the `-w` option to set the hardware clock to the current system time.

```
# hwclock -w
```

c. Run the `date` command and the `hwclock` command to confirm that the date and time settings are the same.

- The variation in seconds is due to the delay between running the two commands.

```
# date
Thu Sep 18 14:31:45 MDT 2014
# hwclock
Thu 18 Sep 2014 02:31:49 PM MDT  -0.864506 seconds
```

3. Use the `timedatectl` command.

Sample output is displayed.

a. Run the `timedatectl` command with no arguments to display the system date and time information.

```
# timedatectl
      Local time: Thu 2014-09-18 16:00:33 MDT
  Universal time: Thu 2014-09-18 22:00:33 UTC
        RTC time: Thu 2014-09-18 22:00:32
        Timezone: America/Denver (MDT, -0600)
     NTP enabled: yes
NTP synchronized: no
 RTC in local TZ: no
      DST active: yes
 Last DST change: DST began at
                  Sun 2014-03-09 01:59:59 MST
                  Sun 2014-03-09 03:00:00 MDT
 Next DST change: DST ends (the clock jumps one hour back...
                  Sun 2014-11-02 01:59:59 MDT
                  Sun 2014-11-02 01:00:00 MST
```

b. Use the `timedatectl` command with the `set-time` argument to change the date to another date, for example November 19, 2015.

- You can substitute any date. The example changes the date to `11/19/15`.
- Run the `timedatectl` afterwards with no arguments to verify that the date changed.

```
# timedatectl set-time 2015-11-19
# timedatectl
      Local time: Thu 2015-11-19 00:00:08 MDT
  Universal time: Thu 2015-11-19 07:00:08 UTC
        RTC time: Thu 2015-11-19 07:00:09
...
```

c. Use the `timedatectl` command to change the date to the correct date.

- Substitute today's correct date. The example changes the date to `9/18/14`.
- Run the `timedatectl` afterwards with no arguments to verify that the date changed.

```
# timedatectl set-time 2014-09-18
# timedatectl
      Local time: Thu 2014-09-18 00:00:03 MDT
  Universal time: Thu 2014-09-18 06:00:03 UTC
        RTC time: Thu 2014-09-18 06:00:03
...
```

d. Use the `timedatectl` command to change the time to the correct time.

- Substitute the current local time. The example changes the time to `4:13 PM`.
- Enter the hour using a 24-hour clock.
- Run the `timedatectl` afterwards with no arguments to verify that the date changed.

```
# timedatectl set-time 16:13:00
# timedatectl
      Local time: Thu 2014-09-18 16:13:05 MDT
  Universal time: Thu 2014-09-18 22:13:05 UTC
        RTC time: Thu 2014-09-18 22:13:05
...
```

Practice 5-2: Configuring NTP and Chrony

Overview

In this practice, you configure NTP on **dom0**, install the `ntp` software package on **host03**, configure NTP on **host03** to synchronize with **dom0**, and configure chrony on **host03**.

Assumptions

- This practice is performed on both **dom0** and **host03**.
- You are the `root` user and have a terminal window open on **dom0**.
- You are the `root` user and have a terminal window open on **host03**.

Tasks

1. Enable Network Time Protocol (NTP) on **dom0**.

   This task is performed on **dom0**.

   a. From **dom0**, use the `service` command to check the status of the NTP daemon, `ntpd`.

```
[dom0]# service ntpd status
ntpd is stopped
```

   - The NTP daemon is stopped.

   b. Use the `grep` command to search for "`server`" in the `/etc/ntp.conf` file.

```
[dom0]# grep server /etc/ntp.conf
# Use public servers from the pool.ntp.org project.
#server 0.rhel.pool.ntp.org
#server 1.rhel.pool.ntp.org
#server 2.rhel.pool.ntp.org
#broadcast 192.168.1.255 key 42          # broadcast server
#broadcast 224.0.1.1 key 42              # multicast server
#manycastserver 239.255.254.254          # manycast server
server 127.127.1.0        # local clock
```

   - Notice that, in addition to the public servers, a server is also configured for the local clock (`server 127.127.1.0`).
   - The local server is added because systems behind the Oracle University firewall are prevented access to a public NTP server.

   c. Use the `grep` command to search for "`driftfile`" in the `/etc/ntp.conf` file.

```
[dom0]# grep driftfile /etc/ntp.conf
driftfile /var/lib/ntp/drift
```

   - In this example, the `driftfile` is specified as `/var/lib/ntp/drift`.

   d. Use the `ls` command to check for the existence of the `/var/lib/ntp/drift` file.

```
[dom0]# ls /var/lib/ntp/drift
ls: /var/lib/ntp/drift: No such file or directory
```

   - In this example, the file does not exist.

   e. Use the `touch` command to create the `/var/lib/ntp/drift` file.

```
[dom0]# touch /var/lib/ntp/drift
```

f. Use the `service` command to start the NTP daemon, `ntpd`.

```
[dom0]# service ntpd start
Starting ntpd:                  [  OK  ]
```

g. Use the `service` command to check the status of the NTP daemon, `ntpd`.

```
[dom0]# service ntpd status
ntpd (pid ...) is running...
```

- The `ntpd` service is now running.

h. Use the `ntpq -p` command to display a list of known NTP peers.

- This command returns a remote host of `LOCAL(0)` and a refid of `.LOCL.` as shown.
- The preceding `*` indicates that your system is synchronized with the local server.
- The stratum is `10` as shown. The remaining output might vary from your output.
- Refer to the `ntpq(8)` man page for a description of the command output.
- If the command times out and returns an error, wait a few seconds and try again.

```
[dom0]# ntpq -p
     remote           refid      st t when pool reach   delay   offset  jitter
==============================================================================
*LOCAL(0)       .LOCL.          10 l   46   64   17    0.000    0.000   0.001
```

i. Use the `ntpstat` command to show network time synchronization status.

- Sample output is shown.
- If the command returns “unsynchronised”, wait awhile – between one and fifteen minutes – and try again.

```
[dom0]# ntpstat
synchronized to local net at stratum 11
   time correct to within 11 ms
   polling server every 128 s
```

2. Copy the `ntp` software package from **dom0** to **host03**.

– You need to install the `ntp` software package on **host03**.

– The `ntp` package is staged on **dom0**.

- Normally you get packages from the Public Yum Server, or from the Unbreakable Linux Network (ULN), or from another Yum repository.
- The Public Yum Server, ULN, and Yum repositories are covered in Lesson 6 – Package Management.

– Use the `sftp` command to transfer the `ntp` package from **dom0** to **host03**.

- The `sftp` command is covered in Lesson 17 – OpenSSH.

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 list the `ntp*` package.

```
[dom0]# ls ntp*
ntp-4.2.6p5-18.el7.x86_64.rpm
```

c. Use the `sftp` command to connect to **host03** as `root`.

- Answer "`yes`" when asked "Are you sure…"
- The `root` password is `oracle`.

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

d. Use the `mput` command to copy the `ntp*` package to **host03**.

```
sftp> mput ntp*
Uploading ntp-4.2.6p5-18.x86_64.rpm to /root/...
...
```

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

```
sftp> quit
```

The remaining tasks are performed on **host03**.

3. Install the `ntp` software package on **host03**.

a. Use the `cd` command to ensure you are in the `root` user's home directory.

- Use the `ls` command to list the `ntp` software package.

```
# cd
# ls ntp*
ntp-4.2.6p5-18.el7.x86_64.rpm
```

b. Use the `rpm` command to install the `ntp` package.

- The `rpm` command is covered in Lesson 6 – Package Management.

```
# rpm -Uvh ntp-4.2.6p5-18.el7.x86_64.rpm
Preparing...                ########################################## [100%]
Updating / installing...
   1:ntp-4.2.6p5-18.el7     ########################################## [100%]
```

4. Enable Network Time Protocol (NTP) on the **host03** VM.

In this task, you configure **host03**'s computer time to synchronize with **dom0**'s computer time.

a. Use the `cp` command to make a backup copy of the `/etc/ntp.conf` file.

```
# cp /etc/ntp.conf /etc/ntp.conf_SAV
```

b. Use the `vi` command to edit the `/etc/ntp.conf` file and make the following changes.

```
# vi /etc/ntp.conf
```

- Change the original line, shown as follows:
  - `#restrict 192.168.1.0 mask 255.255.255.0 nomodify notrap`
  - Remove the # character to uncomment the line. Change the IP address from `192.168.1.0` to `192.0.2.0`, shown as follows:
  - `restrict 192.0.2.0 mask 255.255.255.0 nomodify notrap`
- Insert a # character for each line beginning with "`server`" as shown:
  - `#server 0.rhel.pool.ntp.org iburst`
  - `#server 1.rhel.pool.ntp.org iburst`
  - `#server 2.rhel.pool.ntp.org iburst`
  - `#server 3.rhel.pool.ntp.org iburst`
- Add a "`server`" line to designate **dom0** (`192.0.2.1`) as the remote clock synchronization server as shown:
  - `server 192.0.2.1`
- The following represents all of the required changes needed to the `/etc/ntp.conf` file.
  - The specific changes are in **bold**.
  - Remember to remove the # sign from the beginning of the "`restrict`" line.

```
restrict 192.0.2.0 mask 255.255.255.0 nomodify notrap
#server 0.rhel.pool.ntp.org iburst
#server 1.rhel.pool.ntp.org iburst
#server 2.rhel.pool.ntp.org iburst
#server 3.rhel.pool.ntp.org iburst
server 192.0.2.1
```

c. Use the `grep` command to search for "`driftfile`" in the `/etc/ntp.conf` file.

```
# grep driftfile /etc/ntp.conf
driftfile /var/lib/ntp/drift
```

- In this example, the `driftfile` is specified as `/var/lib/ntp/drift`.

d. Use the `ls` command to check for the existence of the `/var/lib/ntp/drift` file.

```
# ls /var/lib/ntp/drift
ls: cannot access /var/lib/ntp/drift: No such file or directory
```

- In this example, the file does not exist.

e. Use the `touch` command to create the `/var/lib/ntp/drift` file.

```
# touch /var/lib/ntp/drift
```

f. Use the `systemctl` command to stop the `firewalld` service.

- The purpose of this command is to open UDP port 123 for NTP packets.
- The `firewalld` service is discussed in Lesson 18, "Security Administration."
- In Lesson 18 you learn how to create firewall rules to open up specific ports.
- For now, you are just going to stop the `firewalld` service, which essentially opens up all ports.

```
# systemctl stop firewalld
```

g. Use the `systemctl` command to start the NTP daemon, `ntpd`.

- This command might take a few seconds to complete.

```
# systemctl start ntpd
```

h. Use the `ntpq -p` command to verify that **host03** is synchronizing the system clock with **dom0** (`192.0.2.1`).

```
# ntpq -p
     remote           refid      st t when poll reach   delay   offset  jitter
==============================================================================
*192.0.2.1       .LOCL.          11 u    1   64  177    0.101 632.782 399.510
```

- The "`*192.0.2.1`" indicates **host03** is synchronized with **dom0**.

5. Disable Network Time Protocol (NTP) on the **host03** VM.

a. Use the `systemctl` command to stop the NTP daemon, `ntpd`.

```
# systemctl stop ntpd
```

b. Run the `ntpq -p` command and note that the connection to a remote NTP synchronization server is refused.

```
# ntpq -p
ntpq: read: Connection refused
```

6. Configure NTP on **host03** using Chrony.

a. Use the `cp` command to make a backup copy of the `/etc/chrony.conf` file.

```
# cp /etc/chrony.conf /etc/chrony.conf_SAV
```

b. Use the `vi` command to edit the `/etc/chrony.conf` file and make the following changes.

```
# vi /etc/chrony.conf
```

- Insert a # character for each line beginning with "`server`" as shown:
  - `#server 0.rhel.pool.ntp.org iburst`
  - `#server 1.rhel.pool.ntp.org iburst`
  - `#server 2.rhel.pool.ntp.org iburst`
  - `#server 3.rhel.pool.ntp.org iburst`
- Add a "`server`" line to designate **dom0** (`192.0.2.1`) as the remote clock synchronization server as shown:
  - `server 192.0.2.1`
- Change the original line, shown as follows:
  - `#allow 192.168/16`
- Remove the # character to uncomment the line and change the IP address from `192.168/16` to `192.0.2/24`, shown as follows:
  - `allow 192.0.2/24`
- The following represents all of the required changes needed to the `/etc/chrony.conf` file.
- The specific changes are in **bold**.
- Remember to remove the # sign from the beginning of the "`allow`" line.

```
#server 0.rhel.pool.ntp.org iburst
#server 1.rhel.pool.ntp.org iburst
#server 2.rhel.pool.ntp.org iburst
#server 3.rhel.pool.ntp.org iburst
server 192.0.2.1
allow 192.0.2/24
```

c. Use the `systemctl` command to start the chrony daemon, `chronyd`.

```
# systemctl start chronyd
```

To check if chrony is synchronized, use the `tracking`, `sources`, and `sourcestats` commands.

d. Run the `chronyc tracking` command to check chrony tracking.
   - Alternatively you could run `chronyc` to display a `chronyc>` prompt, and then run the `tracking` command from the `chronyc>` prompt.
   - Sample output is shown.
   - It might take a few seconds to synchronize.

```
# chronyc tracking
Reference ID    : 192.0.2.1 (192.0.2.1)
Stratum         : 12
Ref time (UTC)  : Fri Sep 19 19:06:51 2014
System time     : 0.000823375 seconds fast of NTP time
Last offset     : 0.001989304 seconds
RMS offset      : 0.060942811 seconds
Frequency       : 1728.043 ppm slow
Residual freq   : 1.100 ppm
Skew            : 94.293 ppm
Root delay      : 0.000207 seconds
Root dispersion : 0.016767 seconds
Update interval : 65.1 seconds
Leap status     : Normal
```

   - Some of the fields are described:
     - `Reference ID`: This is the reference ID and name (or IP address) if available, of the server to which the computer is currently synchronized.
     - `Stratum`: The stratum indicates how many hops away from a computer with an attached reference clock you are.
     - `Ref time`: This is the time (UT C) at which the last measurement from the reference source was processed.

e. Run the `chronyc sources` command to display information about the current time sources that `chronyd` is accessing.
   - The verbose option (`-v`) displays extra caption lines as a reminder of the meanings of the columns.
   - Sample output is shown.

```
# chronyc sources
```

```
210 Number of sources = 1
MS Name/IP address Stratum Poll Reach LastRx Last sample
===============================================================
^* 192.0.2.1             11   6   377    63  +1827us[+6783us]...
```

– Some of the fields are described:
  - M: The mode of the source. ^ means a server, = means a peer, and # indicates a locally connected reference clock.
  - S: The state of the sources. "*" indicates the source to which chronyd is currently synchronized. "+" indicates acceptable sources that are combined with the selected source. "-" indicates acceptable sources that are excluded by the combining algorithm. "?" indicates sources to which connectivity has been lost or whose packets do not pass all tests. "x" indicates a clock that chronyd thinks is a false ticker, that is, its time is inconsistent with a majority of other sources. "~" indicates a source whose time appears to have too much variability. The "?" condition is also shown at start-up, until at least three samples have been gathered from it.
  - Name/IP address: This shows the name or the IP address of the source, or reference ID for reference clocks.

f. Run the chronyc sourcestats command.
  – This command displays information about the drift rate and offset estimation process for each of the sources currently being examined by chronyd.
  – The verbose option (-v) displays extra caption lines as a reminder of the meanings of the columns.
  – Sample output is shown.

```
# chronyc sourcestats
210 Number of sources = 1
Name/IP Address NP NR Span Frequency Freq Skew  Offset Std Dev
==============================================================
192.0.2.1        5  4  259  -747.564  1623.869 -2873us    30ms
```

g. Run the command again but include the -v option.

```
# chronyc sourcestats -v
...
```

7. Disable Chrony on the **host03** VM.
   a. Use the `systemctl` command to stop the chrony daemon, `chronyd`.

```
# systemctl stop chronyd
```

   b. Run the `chronyc tracking` command and notice `chronyc` cannot talk to the `chronyd` daemon.

```
# chronyc tracking
506 Cannot talk to daemon
```

Practice 5-3: 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 `cd` command to change to the `/etc/sysconfig` directory.

```
# cd /etc/sysconfig
```

   b. Use `ls -l` command to display the contents of the `/etc/sysconfig` directory.

```
# ls -l
-rw-r--r-- ... root  root  ...  atd
-rw-r--r-- ... root  root  ...  authconfig
-rw-r--r-- ... root  root  ...  autofs
drwxr-xr-x ... root  root  ...  cbq
-rw-r--r-- ... root  root  ...  cgred
drwxr-xr-x ... root  root  ...  console
-rw-r--r-- ... root  root  ...  cpupower
-rw------- ... root  root  ...  crond
...
lrwxrwxrwx ... root  root  ...  grub -> /etc/default/grub
...
drwxr-xr-x ... root  root  ...  modules
-rw-r--r-- ... root  root  ...  netconsole
-rw-r--r-- ... root  root  ...  network
drwxr-xr-x ... root  root  ...  network-scripts
-rw-r--r-- ... root  root  ...  nfs
-rw-r--r-- ... root  root  ...  ntpd
...
```

   - Some of these files contain configuration settings for the respective service.
   - Some of these files contain command-line arguments for the respective service.
   - Some of these are directories and some are symbolic links.

   c. Use the `less` command to view selected files.

- Press **q** to quit the `less` command and close the file.
- Some files are given as examples but you can view files and directories of your choice.

```
# less atd
# specify additional command line arguments for atd
...
# less authconfig
IPADOMAOINJOINED=no
USEMKHOMEDIR=no
...
# less autofs
# Define default options for autofs.
...
# less crond
# Settings for the CRON daemon.
# CRONDARGS= : any extra command-line startup arguments for ...
...
# less firstboot
RUN_FIRSTBOOT=NO

# less iptables
# sample configuration for iptables service
# you can edit this manually or use system-config-firewall
...
# less kernel
# UPDATEDEFAULT specifies if new-kernel-pkg should make
# new kernels the default
UPDATEDEFAULT=yes
...
# less ntpd
# Command line options for ntpd
OPTIONS=" -g"
```

- Note that the files contain configuration settings, command-line options, etc.

2. Explore the `/usr/share/doc/initscripts*/sysconfig.txt` file.
   a. Use the `cd` command to change to the `/usr/share/doc/init*` directory.

```
# cd /usr/share/doc/initscripts*
# pwd
/usr/share/doc/initscripts-9.49.17
# ls
changes.ipv6  ipv6-6to4.howto  static-routes-ipv6  sysvinitfiles
COPYING       ipv6-tunnel.howto    sysconfig.txt
```

   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 slash (`/`) followed by `etc/sysconfig` to search for this string.
   - Press **n** (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
GATEWAY=<gateway IP>
NISDOMAIN=<nis domain name>
```

f. Press **q** to quit the "`less`" command.

Practice 5-4: 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.
      - Sample output is shown.

```
# ls /proc
1   17  29527  29749  341  497  acpi/      kcore      slabinfo
10  18  29594  29763  382  5    buddyinfo  keys       softirqs
11  19  29615  29770  39   507  bus/       key-users  stat
...
```

      - 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.  ...  root  root  ...  acpi
-r--r--r--.  ...  root  root  ...  buddyinfo
dr-xr-xr-x.  ...  root  root  ...  bus
-r--r--r--.  ...  root  root  ...  cgroups
-r--r--r--.  ...  root  root  ...  cmdline
-r--r--r--.  ...  root  root  ...  consoles
-r--r--r--.  ...  root  root  ...  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:  2050984 kB
MemFree:    919140 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.

- Sample output is shown.

```
# ls -d /proc/*[0-9]*
1    1305  16430  221  29452  29672  29737  29846  315  422  ...
10   1389  17     23   29517  29680  29741  29853  333  423  ...
11     14  18     24   29527  29684  29747  29883  336  428  ...
...
```

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

```
# ls -d /proc/*[0-9]* | wc -l
140
# ps -e | wc -l
141
```

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

```
# ps -e | less
PID  TTY          TIME   CMD
  1  ?        00:00:03   systemd
...
```

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

```
# cd /proc/1
# ls -l
dr-xr-xr-x.  ...  root  root  ...  attr
-r--------.  ...  root  root  ...  auxv
-r--r--r--.  ...  root  root  ...  cgroup
```

```
--w-------.  ...  root  root  ...  clear_refs
-r--r--r--.  ...  root  root  ...  cmdline
-rw-r--r--.  ...  root  root  ...  comm
-rw-r--r--.  ...  root  root  ...  coredump_filter
-r--r--r--.  ...  root  root  ...  cpuset
lrwxrwxrwx.  ...  root  root  ...  cwd -> /
-r--------.  ...  root  root  ...  environ
lrwxrwxrwx.  ...  root  root  ...  exe -> /usr/lib/systemd/...
dr-x------.  ...  root  root  ...  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:   systemd
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 5-5: 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
drwxr-xr-x.  ... fs
drwxr-xr-x.  ... hypervisor
drwxr-xr-x.  ... kernel
drwxr-xr-x.  ... module
drwxr-xr-x.  ... power
```

   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
```

   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_async  reserved_size  resume  state
wakeup_count
# cat state
mem  disk
```

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

Practice 5-6: 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.
      - Setting the `ip_forward` variable to `0` disables 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 = 24244   514
fs.inode-state = 24244   514  0  0  0  0  0
...
kernel.sched_child_runs_first = 0
kernel.sched_latency_ns = 6000000
...
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
...
vm.page-cluster = 3
```

```
vm.panic_on_oom = 0
...
```

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

```
# less /etc/sysctl.conf
# System default settings live in /usr/lib/sysctl.d/00-system.conf.
# To override those settings, enter new settings here, or in an
/etc/sysctl.d/<name>.conf file
#
# For more information, see sysctl.conf(5) and sysctl.d(5).
```

- 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, or in an `/etc/sysctl.d/<name>.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: Overview

Practices Overview

In these practices, you do the following:

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

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
...
gnome-color-manager-3.8.2-6.el7.x86_64
nmap-ncat-6.40-4.el7.x86_64
ghostscript-9.07-16.el7.x86_64
libdv-1.0.0-17.el7.x86_64
```

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

```
# rpm -q bash
bash-4.2.45-5.el7.x86_64
```

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

```
# rpm -qi bash
Name        : bash
Version     : 4.2.45
Release     : 5.el7
Architecture: x86_64
Install Date: <date_time>
Group       : System Environment/Shells
...
```

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

```
# rpm -ql bash
/etc/skel/.bash_logout
/etc/skel/.bash_profile
/etc/skel/.bashrc
/usr/bin/alias
/usr/bin/bash
...
```

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

```
# rpm -qf /etc/sysconfig
filesystem-3.2-18.el7.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/cups-files.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   ... /run/media/oracle/OL-7.0 Server.x86_64
```

- In this example, the Oracle Linux installation media is mounted on /run/media/oracle/OL-7.0 Server.x86_64.
- 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 -p command to create the /run/media/oracle/"OL-7.0 Server.x86_64" 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 -p /run/media/oracle/"OL-7.0 Server.x86_64"
# mount -o ro /dev/sr0 /run/media/oracle/"OL-7.0 Server.x86_64"
```

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

```
# cd /run/media/oracle/OL*
```

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

```
# ls
addons  EULA    GPL     isolinux  Packages  RPM-GPG-KEY ...
EFI     Extras  images  LiveOS    repodata  RPM-GPG-KEY-oracle
```

- 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-1.3.1.6-25.el7.x86_64.rpm
```

```
389-ds-base-libs-1.3.1.6-25.el7.x86_64.rpm
abattis-cantarell-fonts-0.0.12-3.el7.noarch.rpm
...
zsh-5.0.2-7.el7.x86_64.rpm
zziplib-0.13.62-5.el7.i686.rpm
zziplib-0.13.62-5.el7.x86_64.rpm
```

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

```
# zsh
bash: zsh: command not found...
Similar command is: 'ssh'
# rpm -q zsh
Package zsh is not installed
```

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

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

```
# rpm -Uvh zsh-5.0.2-7.el7.x86_64.rpm
Preparing...                ################################# [100%]
Updating / installing...
   1:zsh-5.0.2-7.el7        ################################# [100%]
```

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

```
# rpm -q zsh
zsh-5.0.2-7.el7.x86_64
```

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

```
# zsh
# ps
  PID  TTY          TIME CMD
29038  pts/0    00:00:00 zsh
29062  pts/0    00:00:00 ps
30106  pts/0    00:00:00 su
30115  pts/0    00:00:02 bash
```

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

```
# exit
# ps
  PID  TTY          TIME CMD
29071  pts/0    00:00:00 ps
30106  pts/0    00:00:00 su
30115  pts/0    00:00:02 bash
```

3. Remove packages by using `rpm`.

a. Remove the `zsh` package.

```
# rpm -e zsh
```

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

```
# rpm -q zsh
package zsh is not installed
# zsh
-bash: /bin/zsh: No such file or directory
```

Practice 6-2: Accessing the Public Yum Server

Overview

In this practice, you review the configuration required to access the Public Yum Server from **host03** VM. You then simulate upgrading your system (**you *do not* actually perform the upgrade**) from Public Yum.

Assumptions

- You are the `root` user on **host03** VM.
- You are the `root` user on **dom0**.

Tasks

This task refers to the following networking-related files and service. These are discussed further in the lesson titled "Network Configuration":

- Network Manager service.
- Domain Name Service (DNS) resolver configuration file, `/etc/resolv.conf`.
- Network interface configuration files, `/etc/sysconfig/network-scripts/ifcfg*`.

1. On **host03**, review the Domain Name Service (DNS) configuration.

   The DNS resolver configuration file, `/etc/resolv.conf`, provides access to DNS.

   a. Use the `cat` command to view the `/etc/resolv.conf` file.

```
# cat /etc/resolv.conf
# Generated by NetworkManager
search example.com
nameserver 152.68.154.3
nameserver 10.216.106.3
```

   - 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 an `/etc/sysconfig/network-scripts/ifcfg*` file.
   - Therefore, the DNS information needs to be specified in an `ifcfg*` file.

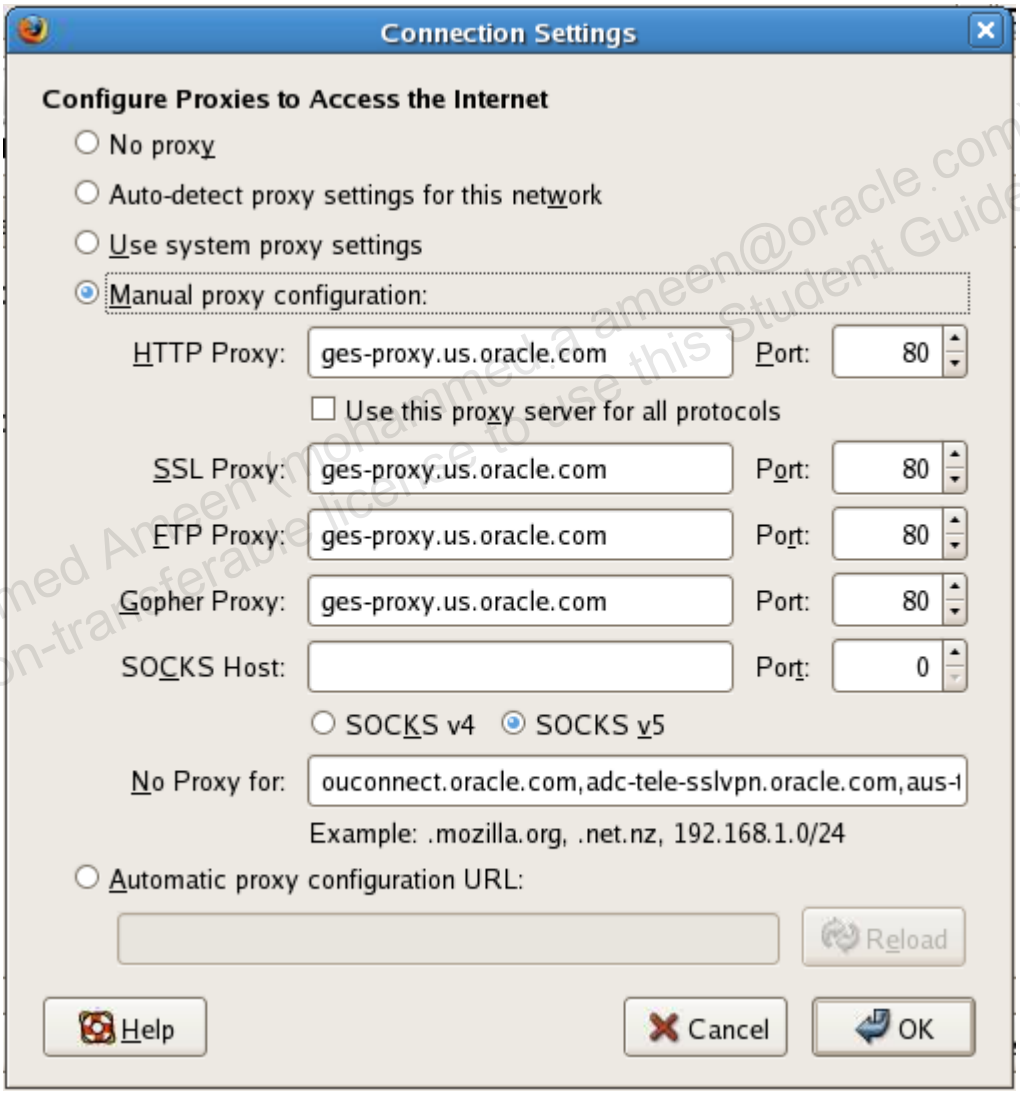
   b. Review the `DOMAIN` directive and the `DNS1`and `DNS2` directives in the `/etc/sysconfig/network-scripts/ifcfg-eth0` file.

```
# cat /etc/sysconfig/network-scripts/ifcfg-eth0
...
DNS1=152.68.154.3
DNS2=10.216.106.3
DOMAIN=example.com
...
```

   - The `DOMAIN` directive in the `ifcfg-eth0` file corresponds to the `search` directive in `/etc/resolv.conf`.
   - The `DNS#` directives in the `ifcfg-eth0` file corresponds to the `nameserver` directives in `/etc/resolv.conf`.

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 in the Firefox Preferences window, and then select the **Network** tab.
   d. Click the **Settings** button. The Connection Settings window appears.



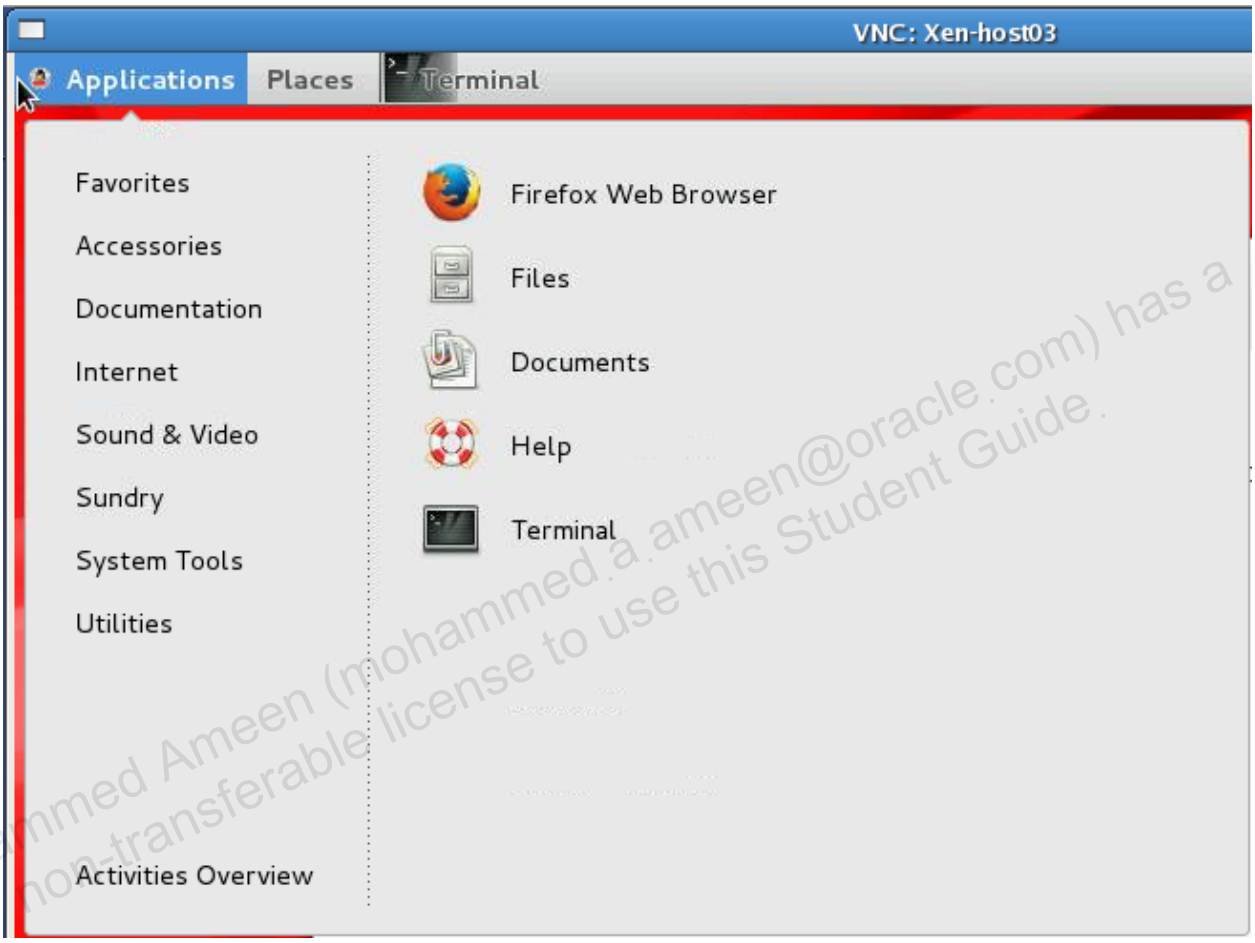
   e. Note the **HTTP Proxy** setting.
      - In this example, the setting is `ges-proxy.us.oracle.com` on Port `80`.
   f. Click **Cancel** to close the **Connection Settings** window.
   g. Click **Close** to close the **Firefox Preferences** window.

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 on **host03**.
   - In this example, the **HTTP Proxy** is `ges-proxy.us.oracle.com`.

   a. From the **host03** GNOME menu bar, click **Applications**, and then click the **Firefox Web Browser** icon.



   - 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.

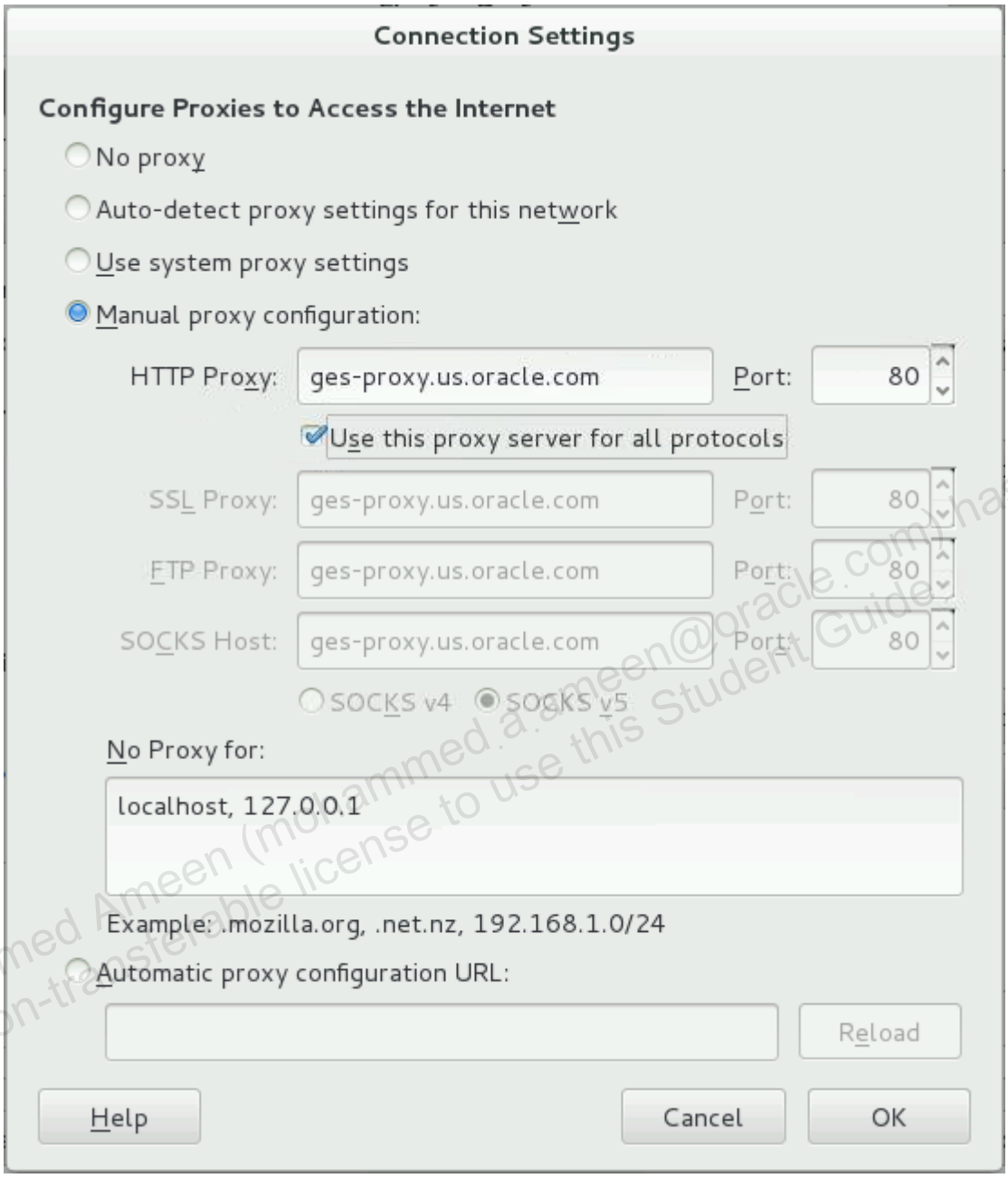
   d. Click the **Settings** tab and enter the following settings:

      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**):



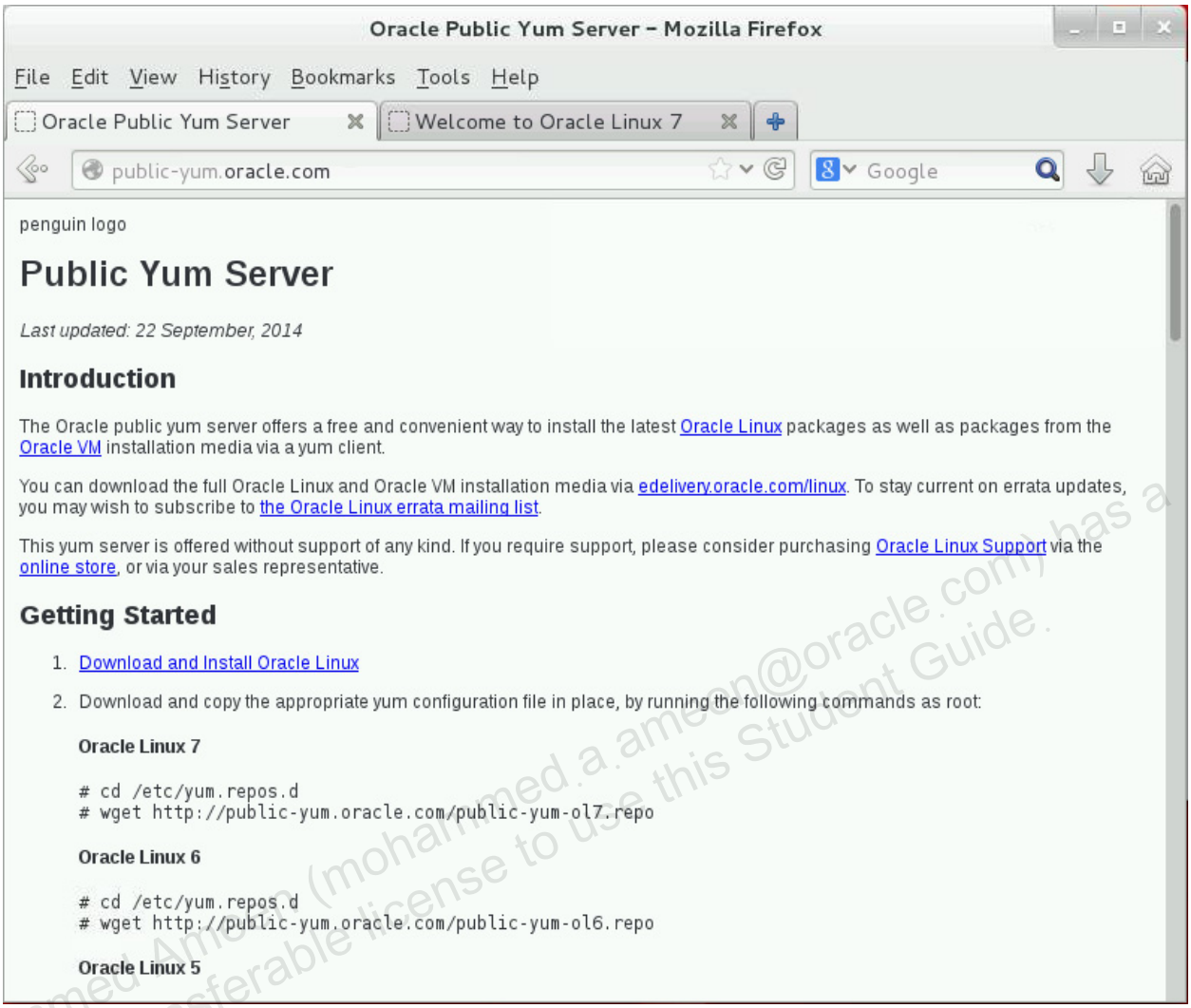
e. Click **OK**.

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. View the instructions for **Oracle Linux 7**, which are:

```
cd /etc/yum.repos.d
wget http://public-yum.oracle.com/public-yum-ol7.repo
```

- This is information only; you do not need to run these commands.
- The `public-yum-ol7.repo` file is already present on **host03**.

c. Close the browser by selecting **File > Quit**.

5. Set the `http_proxy` variable in the `/etc/profile` file.

a. Use the `vi` command to add the `http_proxy` variable definition to the `/etc/profile` file.

- Add the entry to the bottom of the `/etc/profile` file.
- This proxy variable is needed to access the Public Yum Server from your classroom machine.

```
# vi /etc/profile
export http_proxy=http://ges-proxy.us.oracle.com:80
```

b. Use the `source` command to execute the commands from the `/etc/profile` file in the current shell.

- Run the `echo $http_proxy` command before and after the `source` command to verify that the variable is set.

```
# echo $http_proxy

# source /etc/profile
# echo $http_proxy
http://ges-proxy.us.oracle.com:80
```

- In this example, the variable is set correctly.
- If necessary, repeat tasks 5a and 5b to set the variable.

6. Review the Public Yum Server repository file on **host03**.
   a. Use the `cd` command to change to the `/etc/yum.repos.d` directory. Use the `ls` command to display the contents of the directory.

```
# cd /etc/yum.repos.d
# ls
public-yum-ol7.repo
```

   - Note that the `public-yum-ol7.repo` file already exists in this directory.

   b. Use the `less` command to view the `public-yum-ol7.repo` file.
   - Your system is configuring to access the latest Oracle Linux 7 repository on the Public Yum Server, `[ol7_latest]` and the Oracle Linux 7 UEK R3 repository, `[ol7_UEKR3]`.
   - These are enabled by the `enabled=1` setting.
   - All other repositories are disabled by the `enabled=0` setting.

```
# less public-yum-ol7.repo
[ol7_latest]
...
enabled=1
...
[ol7_UEKR3]
...
enabled=1
...
```

   c. Run the `yum repolist` command to list the configured repositories.

```
# yum repolist
Loaded plugins: langpacks
repo id            Repo name                                  status
ol7_UEKR3/x86_64  Latest Unbreakable Enterprise Kernel ...
ol7_latest/x86_64 Oracle Linux 7Server Latest (x86_64) ...
```

   - Note that the two repositories correspond to the two "`enabled=1`" settings in the `public-yum.ol7.repo` file.

7. Run the `yum update` command to display available updates.

DO NOT APPLY UPDATES!

- ***Do not apply updates. It takes a couple hours to do so.***

- Answer **NO** (**n**) when asked, “Is this ok”.

```
# yum update
...
Transaction Summary
============================================================
Install ...
Upgrade ...
Total download size: ...
Is this ok [y/d/N]: n
Exiting on user Command
Your transaction was saved, rerun it with:
 yum load-transaction /tmp/yum_save_tx...
```

- 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.***

Practice 6-3: Creating a Local Yum Repository

Overview

In this practice, you use the `createrepo` utility to 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 `/run/media/oracle/`.

Tasks

1. Disable the Public Yum repositories.
   a. Use the `vi` editor to edit the `/etc/yum.repos.d/public-yum-ol7.repo` file and set all "`enabled=1`" to "`enabled=0`".

```
# vi /etc/yum.repos.d/public-yum-ol7.repo
[ol7_latest]
...
enabled=1                              (old value)
enabled=0                              (new value)
...
[ol7_UEKR3]
...
enabled=1                              (old value)
enabled=0                              (new value)
...
```

   b. To ensure all repositories are disabled, use the `grep` command and search for the string "`enabled`" in the `/etc/yum.repos.d/public-yum-ol7.repo` file.

```
# grep enabled /etc/yum.repos.d/public-yum-ol7.repo
enabled=0
enabled=0
enabled=0
enabled=0
enabled=0
enabled=0
enabled=0
```

   - In this example, all repositories are disabled (`enabled=0`).
   - Repeat task 1a and task 1b if necessary to ensure that all repositories are disabled.

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

```
# yum clean all
There are no enabled repos.
 Run "yum repolist all" to see the repos you have.
 You can enabled repos with yum-config-manager –enabled <repo>
```

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             3.9G  3.9G     0  100% /run/media/oracle/OL-
7.0 Server.x86_64
```

   - Note that the OL7 media is mounted on `/run/media/oracle/`.

3. Create the local repository.
   a. Use the `rpm` command to check whether the `createrepo` package is installed.

```
# rpm -q createrepo
createrepo-0.9.9-23.el7.noarch
```

      - In this example, the package is installed.

   b. Change to the `/run/media/oracle` directory.

```
# cd /run/media/oracle
# ls
OL-7.0 Server.x86_64
```

   c. Use the `createrepo` command to create a repository of the current directory.
      - This command takes several minutes to complete.
      - Include the "." argument to represent the current directory.

```
# createrepo .
Spawning worker 0 with 4364 pkgs
Workers Finished
Saving Primary metadata
Saving file lists metadata
Saving other metadata
Generating sqlite DBs
Sqlite DBs complete
```

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

```
# pwd
/run/media/oracle
# ls -l
drwx-r-x-r-x.        OL-7.0 Server.x86_64
drwx-r-x-r-x.        repodata
```

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

   e. View the contents of the `repodata` directory.

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

```
-rw-r--r--   ...other.sqlite.bz2
-rw-r--r--   ...other.xml.gz
-rw-r--r--   repomd.xml
```

f. Use the `cd` command to change to the `yum` repository directory. Use the `vi` editor to create the `iso.repo` file:

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

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

```
# cd /run/media/oracle/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`.

- Copy the `RPM-GPG-KEY-oracle` file because this is the file you designated in the `iso.repo` file.

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

i. Manually install the public key.

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

j. Confirm the import of the public key.

```
# rpm -q gpg-pubkey
gpg-pubkey-...
```

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

- Note that the "`Myrepo`" repository is the only enabled repository.

```
# yum repolist
...
repo id                    repo name                    status
Myrepo                     Oracle Linux                 4,364
repolist: 4,364
```

Practice 6-4: 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.
- You are the `root` user on **host03** VM.

Tasks

1. List packages by using `yum`.
   a. List all packages installed on your system and all packages available in all configured repositories.
      - Notice that some packages were installed during installation (`@anaconda/7.0`) and some are available in the new repository (`Myrepo`).

```
# yum list
Loaded plugins: langpacks
Installed Packages
GConf2.x86_64                3.2.6-8.el7          @anaconda/7.0
ModemManager.x86_64          1.1.0-6.git...       @anaconda/7.0
ModemManager-glib.x86_64     1.1.0-6.git...       @anaconda/7.0
...
zsh.x86_64                   5.0.2-7.el7          Myrepo
zziplib.i686                 0.13.62-5.el7        Myrepo
zziplib.x86_64               0.13.62-5.el7        Myrepo
```

   b. List only the installed packages.

```
# yum list installed
Loaded plugins: langpacks
Installed Packages
GConf2.x86_64                3.2.6-8.el7          @anaconda/7.0
ModemManager.x86_64          1.1.0-6.git...       @anaconda/7.0
ModemManager-glib.x86_64     1.1.0-6.git...       @anaconda/7.0
...
zenity.x86_64                3.8.0-4.el7          @anaconda/7.0
zip.x86_64                   3.0-10.el7           @anaconda/7.0
zlib.x86_64                  1.2.7-13.el7         @anaconda/7.0.
```

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

```
# yum list available
Available Packages
389-ds-base.x86_64             1.3.1.6-25.el7       Myrepo
389-ds-base-libs.x86_64        1.3.1.6-25.el7       Myrepo
```

```
ElectricFence.i686              2.2.2-39.el7          Myrepo
...
zsh.x86_64                      5.0.2-7.el7           Myrepo
zziplib.i686                    0.13.62-5.el7         Myrepo
zziplib.x86_64                  0.13.62-5.el7         Myrepo
```

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

```
# yum provides /etc/sysconfig/crond
cronie-1.4.11-11.el7.x86_64 : Cron daemon for executing prog...
Repo           : Myrepo
...
```

2. Install packages by using `yum`.

Install the `389-ds-base` package.

- Answer **`y`** when prompted.

```
# yum install 389-ds-base
Loaded plugins: langpacks
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    1 Package (+8 Dependent packages)
Total download size: 3.3 M
Installed size: 10 M
Is this ok [y/d/N]: y
Downloading packages
...
Running transaction
  Installing : svrcore-4.0.4-11.el7.x86_64
...
Installed:
 389-ds-base.x86_64 0:1.3.1.6-25.el7
```

```
Dependency Installed:
 389-ds-base-libs.x86_64 0:1.3.1.6-25.el7
...
Complete!
```

3. Update packages by using `yum`.
   a. Check which installed packages have updates available.
      - This example shows `mariadb-lib` has an update available.
      - If interested in using MySQL, see http://dev.mysql.com/doc/mysql-yum-repo-quick-guide/en/index.html.

```
# yum check-update
Loaded plugins: langpacks
Obsoleting Packages
mysql-community-libs.x86_64    5.6.19-3.el7        Myrepo
    mariadb-libs.x86_64        1:5.5.35-3.el7      Myrepo
```

   b. Update the `mariadb-libs` package.
      - Answer **`y`** when prompted.

```
# yum update mariadb-libs
...
Transaction Summary
================================================
Upgrade     1 Package (+1 Dependent package)

Total download size: 2.2 M
Is this ok [y/d/N]: y
...
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
...
Transaction Summary
================================================
Remove     1 Package

Installed size: 5.0 M
Is this ok [y/d/N]: y
...
Complete!
```

Practice 6-5: Unbreakable Linux Network (ULN)

Overview

In this practice, you become familiar with registering a system with ULN, and 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

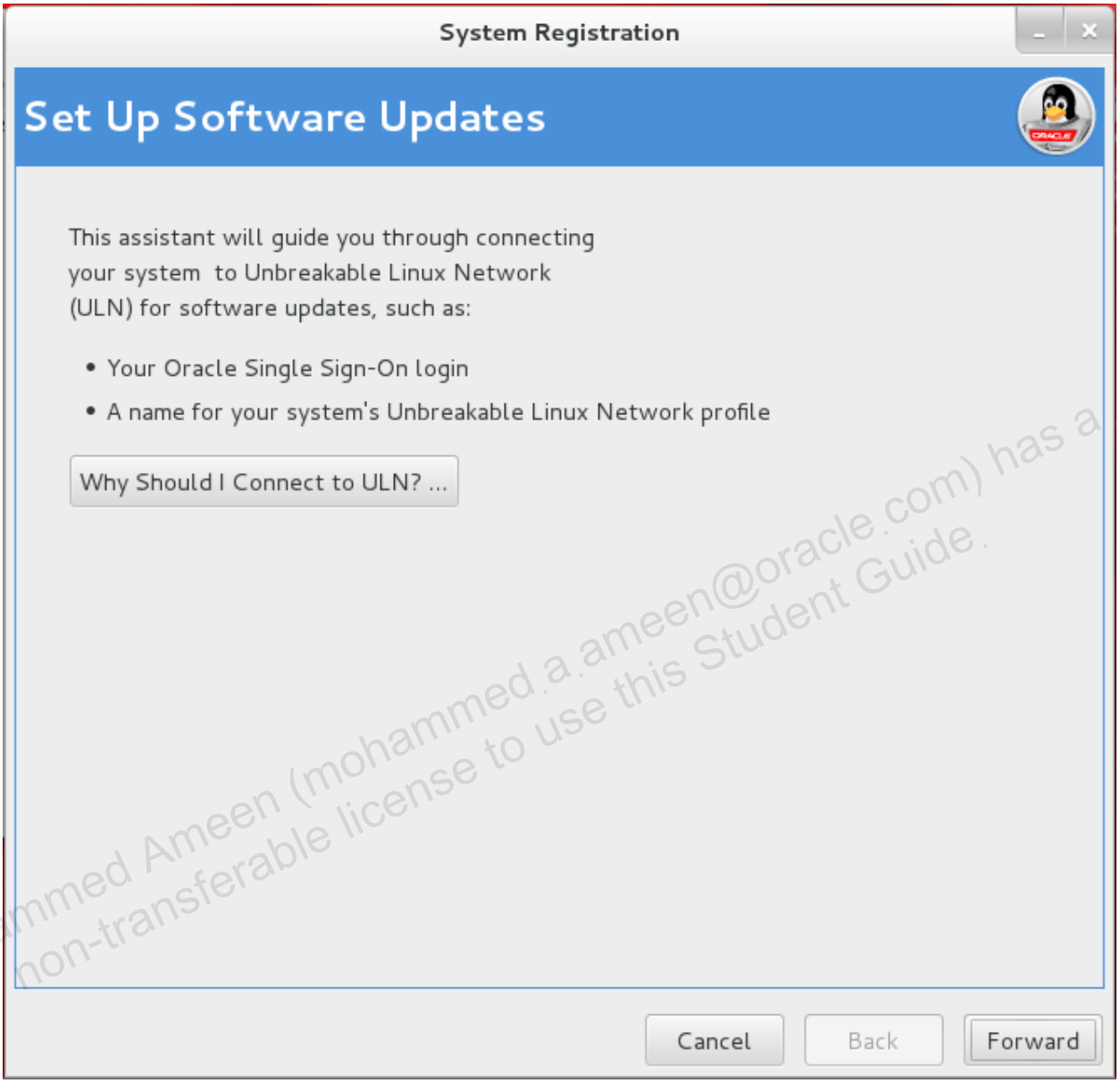
This practice is not intended to be a hands-on exercise.

Tasks

1. Register your system with the Unbreakable Linux Network (ULN).
   a. Run the `uln_register` command to begin the registration process.

```
# uln_register
```

The following window appears:



b. Click the **Why Should I Connect to ULN?** button.

The following window appears:

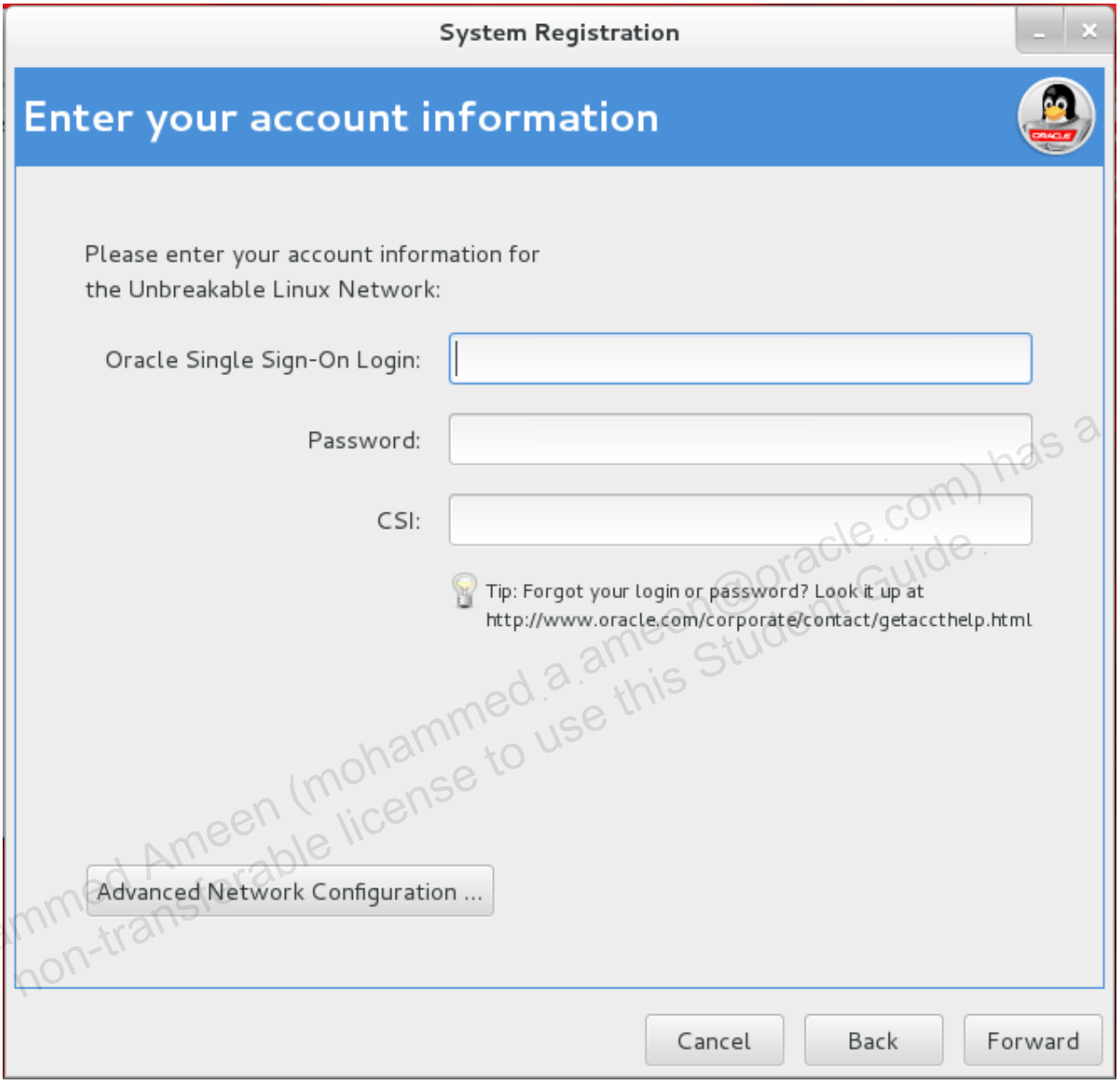
- This window is for information only.

c. Click **Take me back to the registration**.

- The window shown in task 1a reappears.

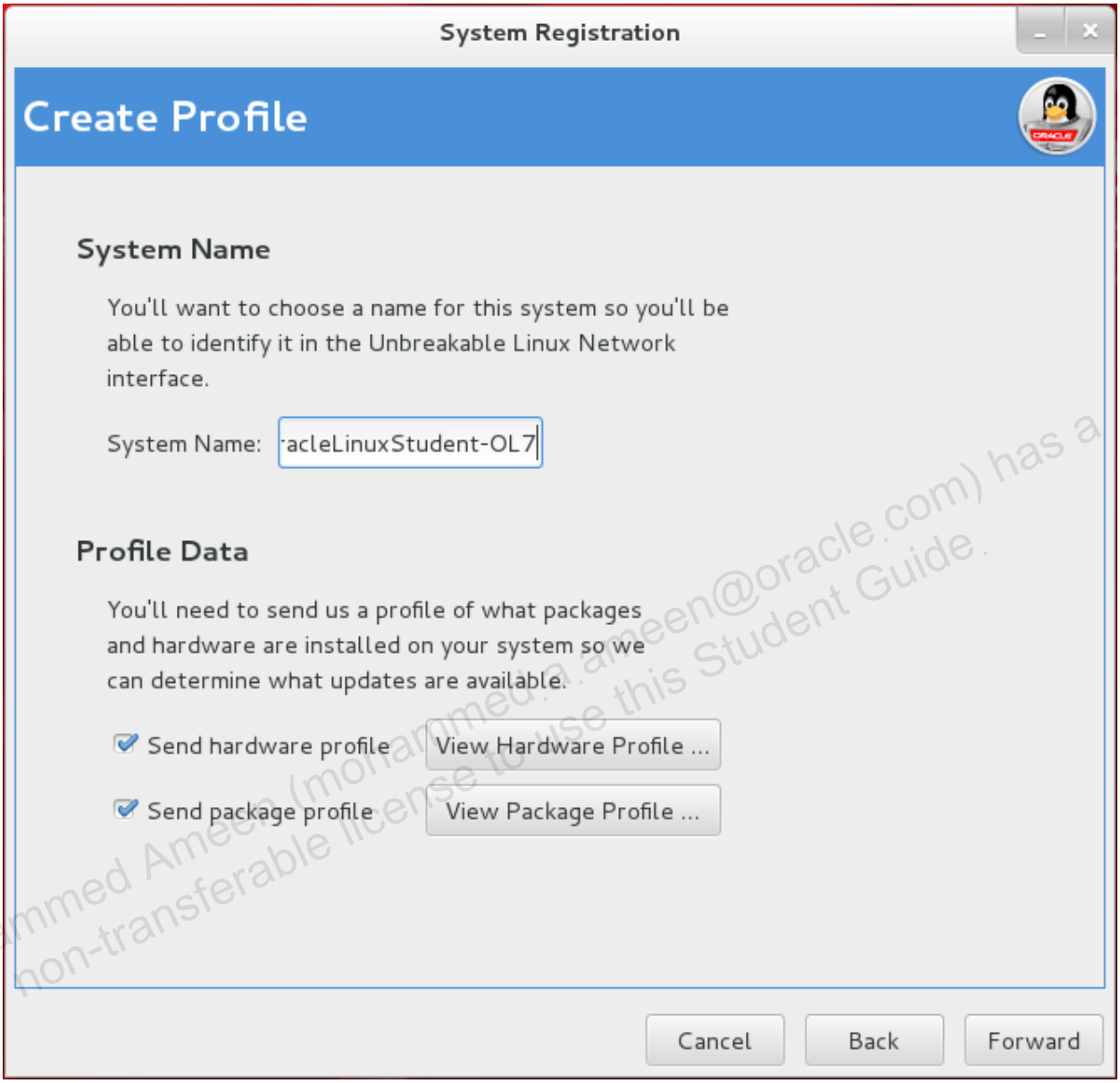
d. Click **Forward**.

The following window appears:



e. Enter your **Login**, **Password**, and **CSI** information. Click **Forward**.

The following window appears:

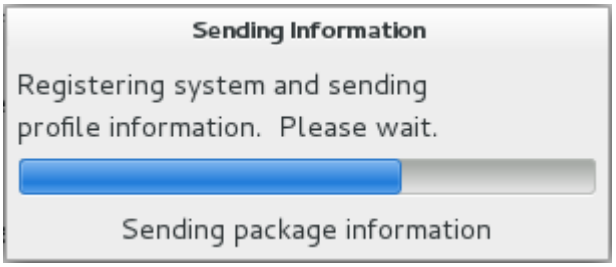


f. Change your **System Name** (if necessary).
   - In this example, the System Name is set to **OracleLinuxStudent-OL7** although only part of the System Name is displayed.

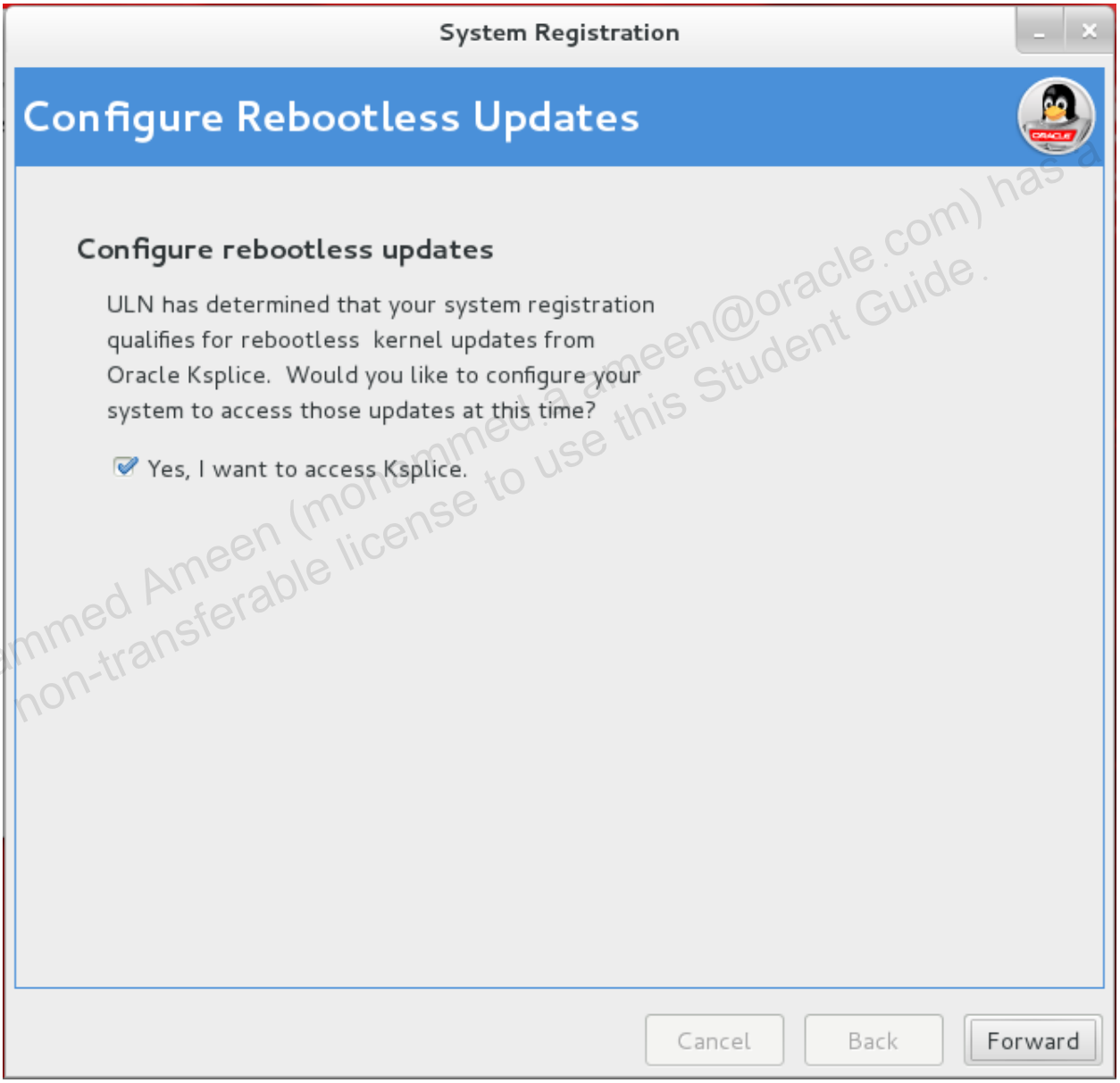
g. Ensure the **Send hardware profile** and the **Send package profile** check boxes are selected.
   - Optionally, click the **View Hardware Profile** and **View Package Profile** buttons to display the information that will be sent.

h. Click **Forward**.

The **Sending Information** window is displayed.



After the information is sent, the following window appears:

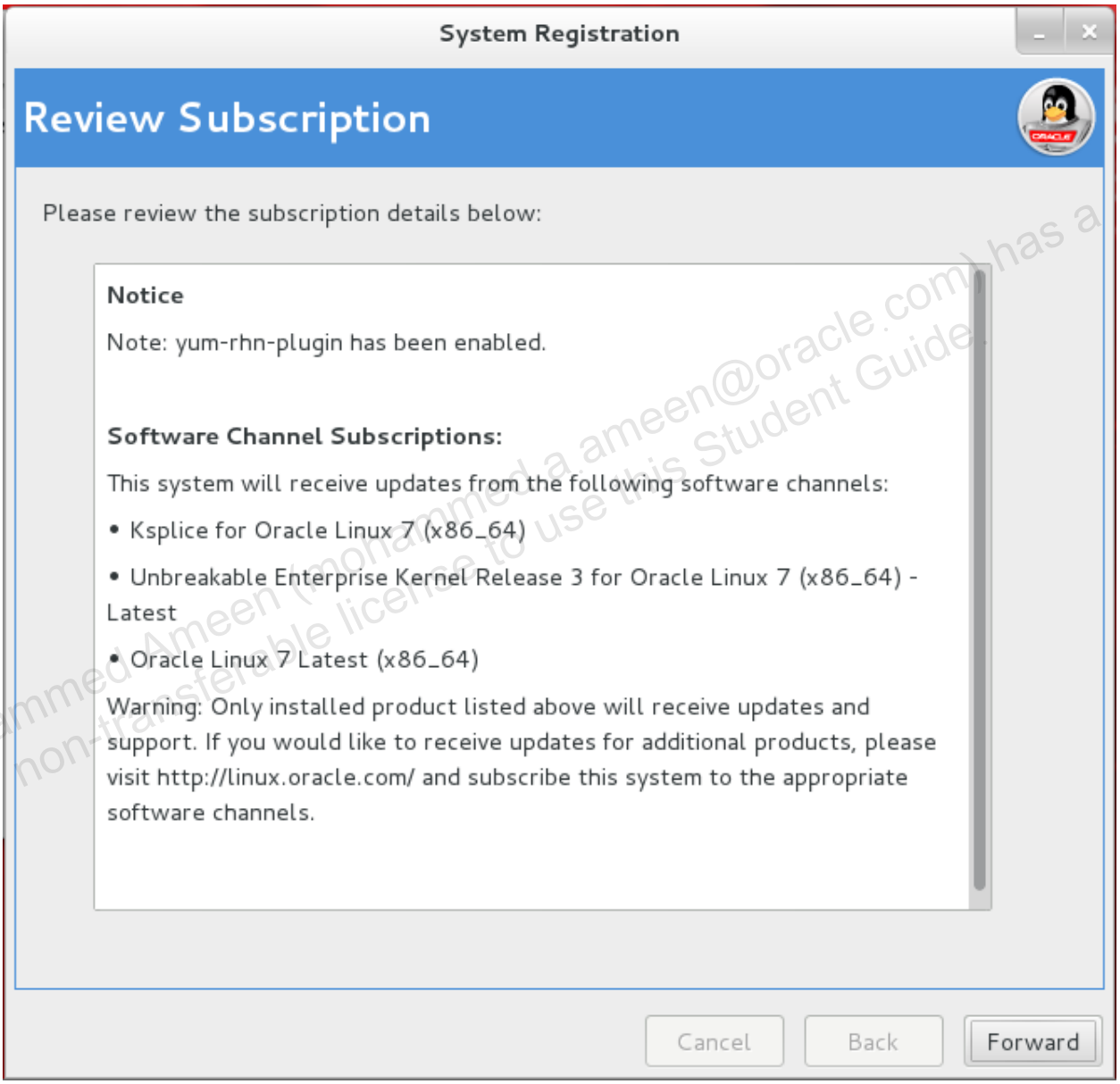


i. Select the **Yes, I want to access Ksplice** check box, which is selected by default.
   - Ksplice is discussed in Lesson 7.

j. Click **Forward**.

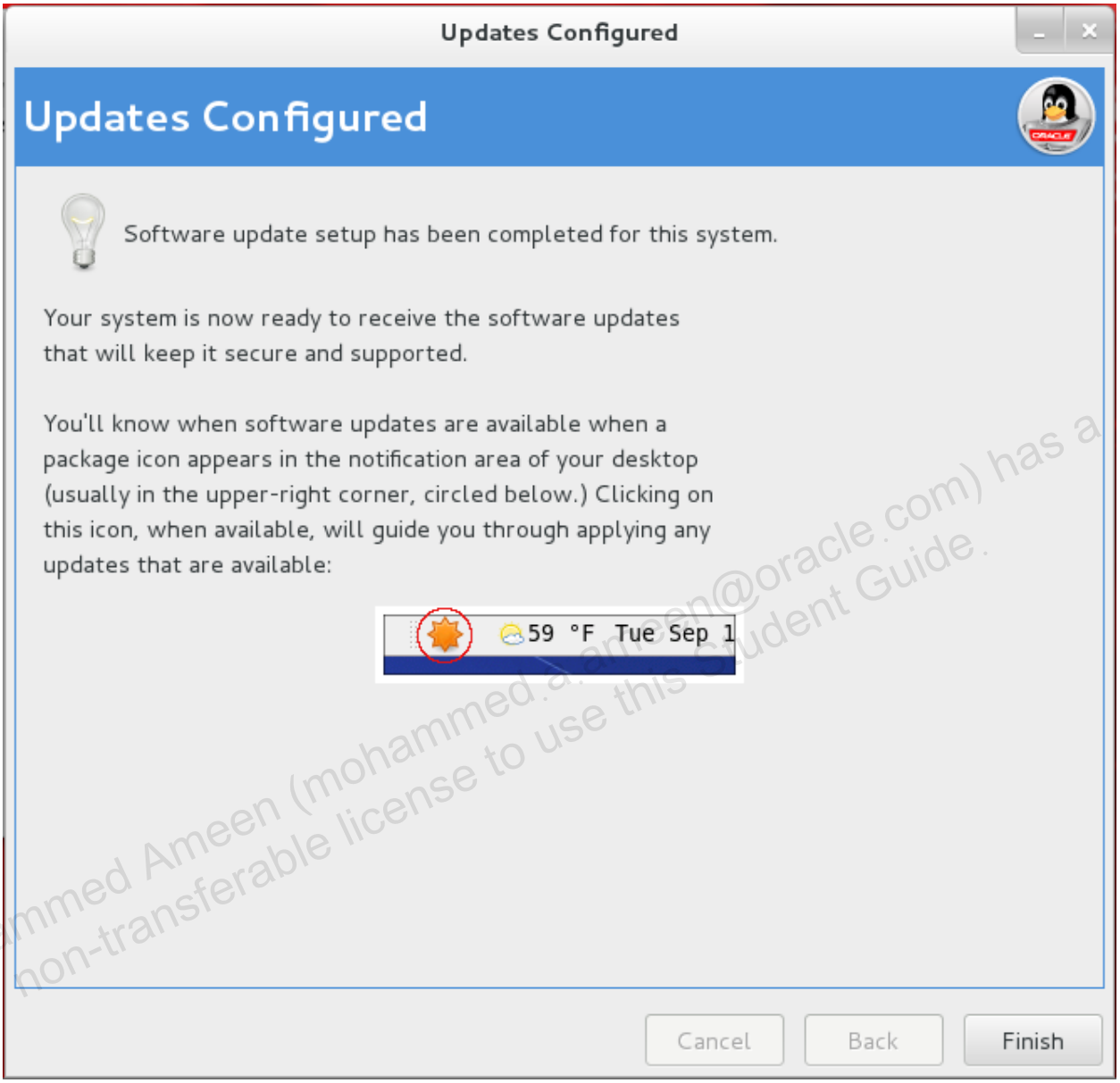
The following window appears:

- Notice that when you register for the first time, you are subscribed to the default channel of `<OL_version>_<arch>_latest`., which is `ol7_x86_64_latest`.
- Also notice that you are subscribed to the Ksplice channel, `ol7_x86_64_ksplice`, because you clicked **Yes, I want to access Ksplice** in the previous window.
- After registering, you can specify additional, or different, channels by using the web interface.



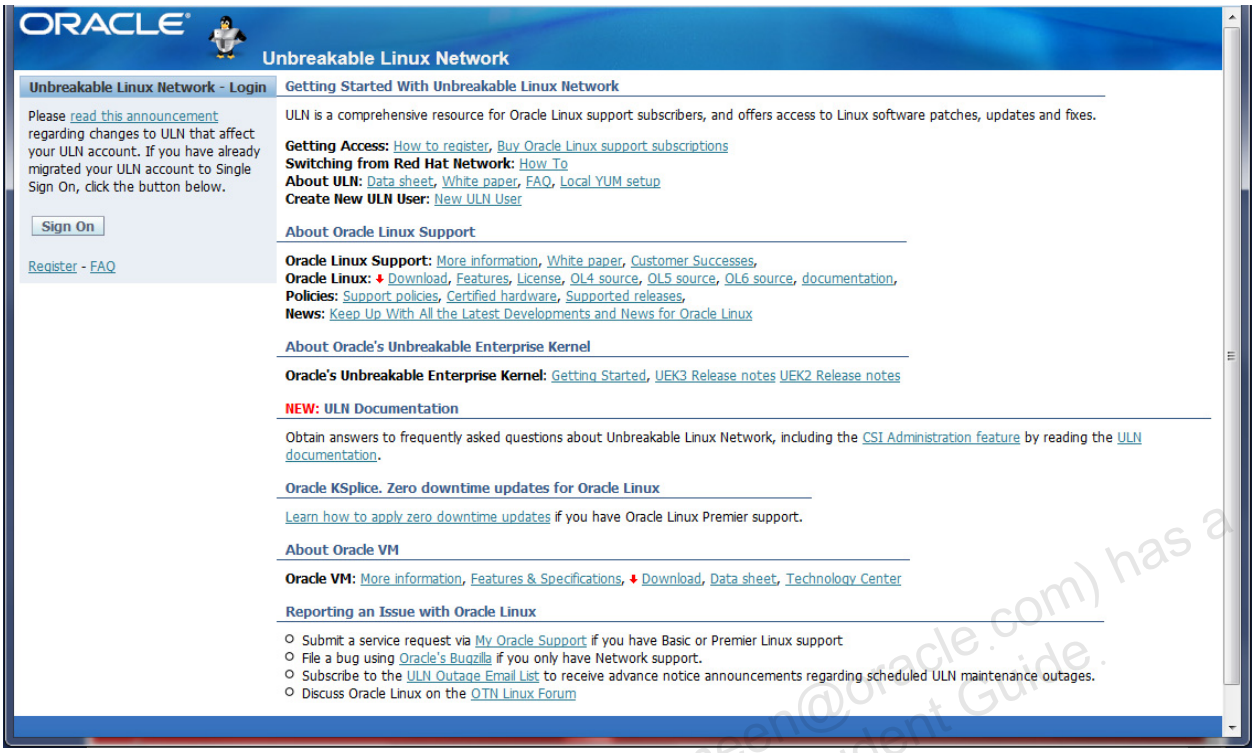
k. Click **Forward**.

The following window appears:



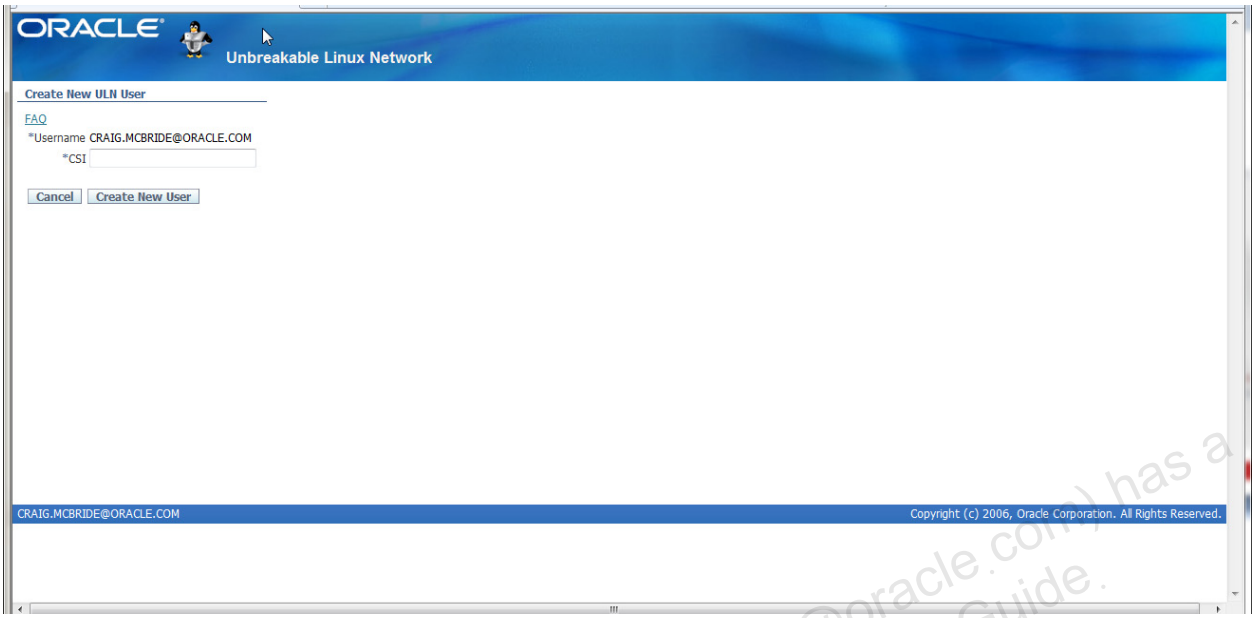
l. Click **Finish**.

2. Access the Unbreakable Linux Network (ULN) web interface.
   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.

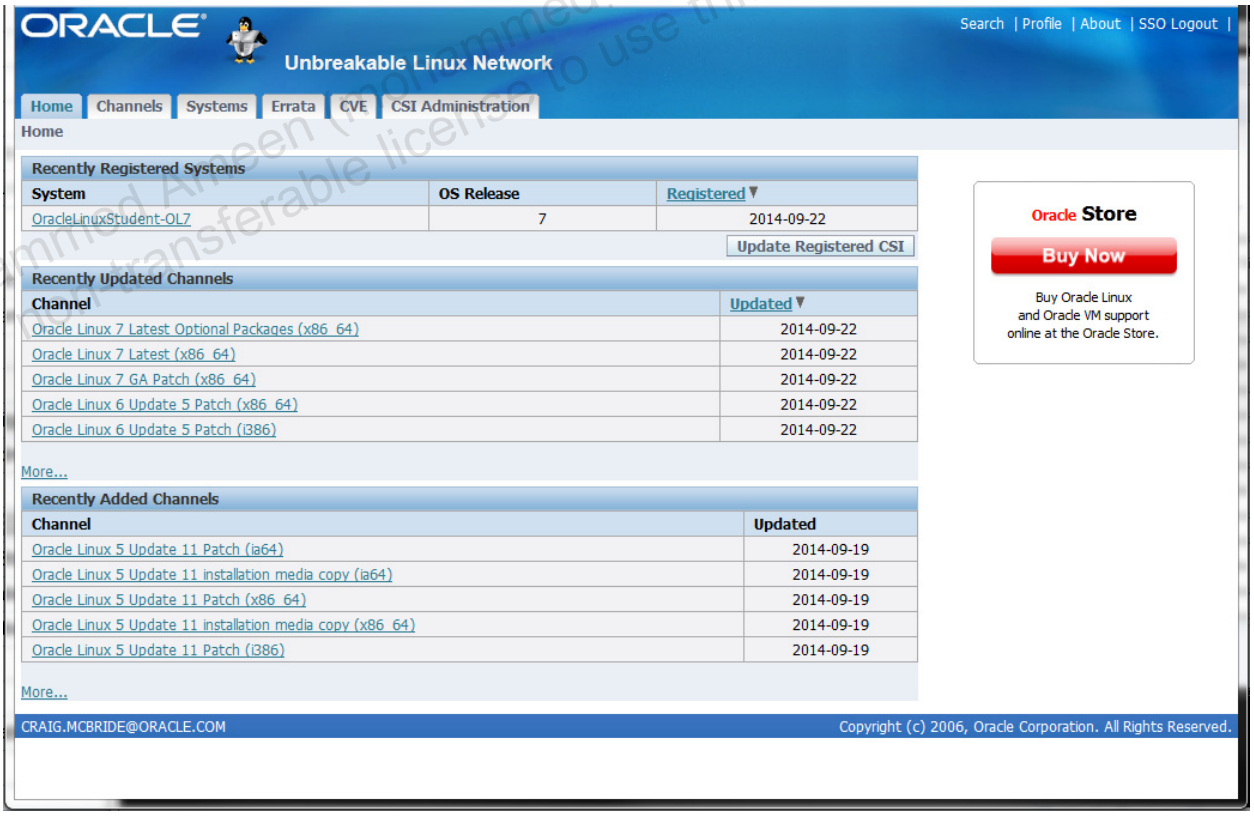
c. The first time you access the ULN, use your email address and your Customer Support Identifier (CSI) to log in. You are then required to create a password.



- In the future, use your email address and password to log in to ULN.

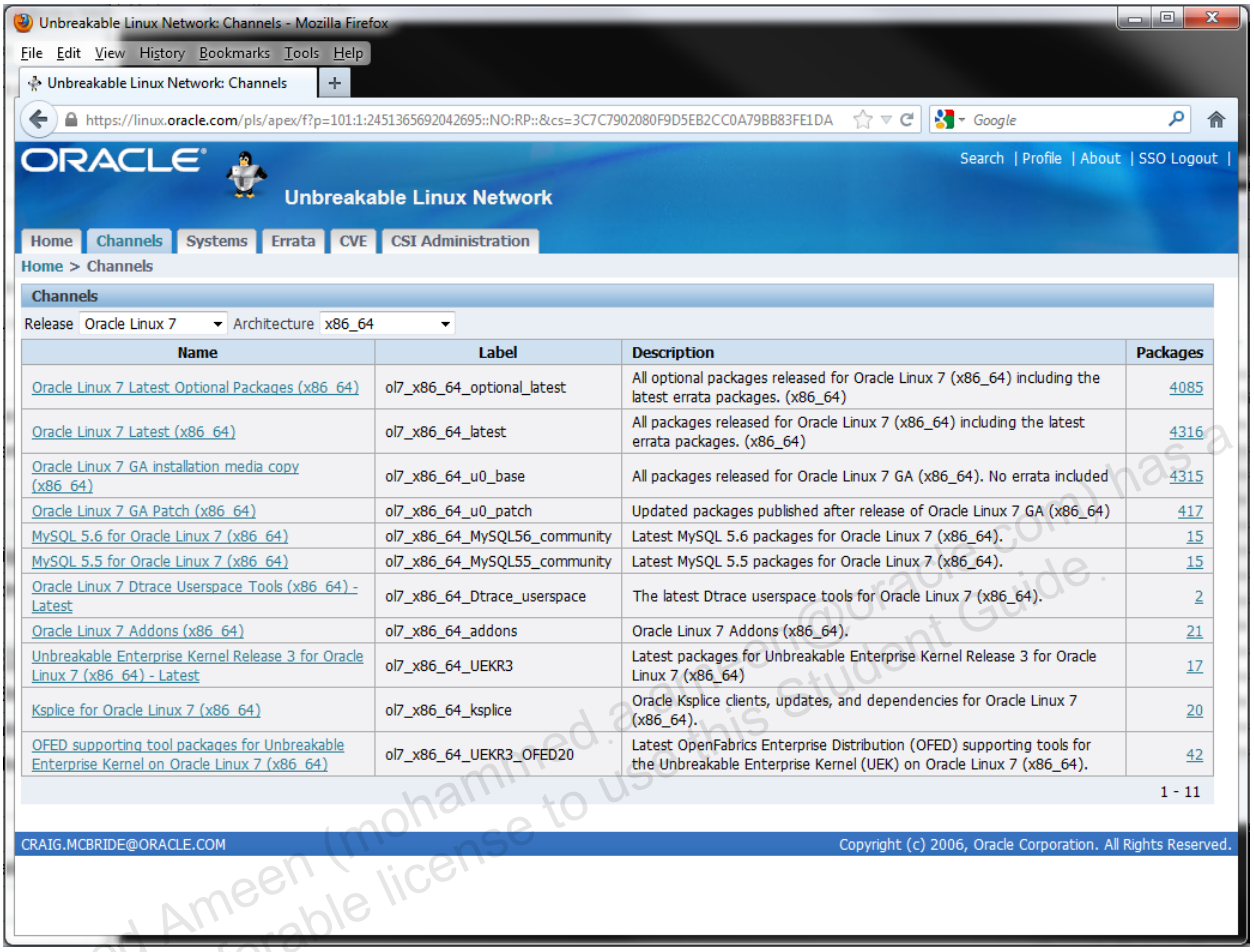
3. Use the ULN web interface.

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



- Note that the system, **OracleLinuxStudent-OL7**, is registered.

b. Click the **Channels** tab for a list of all the channels available via ULN.
The following window is displayed.



- 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 7 x86_64 architecture.

c. Click the **Systems** tab to see **System Profiles** of the registered systems.

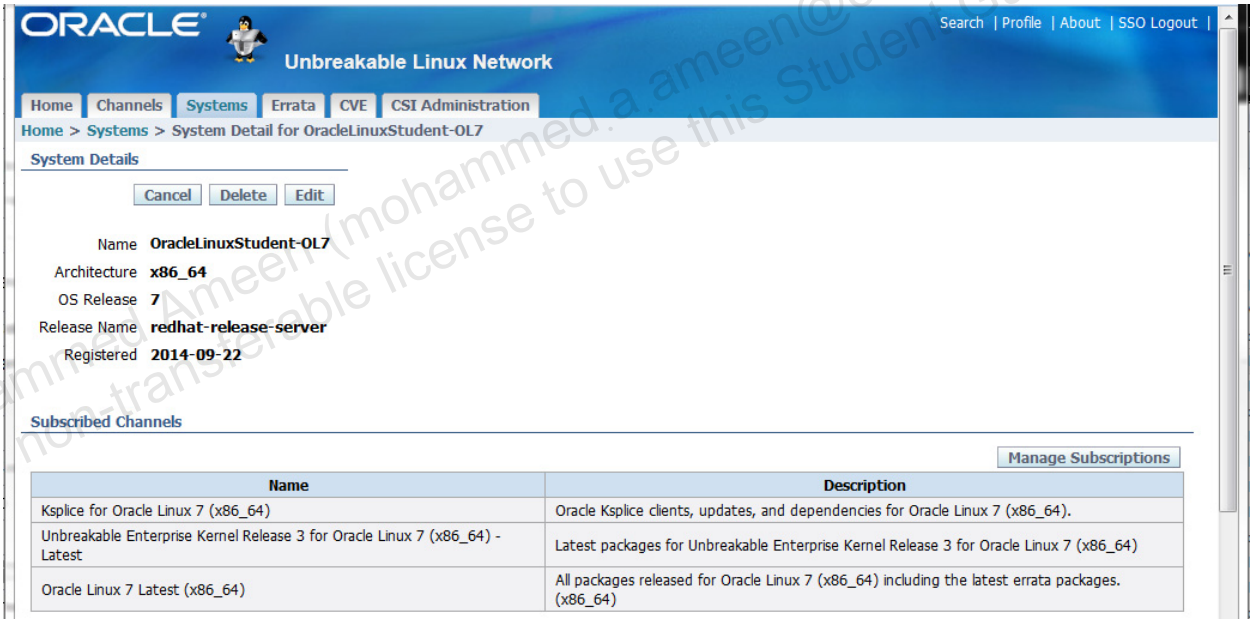
The following window is displayed.

- The **CSI** and **Ksplice Access Key** are blacked out in this view.
- The **Ksplice Access Key** is discussed in Lesson 7 – Ksplice.

d. Click the **OracleLinuxStudent-OL7** system link.

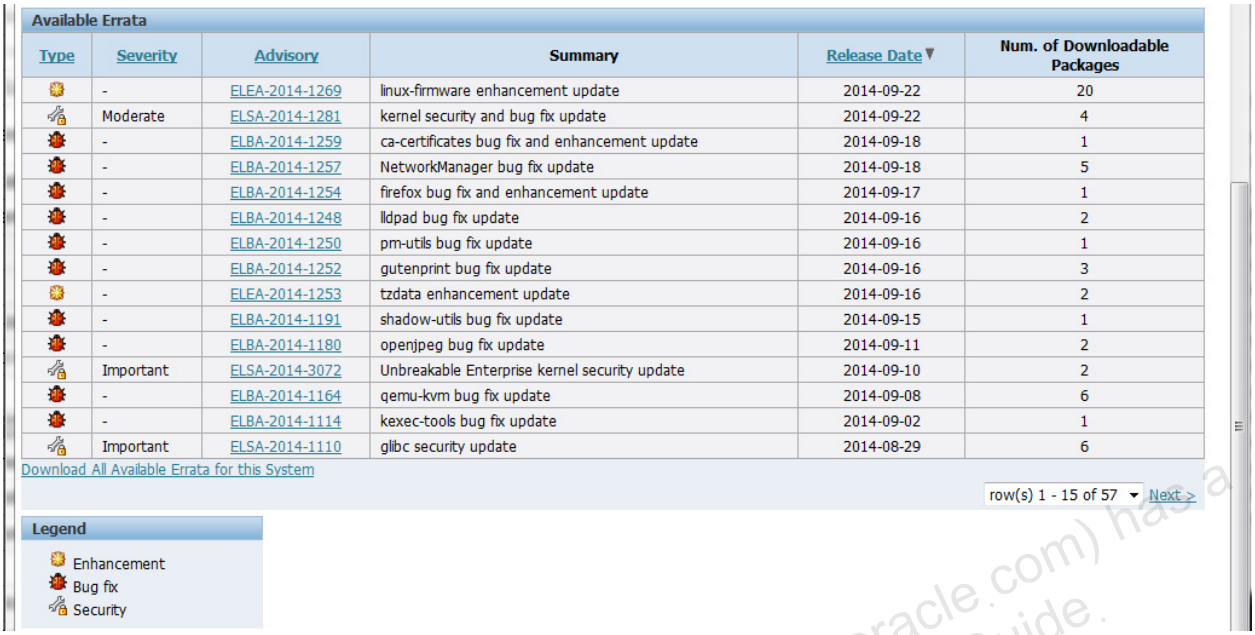
The following window is displayed.

- This window shows only the top portion of the window.



- From the top portion of the window, you can update **System Details**.
- You can also **Manage Subscriptions**, that is, subscribe to or unsubscribe from other available channels.

Scroll down to display the bottom portion of the window.



Available Errata

Type	Severity	Advisory	Summary	Release Date	Num. of Downloadable Packages
	-	ELEA-2014-1269	linux-firmware enhancement update	2014-09-22	20
	Moderate	ELSA-2014-1281	kernel security and bug fix update	2014-09-22	4
	-	ELBA-2014-1259	ca-certificates bug fix and enhancement update	2014-09-18	1
	-	ELBA-2014-1257	NetworkManager bug fix update	2014-09-18	5
	-	ELBA-2014-1254	firefox bug fix and enhancement update	2014-09-17	1
	-	ELBA-2014-1248	lldpad bug fix update	2014-09-16	2
	-	ELBA-2014-1250	pm-utils bug fix update	2014-09-16	1
	-	ELBA-2014-1252	gutenprint bug fix update	2014-09-16	3
	-	ELEA-2014-1253	tzdata enhancement update	2014-09-16	2
	-	ELBA-2014-1191	shadow-utils bug fix update	2014-09-15	1
	-	ELBA-2014-1180	openjpeg bug fix update	2014-09-11	2
	Important	ELSA-2014-3072	Unbreakable Enterprise kernel security update	2014-09-10	2
	-	ELBA-2014-1164	qemu-kvm bug fix update	2014-09-08	6
	-	ELBA-2014-1114	kexec-tools bug fix update	2014-09-02	1
	Important	ELSA-2014-1110	glibc security update	2014-08-29	6

Download All Available Errata for this System

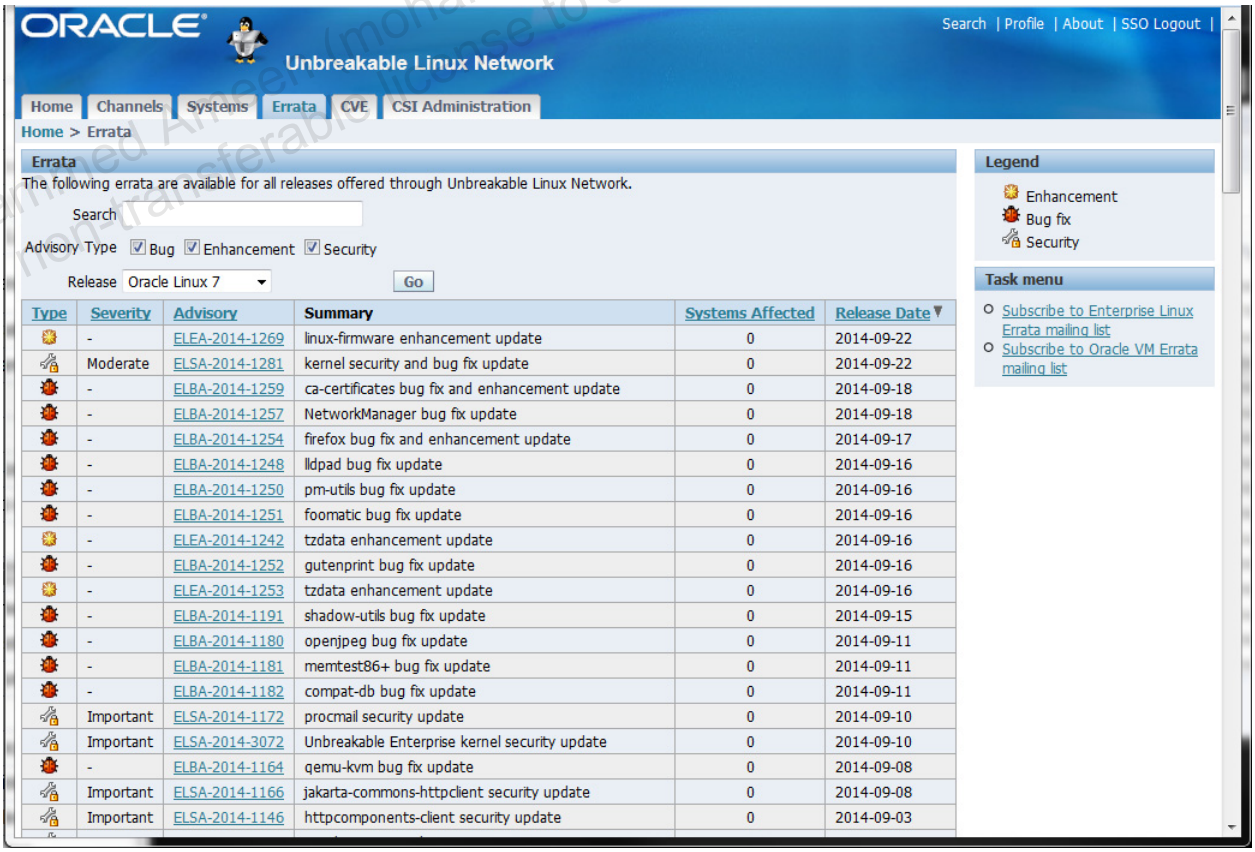
row(s) 1 - 15 of 57 Next >

Legend

Enhancement
Bug fix
Security

- From the bottom portion of the window, you can see the available errata for the system categorized by **Enhancement**, **Bug fix**, and **Security**.
- You can click the **Advisory** link to display details about a particular errata.
- You can also **Download All Available Errata for this System**.

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



ORACLE Unbreakable Linux Network

Search | Profile | About | SSO Logout |

Home | Channels | Systems | Errata | CVE | CSI Administration

Home > Errata

Errata

The following errata are available for all releases offered through Unbreakable Linux Network.

Search

Advisory Type Bug Enhancement Security

Release Oracle Linux 7 Go

Type	Severity	Advisory	Summary	Systems Affected	Release Date
	-	ELEA-2014-1269	linux-firmware enhancement update	0	2014-09-22
	Moderate	ELSA-2014-1281	kernel security and bug fix update	0	2014-09-22
	-	ELBA-2014-1259	ca-certificates bug fix and enhancement update	0	2014-09-18
	-	ELBA-2014-1257	NetworkManager bug fix update	0	2014-09-18
	-	ELBA-2014-1254	firefox bug fix and enhancement update	0	2014-09-17
	-	ELBA-2014-1248	lldpad bug fix update	0	2014-09-16
	-	ELBA-2014-1250	pm-utils bug fix update	0	2014-09-16
	-	ELBA-2014-1251	foomatic bug fix update	0	2014-09-16
	-	ELEA-2014-1242	tzdata enhancement update	0	2014-09-16
	-	ELBA-2014-1252	gutenprint bug fix update	0	2014-09-16
	-	ELEA-2014-1253	tzdata enhancement update	0	2014-09-16
	-	ELBA-2014-1191	shadow-utils bug fix update	0	2014-09-15
	-	ELBA-2014-1180	openjpeg bug fix update	0	2014-09-11
	-	ELBA-2014-1181	memtest86+ bug fix update	0	2014-09-11
	-	ELBA-2014-1182	compat-db bug fix update	0	2014-09-11
	Important	ELSA-2014-1172	procmail security update	0	2014-09-10
	Important	ELSA-2014-3072	Unbreakable Enterprise kernel security update	0	2014-09-10
	-	ELBA-2014-1164	qemu-kvm bug fix update	0	2014-09-08
	Important	ELSA-2014-1166	jakarta-commons-httpclient security update	0	2014-09-08
	Important	ELSA-2014-1146	httpcomponents-client security update	0	2014-09-03

Legend

Enhancement
Bug fix
Security

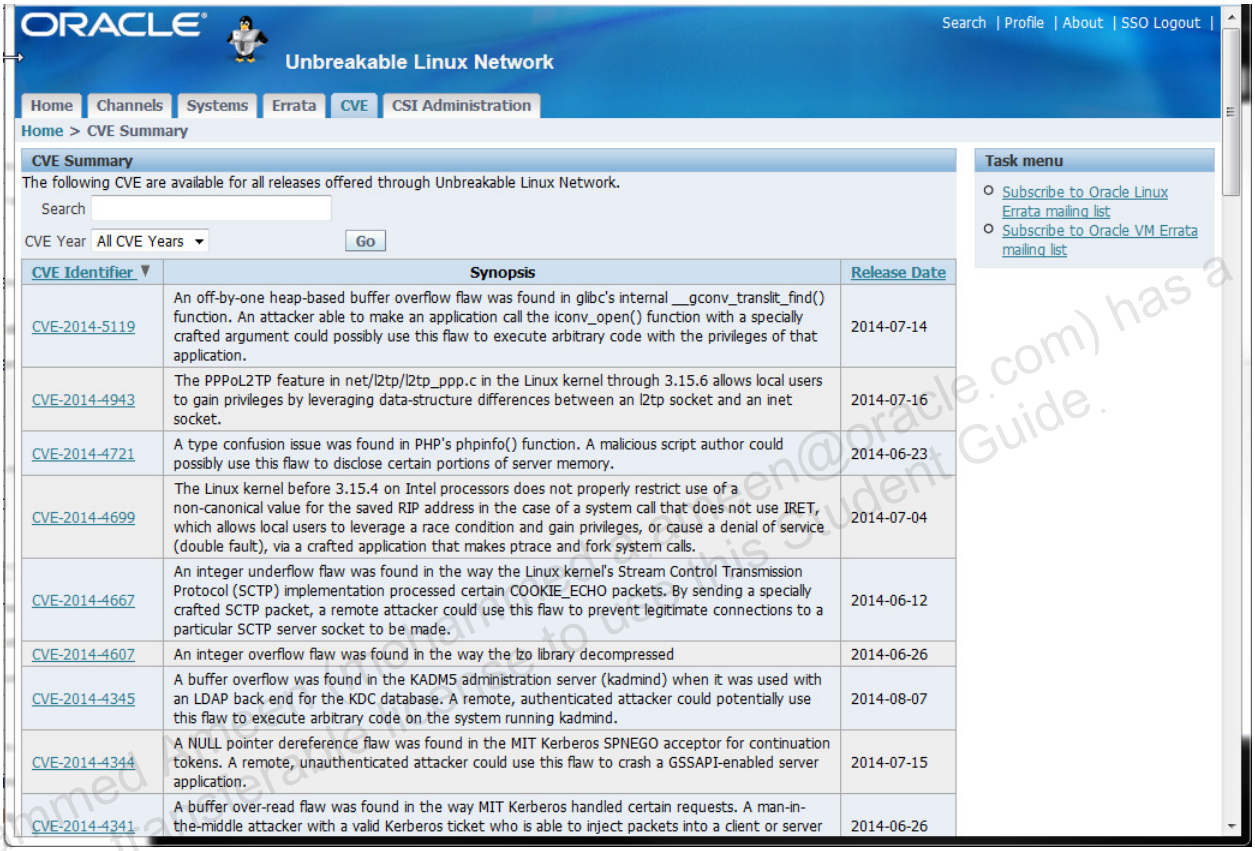
Task menu

Subscribe to Enterprise Linux Errata mailing list
Subscribe to Oracle VM Errata mailing list

- This window has a **Search** feature. You can also filter by **Advisory Type**.
- You can view errata for a particular release of Oracle Linux and Oracle VM.
- You can also subscribe to a mailing list by clicking the links on the right side of the window.

f. Click the **CVE** tab to display the following window.

- CVE stands for Common Vulnerabilities and Exposures.



- You can filter by **CVE Year**, or show all CVE years as shown.
- Click on the **CVE Identifier** link to display detailed information about a specific CVE.

4. Update your system from ULN.

After you have registered your system, you can use the `yum` utility to install the available updates.

a. Run the `yum repolist` command to display the enabled Yum repositories.

```
# yum repolist
Loaded plugins: langpacks, rhnplugin
This system is receiving updates from ULN.
...
repo id            Repo name                                  status
ol7_x86_64_UEKR3   Unbreakable Enterprise Kernel Release 3 ...
ol7_x86_64_ksplice Ksplice for Oracle Linux (x86_64)          ...
ol7_x86_64_latest  Oracle Linux 7 Latest (x86_64)             ...
repolist: ...
```

- Notice the message, “This system is receiving updates from ULN.”

- Notice that the enabled repositories are the ULN channels you are subscribed to.

b. Run the `yum update` command to install the available updates.

```
# yum update
...
Transaction Summary
============================================================
Install    3 Packages
Upgrade  144 Packages

Total download size: 280 M
Is this ok [y/d/N]: y
...
Complete!
```

Practices for Lesson 7: Ksplice

Chapter 7

Practices for Lesson 7: Overview

Practices Overview

In these practices, you do the following:

- Become familiar with the Ksplice Uptrack commands and the Ksplice web interface.
- 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: Using Ksplice Uptrack

Overview

In this practice, you become familiar with installing the Ksplice Uptrack package, using the Ksplice Uptrack commands, and with the Ksplice web interface. Read through the tasks in this practice to help understand the capabilities of Ksplice Uptrack.

Assumptions

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

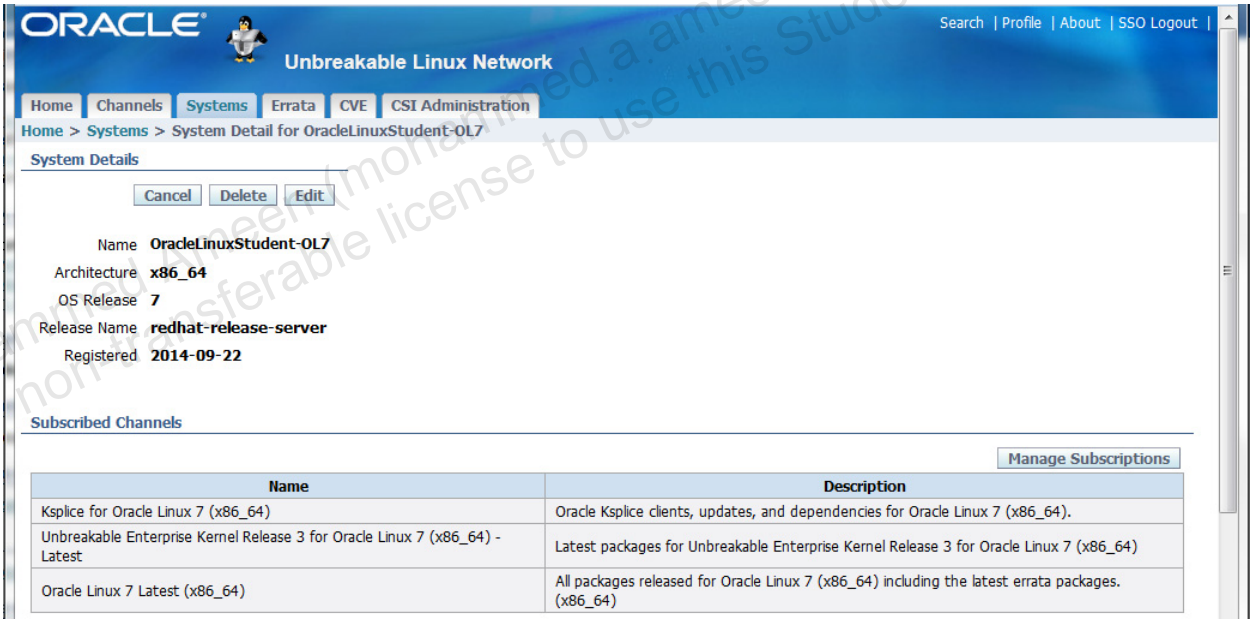
Tasks

1. Review system configuration.
   a. Review system information on ULN.

   The following window shows that the system, **OracleLinuxStudent-OL7**, is registered on ULN.

   It also shows that the system is subscribed to three channels:

   - Ksplice for Oracle Linux 7(x86_64)
   - Unbreakable Enterprise Kernel Release 3 for Oracle Linux 7 (x86_64)
   - Oracle Linux 7 Latest (x86_64)



   b. Run the `yum repolist` command to display the enabled Yum repositories.

```
# yum repolist
Loaded plugins: langpacks, rhnplugin
This system is receiving updates from ULN.
repo id             Repo name                                  status
!ol7_x86_64_UEKR3   Unbreakable Enterprise Kernel Release 3 ...
!ol7_x86_64_ksplice Ksplice for Oracle Linux (x86_64)          ...
!ol7_x86_64_latest  Oracle Linux 7 Latest (x86_64)             ...
```

   - Notice the enabled repositories correspond to the ULN channels.

c. Run the `uname -r` command to display the installed kernel version.

```
# uname -r
3.8.13-35.3.1.el7uek.x86_64
```

- This shows that the UEK version is `3.8.13-35.3.1`.

2. Update the UEK on the system.

a. Run the `yum list available` command. Pipe the output to `grep` and search for the "`kernel-uek`" string.

- This command determines whether an update to the UEK is available.

```
# yum list available | grep kernel-uek
kernel-uek.x86_64               3.8.13-44.1.1.el7uek       ...
kernel-uek-debug.x86_64         3.8.13-44.1.1.el7uek       ...
kernel-uek-debug-devel.x86_64   3.8.13-44.1.1.el7uek       ...
kernel-uek-devel.x86_64         3.8.13-44.1.1.el7uek       ...
kernel-uek-doc.noarch           3.8.13-44.1.1.el7uek       ...
kernel-uek-firmware.x86_64      3.8.13-44.1.1.el7uek       ...
```

- This shows that an update to the UEK is available, version `3.8.13-44.1.1`.

b. Run the `yum update kernel-uek` command to update the UEK.

```
# yum update kernel-uek
...
Dependencies Resolved
================================================================
Package                Arch          Version                ...
================================================================
Installing:
 kernek-uek            x86_64        3.8.13-44.1.1.el7uek   ...
 kernel-uek-firmware   noarch        3.8.13-44.1.1.el7uek   ...

Transaction Summary
================================================================
Install  2 Packages

Total download size: 33 M
Installed size: 112 M
Is this ok [y/d/N]: y
...
Retrieving key from file:///etc/pki/rpm-gpg/RPM-GPG-KEY
...
Is this ok [y/d/N]: y
...
Running transaction
  Installing : kernek-uek-firmware-3.8.13-44.1.1.el7uek.noarch
  Installing : kernek-uek--3.8.13-44.1.1.el7uek.x86_64
```

```
...
Complete!
```

- The installation of the new UEK is successful.

c. Run the `uname -r` command to display the installed kernel version.

```
# uname -r
3.8.13-35.3.1.el7uek.x86_64
```

- Notice the UEK version is still `3.8.13-35.3.1`.
- A reboot is required to install the new UEK.

d. Use the `systemctl reboot` command to reboot the system.

```
# systemctl reboot
```

e. After the reboot completes, log in and become the `root` user.

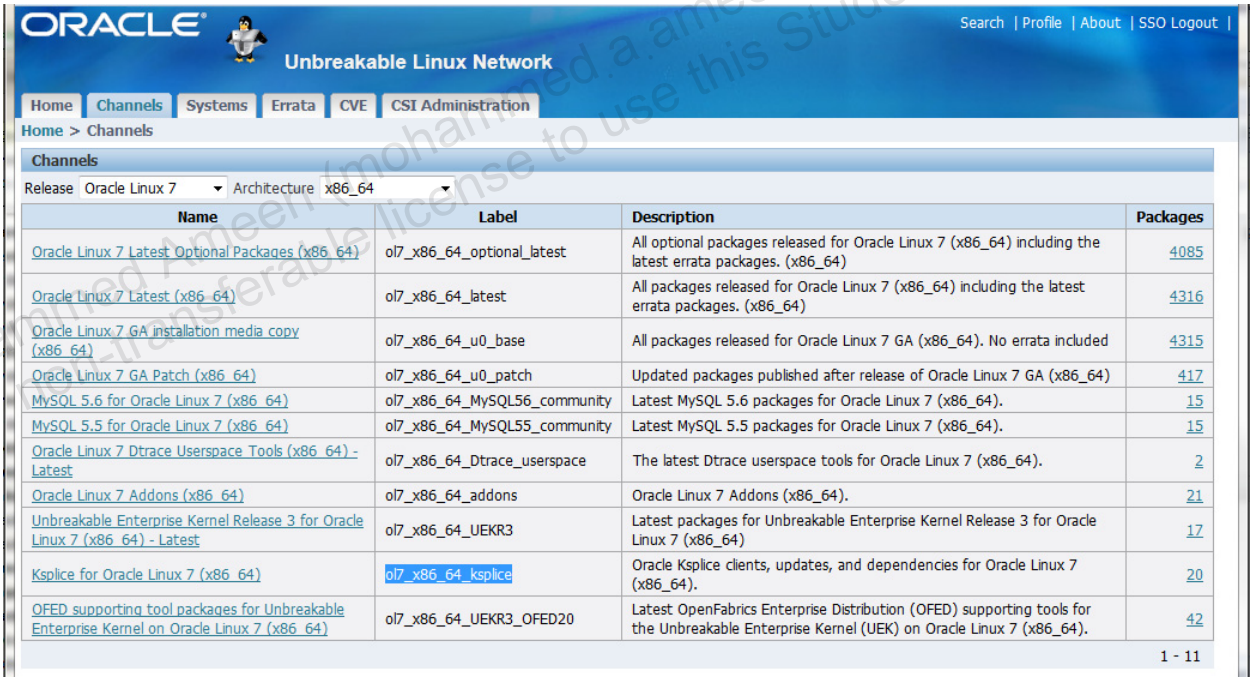
f. Run the `uname -r` command to display the installed kernel version.

```
# uname -r
3.8.13-44.1.1.el7uek.x86_64
```

- Notice that the new UEK version, `3.8.13-44.1.1`, is now running.

3. Browse the **Ksplice for Oracle Linux 7 (x86_64)** channel on ULN.

The following screenshot shows ULN channels for Oracle Linux 7 x86_64 architecture.

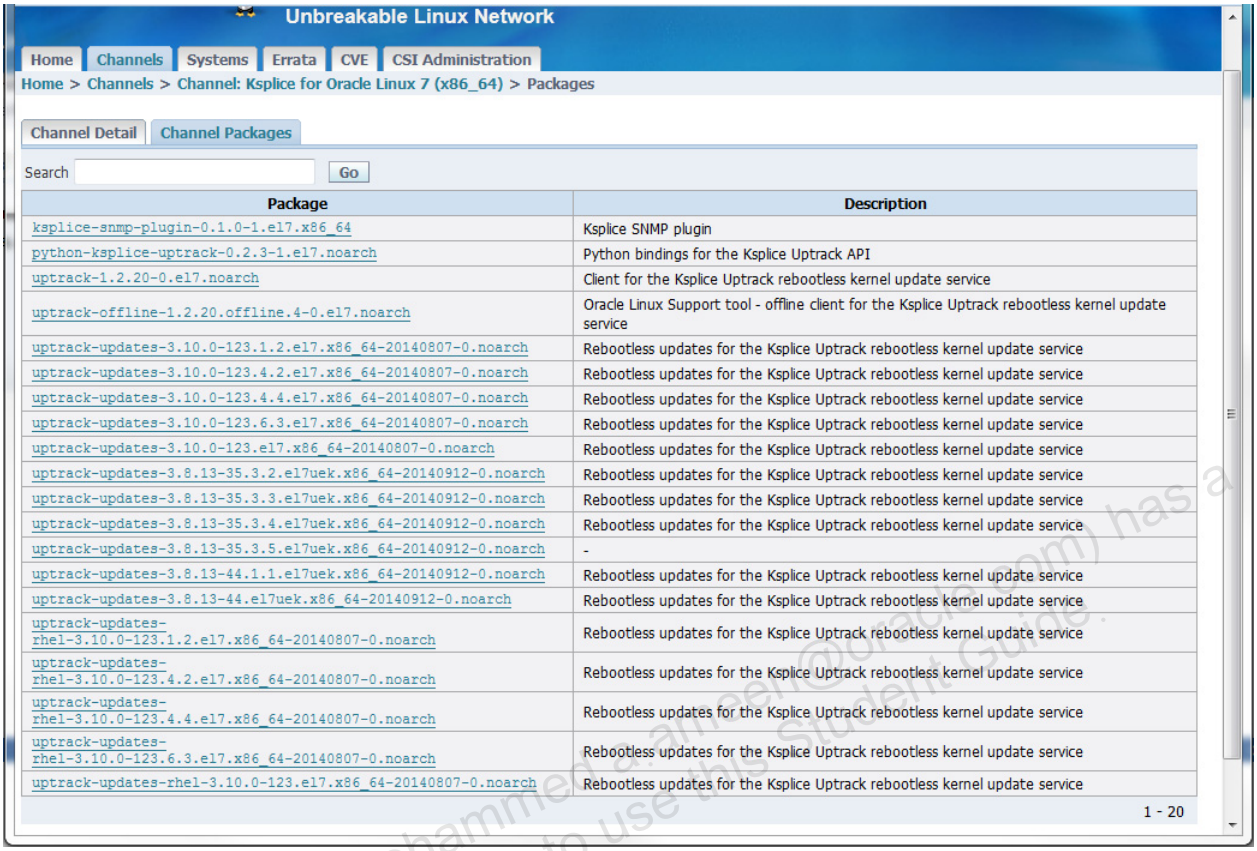


Name	Label	Description	Packages
Oracle Linux 7 Latest Optional Packages (x86_64)	ol7_x86_64_optional_latest	All optional packages released for Oracle Linux 7 (x86_64) including the latest errata packages. (x86_64)	4085
Oracle Linux 7 Latest (x86_64)	ol7_x86_64_latest	All packages released for Oracle Linux 7 (x86_64) including the latest errata packages. (x86_64)	4316
Oracle Linux 7 GA installation media copy (x86_64)	ol7_x86_64_u0_base	All packages released for Oracle Linux 7 GA (x86_64). No errata included	4315
Oracle Linux 7 GA Patch (x86_64)	ol7_x86_64_u0_patch	Updated packages published after release of Oracle Linux 7 GA (x86_64)	417
MySQL 5.6 for Oracle Linux 7 (x86_64)	ol7_x86_64_MySQL56_community	Latest MySQL 5.6 packages for Oracle Linux 7 (x86_64).	15
MySQL 5.5 for Oracle Linux 7 (x86_64)	ol7_x86_64_MySQL55_community	Latest MySQL 5.5 packages for Oracle Linux 7 (x86_64).	15
Oracle Linux 7 Dtrace Userspace Tools (x86_64) - Latest	ol7_x86_64_Dtrace_userspace	The latest Dtrace userspace tools for Oracle Linux 7 (x86_64).	2
Oracle Linux 7 Addons (x86_64)	ol7_x86_64_addons	Oracle Linux 7 Addons (x86_64).	21
Unbreakable Enterprise Kernel Release 3 for Oracle Linux 7 (x86_64) - Latest	ol7_x86_64_UEKR3	Latest packages for Unbreakable Enterprise Kernel Release 3 for Oracle Linux 7 (x86_64)	17
Ksplice for Oracle Linux 7 (x86_64)	ol7_x86_64_ksplice	Oracle Ksplice clients, updates, and dependencies for Oracle Linux 7 (x86_64).	20
OFED supporting tool packages for Unbreakable Enterprise Kernel on Oracle Linux 7 (x86_64)	ol7_x86_64_UEKR3_OFED20	Latest OpenFabrics Enterprise Distribution (OFED) supporting tools for the Unbreakable Enterprise Kernel (UEK) on Oracle Linux 7 (x86_64).	42

1 - 11

- Notice the **Ksplice for Oracle Linux 7 (x86_64)** entry toward the bottom of the screen.
- Also notice that this channel contains **20** packages.

a. Click the **20** packages link to display the following screen:



- The `uptrack` package contains the Ksplice Uptrack utilities.
- Notice there is an "`uptrack-updates`" RPM for the UEK version `3.8.13-44.1.1`. This is the UEK version that is running on the system.

4. Install the `uptrack` package.
   a. Run the `yum install uptrack` command to install the package.

```
# yum install uptrack
...
Dependencies Resolved
================================================================
Package        Arch       Version                ...
================================================================
Installing:
 uptrack       noarch     1.2.20-0.el7    ol7_x86_64_ksplice ...
Installing for dependencies:
 perl-autodie  noarch     2.16-2.el7      ol7_x86_64_latest  ...

Transaction Summary
================================================================
Install  1 Package (+1 Dependent package)
```

```
Total download size: 353 k
Installed size: 961 k
Is this ok [y/d/N]: y
...
Installed:
  uptrack.noarch 0:1.2.20-0.el7

Dependency Installed:
  perl-autodie.noarch 0:2.16-2.el7

Complete!
```

5. Install the Ksplice updates.
   a. Run the `uptrack-show --available` command to see updates available for installation.

```
# uptrack-show --available
Available updates:
[cwy849] CVE-2014-5077: Remote denial-of-service in SCTP on
simultaneous connections.

Effective kernel version is 3.8.13-44.1.1.el7uek
```

   - This shows one update is available.

   b. Run the `uptrack-upgrade -y` command to apply the Ksplice update.

```
# uptrack-upgrade –y
The following steps will be taken:
Install [cwy849] CVE-2014-5077: Remote denial-of-service in SCTP
on simultaneous connections.
Installing [cwy849] CVE-2014-5077: Remote denial-of-service in
SCTP on simultaneous connections.
Your kernel is fully up to date.
Effective kernel version is 3.8.13-44.1.1.el7uek
```

   c. Run the `uptrack-show --available` command to see updates available for installation.

```
# uptrack-show --available
Available updates:
None

Effective kernel version is 3.8.13-44.1.1.el7uek
```

   - Notice there are now no updates available.

   d. Run the `uptrack-show` command to see what updates have been installed.

```
# uptrack-show
Installed updates:
```

```
[cwy849] CVE-2014-5077: Remote denial-of-service in SCTP on
simultaneous connections.


Effective kernel version is 3.8.13-44.1.1.el7uek
```

e. Run the `uname -r` command to display the original kernel version.

```
# uname -r
3.8.13-44.1.1.el7uek
```

- Ksplice Uptrack does not change the output of `uname`.
- The `uname` command displays the version of the kernel into which a machine was booted.

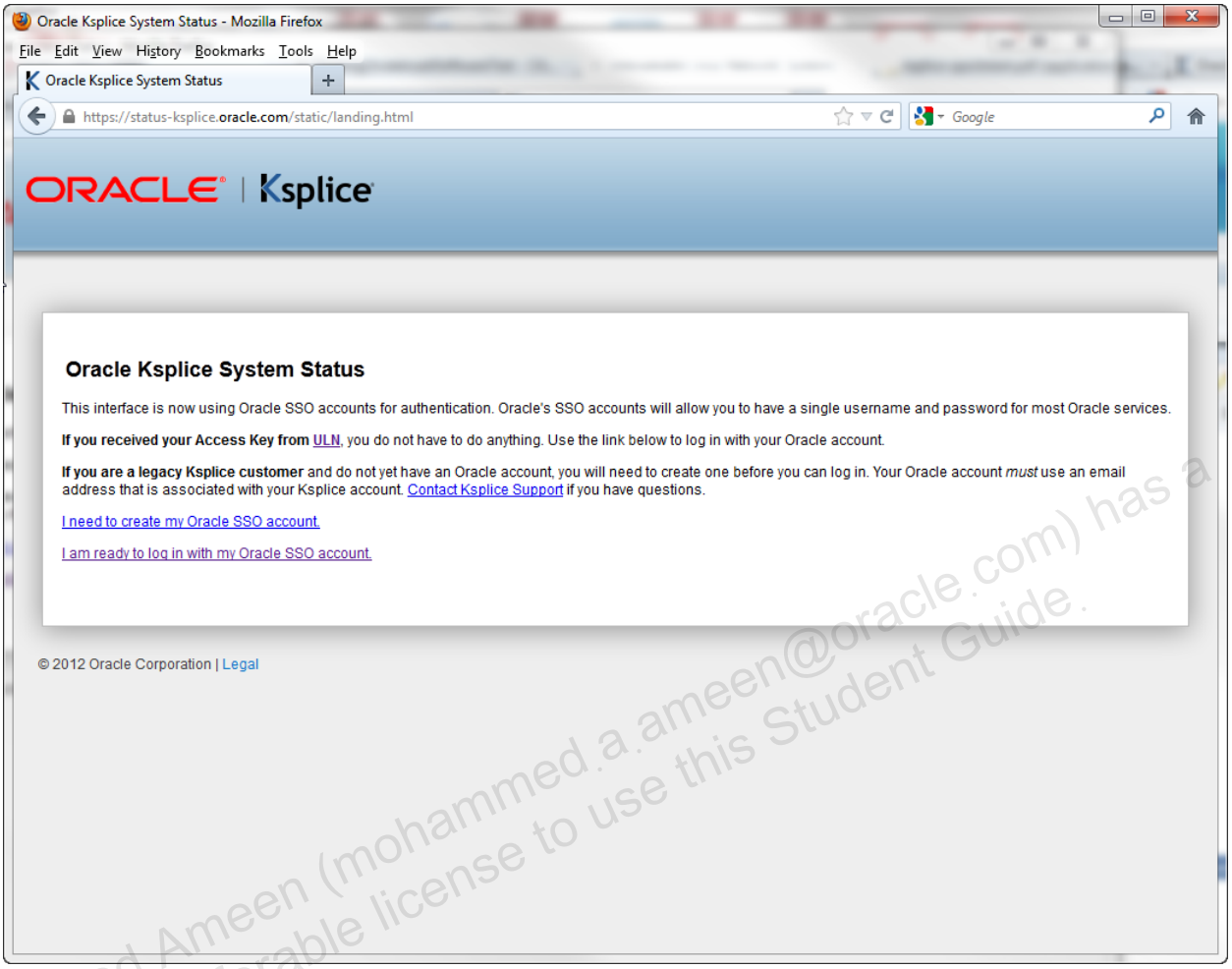
f. Run the `uptrack-uname -r` command to display the effective kernel version.

```
# uptrack-uname -r
3.8.13-44.1.1.el7uek
```

- After installing the updates, `uptrack-uname` reflects the updated running kernel.
- In this example, the original kernel version and the effective kernel version are the same.

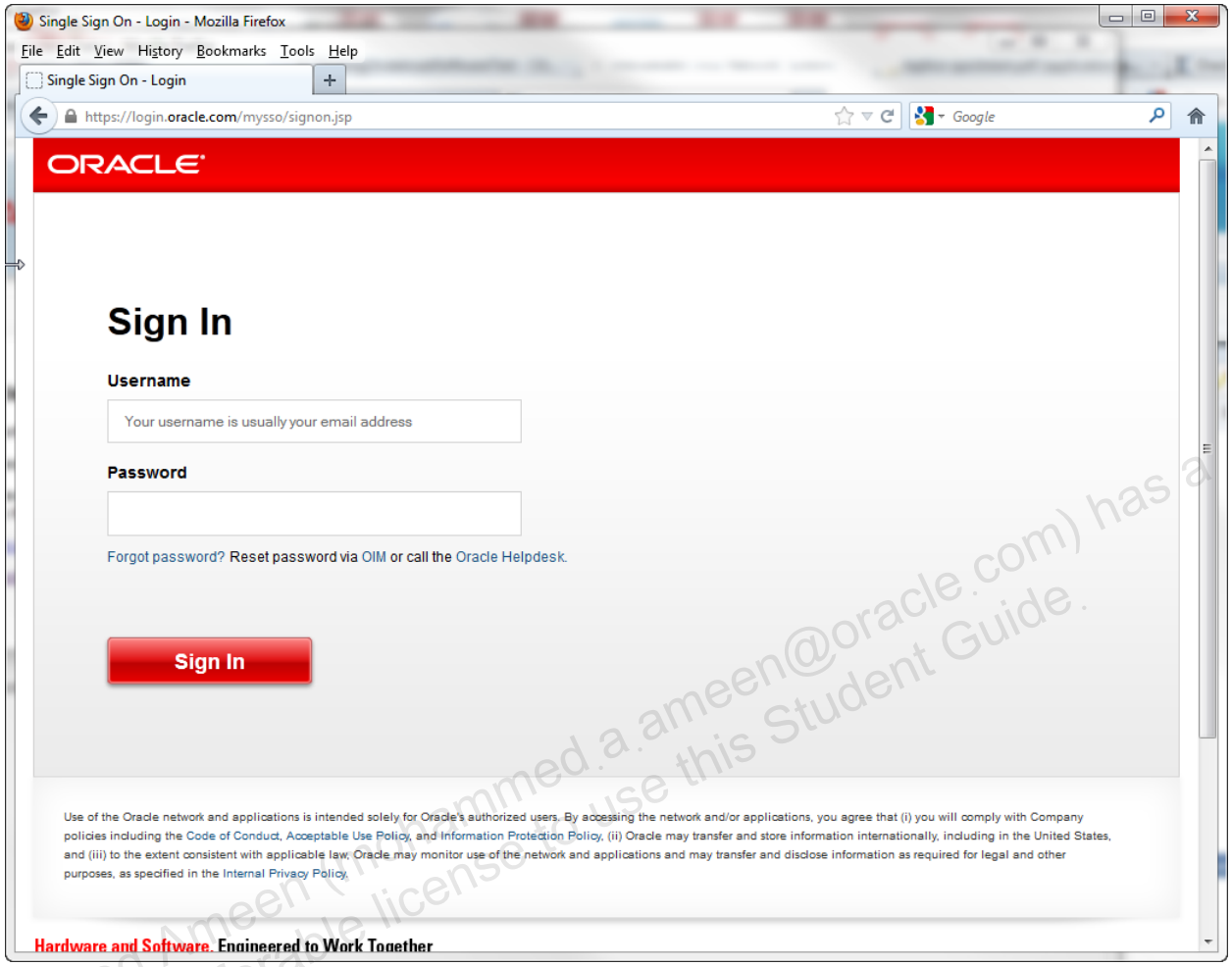
6. Use the Ksplice web interface.
   a. Open a browser and go to https://status-ksplice.oracle.com.

You are prompted to log in.



b. In this example, click **I am ready to log in with my Oracle SSO account**.

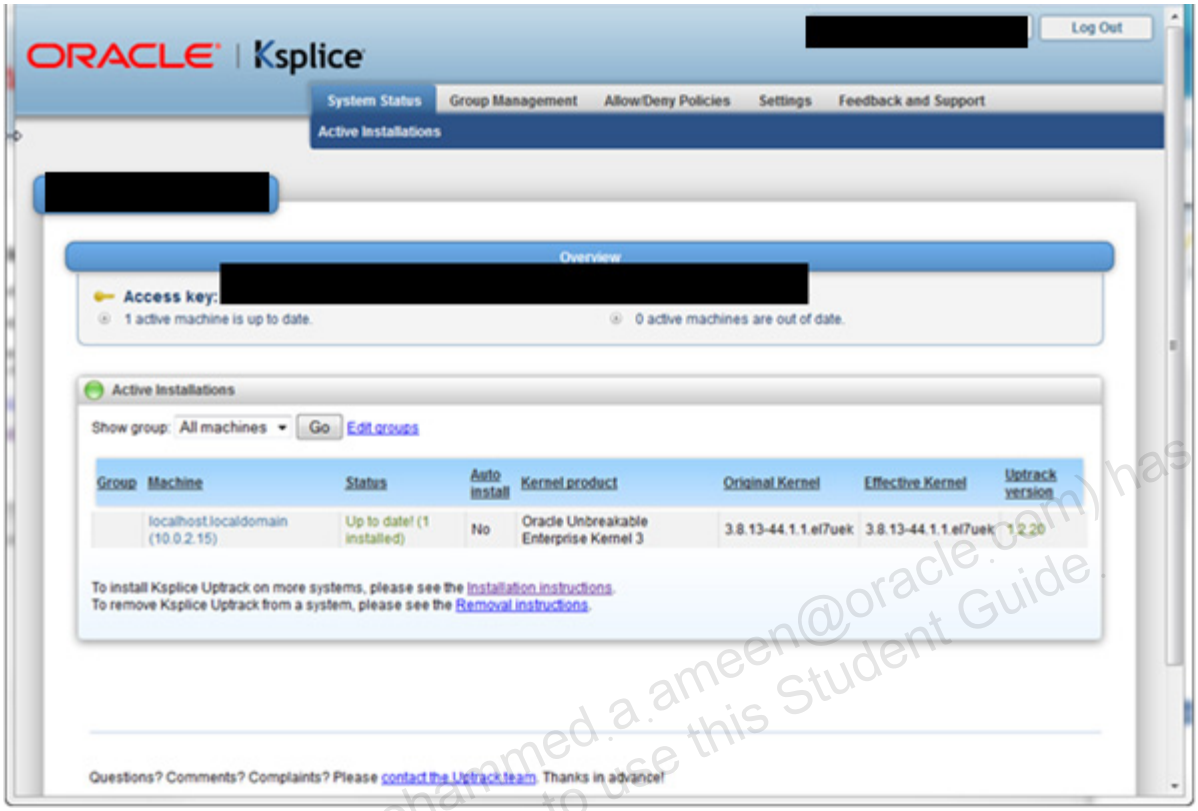
The Oracle Sign In window displays.



c. Provide your login information and click **Sign In**.

The Ksplice web interface window displays.

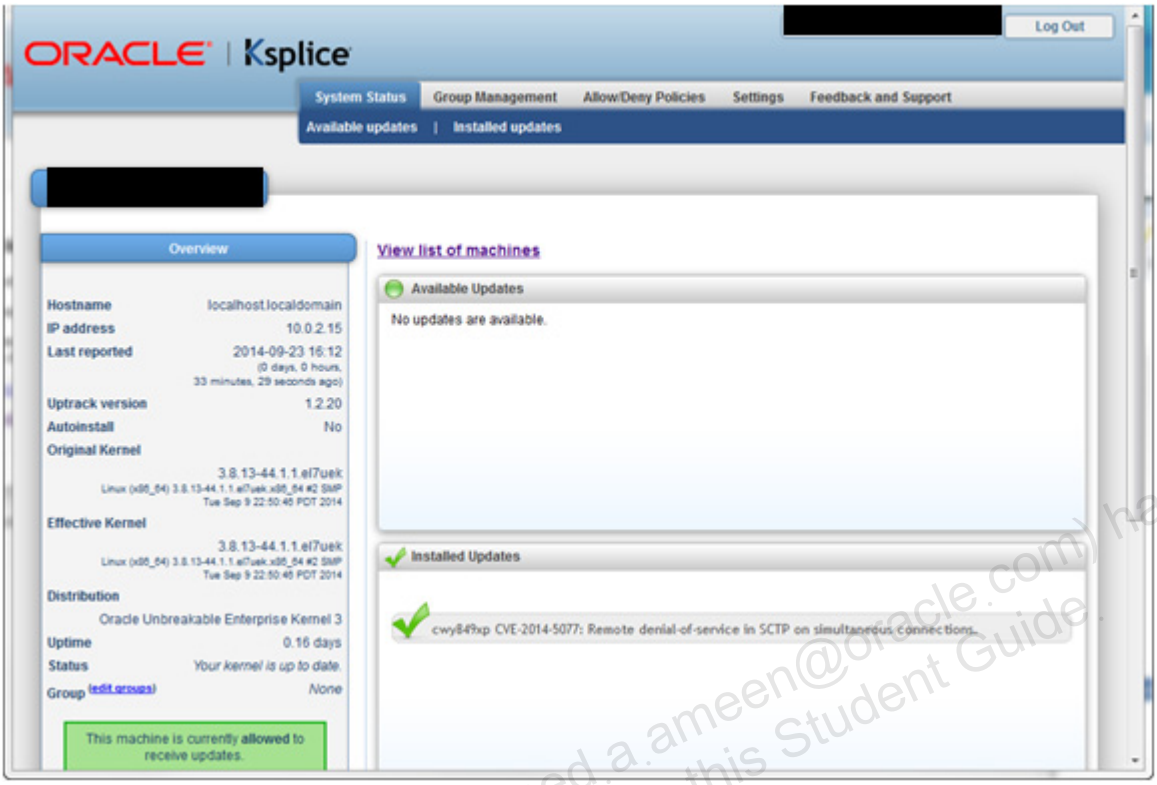
- The **Access key** is blacked-out in this view.



- The window shows one active machine is up-to-date.

d. Click the Machine link, **localhost.localdomain (10.0.2.15)**.

The following window displays.



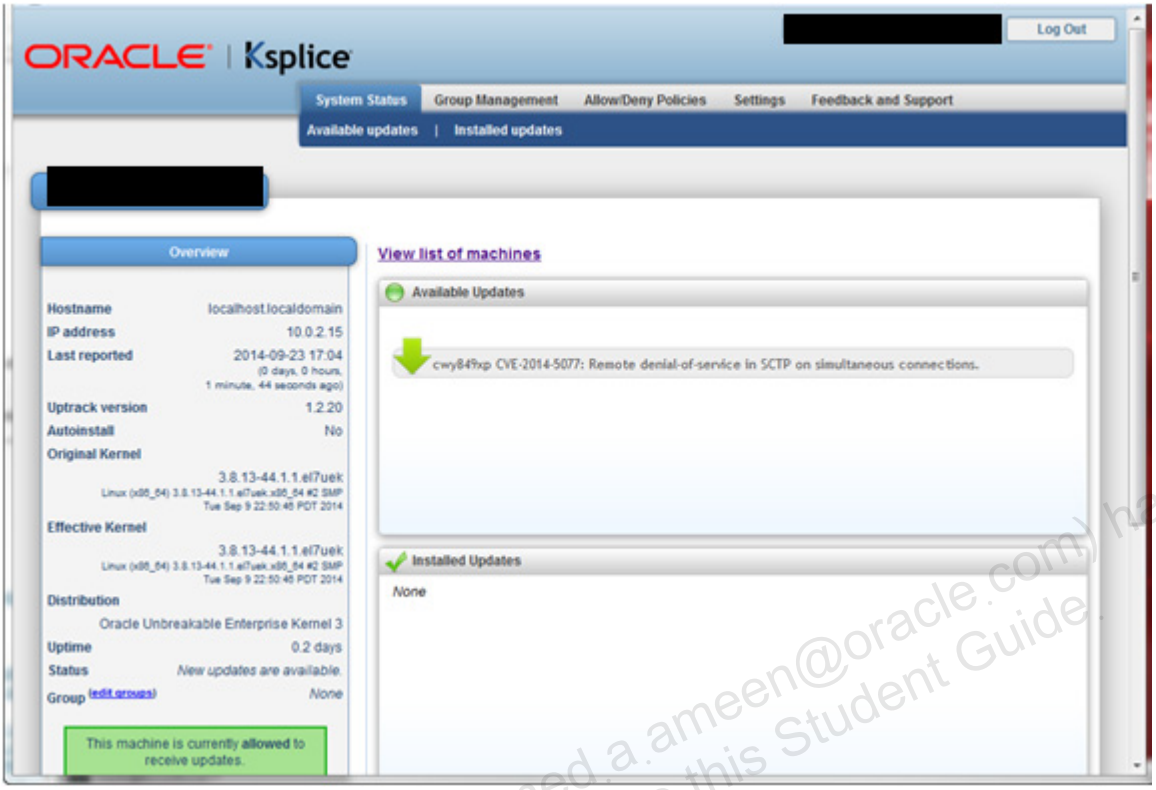
- Note that there are no updates available.
- Note that the installed update is **cwy849 CVE-2014-5077: Remote denial-of-service in SCTP on simultaneous connections**.
- This corresponds to the output of the `uptrack-show` command.

```
# uptrack-show
Installed updates:
[cwy849] CVE-2014-5077: Remote denial-of-service in SCTP on
simultaneous connections.

Effective kernel version is 3.8.13-44.1.1.el7uek
```

7. Remove the Ksplice updates.
   a. Use the `uptrack-remove` command to remove the Ksplice updates.

```
# uptrack-remove
Please specify update IDs to remove or use the --all argument.
# uptrack-remove --all
The following steps will be taken:
Remove [cwy849] CVE-2014-5077: Remote denial-of-service in SCTP
on simultaneous connections.

Go ahead [y/N]? y
Removing [cwy849] CVE-2014-5077: Remote denial-of-service in
SCTP on simultaneous connections.
Effective kernel version is 3.8.13-44.1.1.el7uek
```

The web interface now shows the update is available.



- This corresponds to the output of the `uptrack-show --all` command.

```
# uptrack-show --all
Available updates:
[cwy849] CVE-2014-5077: Remote denial-of-service in SCTP on
simultaneous connections.
Installed updates:
None

Effective kernel version is 3.8.13-44.1.1.el7uek
```

Practice 7-2: Installing the Ksplice Offline Client and Kernel Updates

Overview

In this practice, you perform the following:

- View staged software packages on **dom0**.
- View the `167283.sh` file.
- Use `sftp` to upload the staged software packages from **dom0** to the **host03** VM.
- Install a new UEK on **host03**.
- 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 **dom0**.
- You are the `root` user on **host03**.
- Software packages have been downloaded from ULN and staged on **dom0**.

Tasks

1. View staged packages on **dom0**.
   a. As the `root` user on **dom0**, use the `cd` command to change to the `/OVS/seed_pool/ksplice` directory.

```
[dom0]# cd /OVS/seed_pool/ksplice
```

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

```
[dom0]# ls
167283.sh
kernel-uek-3.8.13-44.1.1.el7uek.x86_64.rpm
kernel-uek-firmware-3.8.13-44.1.1.el7uek.noarch.rpm
perl-autodie-2.16-2.el7.noarch.rpm
uptrack-1.2.20-0.el7.noarch.rpm
uptrack-offline-1.2.20.offline.4-0.el7.noarch.rpm
uptrack-updates-3.8.13-44.1.1.el7uek.x86_64-20140912-
0.noarch.rpm
```

   - All files in this directory, with the exception of the `167283.sh` file, have been downloaded from ULN.
   - The `uptrack-offline` package is provided.
   - The UEK version `3.8.13-44.1.1` is provided.
   - The package containing kernel updates for the UEK version `3.8.13-44.1.1` (`uptrack-updates-3.8.13-44.1.1.el7uek.x86_64-20140912-0.noarch.rpm`) is also provided.

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

- The `167283.sh` file has been downloaded from http://www.oracle.com/ocom/groups/public/@otn/documents/webcontent/167283.sh.
- The `167283.sh` file is not required for this practice. It is required to create only the Yum repositories for the registered channels.
- This purpose of this step is just to familiarize you with the contents of this `167283.sh` script.
- 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 labels ...
...
# 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
...
```

2. Use the `sftp` command to transfer the Ksplice packages from **dom0** to **host03**.

   The `sftp` command is discussed in the lesson titled "OpenSSH Service."

   a. From the `/OVS/seed_pool/ksplice` directory on **dom0**, use the `sftp host03` command to connect to **host03** as `root`.

   - Provide `root` as the password for the username `oracle` when prompted.

```
[dom0]# sftp host03
Connecting to 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-3.8.13-44.1.1.el7uek.x86_64-20140912-
0.noarch.rpm to /root/uptrack-updates-3.8.13-
44.1.1.el7uek.x86_64-20140912-0.noarch.rpm
...
sftp>
```

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

```
sftp> quit
```

3. Install a new UEK on **host03**.

a. From **host03**, run the `uname -r` command to display the version of the running kernel.

```
# uname -r
3.8.13-35.3.1.el7uek.x86_64
```

b. Use the `cd` command to ensure you are in the root user's home directory. Use the `ls` command to display the contents of the directory.

- Not all files in the directory are listed in the following.

```
# cd
# ls
167283.sh
anaconda-ks.cfg
initial-setup-ks.cfg
kernel-uek-3.8.13-44.1.1.el7uek.x86_64.rpm
kernel-uek-firmware-3.8.13-44.1.1.el7uek.noarch.rpm
perl-autodie-2.16-2.el7.noarch.rpm
uptrack-1.2.20-0.el7.noarch.rpm
uptrack-offline-1.2.20.offline.4-0.el7.noarch.rpm
uptrack-updates-3.8.13-44.1.1.el7uek.x86_64-20140912-
0.noarch.rpm
```

- The `anaconda-ks.cfg` file and the `initial-setup-ks.cfg` file were created during the installation of Oracle Linux.
- The remaining files in this directory were transferred from **dom0** by using the `sftp` command.
- Note that the package containing kernel updates is for the UEK version `3.8.13-44.1.1` (`uptrack-updates-3.8.13-44.1.1.el7uek.x86_64-20140912-0.noarch.rpm`).
- From task 3a, the version of the running kernel is `3.8.13-35.3.1`.
- Therefore, you need to install the `3.8.13-44.1.1` kernel before you can apply the Ksplice updates.

c. Before installing the new kernel, use the `grep` command to list the bootable kernels in `/boot/grub2/grub.cfg`.

```
# grep '^menuentry' /boot/grub2/grub.cfg
menuentry 'Oracle ... with Linux 3.10.0-123.el7.x86_64... {
```

```
menuentry 'Oracle ... with ... 3.8.13-35.3.1.el7uek... {
menuentry 'Oracle ... with Linux 0-rescue-... {
```

- Note that there are three entries:
  - RHCK version `3.10.0-123.el7`
  - UEK version 3.8.13-35.3.1.el7uek
  - Rescue kernel

d. Use the `rpm` command to install the `3.8.13-44.1.1` kernel packages.

- The `kernel-uek` packages are provided in the current directory.
- You need to install the `kernel-uek-firmware` file first because it is a dependency.
- When installing a new kernel with the `rpm` command, use the `-i` option and not the `-U` option so as not to overwrite the existing kernel.

```
# rpm -ivh kernel-uek-3.8.13-44.1.1.el7uek.x86_64.rpm
error: Failed dependencies:
        kernel-firmware = 3.8.13-44.1.1.el7uek is needed ...
# rpm -ivh kernel-uek-firmware-3.8.13-44.1.1.el7uek.x86_64.rpm
Preparing...
Updating / installing...
   1:kernel-uek-firmware-3.8.13-44.1.1################## [100%]
```

e. After installing the `kernel-uek-firmware` package, you can now install the `kernel-uek` package.

```
# rpm -ivh kernel-uek-3.8.13-44.1.1.el7uek.x86_64.rpm
Preparing...
Updating / installing...
   1:kernel-uek-3.8.13-44.1.1.el7uek ################## [100%]
```

f. Run the `uname -r` command to display the version of the running kernel.

```
# uname -r
3.8.13-35.3.1.el7uek.x86_64
```

- Note that the UEK version is still `3.8.13-35.3.1`.
- A reboot is required to install the new UEK.

g. Before rebooting, use the `grep` command to list the bootable kernels in `/boot/grub2/grub.cfg`.

```
# grep '^menuentry' /boot/grub2/grub.cfg
menuentry 'Oracle ... with ... 3.8.13-44.1.1.el7uek... {
menuentry 'Oracle ... with Linux 3.10.0-123.el7.x86_64... {
menuentry 'Oracle ... with ... 3.8.13-35.3.1.el7uek... {
menuentry 'Oracle ... with Linux 0-rescue-... {
```

- Now there is a new entry (listed first) corresponding to the newly installed UEK, version `3.8.13-44.1.1.el7uek`.
- Installing a new kernel automatically updates the `/boot/grub/grub.cfg` file.

h. Use the `systemctl reboot` command to reboot the system.

- It might take a couple minutes to reboot.

```
# systemctl reboot
```

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

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

j. Run the `vncviewer&` command.

```
# vncviewer&
```

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

k. Enter the `localhost:<port_number>` command, substituting the correct port number for the **host03** guest. For example, if the port number is 5904, enter the following and click **OK**:

```
localhost:5904
```

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

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

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

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

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

```
# uname -r
3.8.13-44.1.1.el7uek.x86_64
```

- Note that the new UEK version, `3.8.13-44.1.1`, is now running.

4. Install the Ksplice Uptrack kernel update.

a. Use the `cd` command to ensure you are in the root user's home directory. Use the `ls` command to display the contents of the directory.

- Not all files in the directory are listed in the following.

```
# cd
# ls
167283.sh
anaconda-ks.cfg
initial-setup-ks.cfg
kernel-uek-3.8.13-44.1.1.el7uek.x86_64.rpm
kernel-uek-firmware-3.8.13-44.1.1.el7uek.noarch.rpm
perl-autodie-2.16-2.el7.noarch.rpm
uptrack-1.2.20-0.el7.noarch.rpm
uptrack-offline-1.2.20.offline.4-0.el7.noarch.rpm
uptrack-updates-3.8.13-44.1.1.el7uek.x86_64-20140912-
0.noarch.rpm
```

b. Use the `rpm` command to install the Ksplice Offline Client package.

- The `uptrack-offline` package is the Ksplice Offline Client.

Note: The `uptrack-offline` package will not install if the `uptrack-1.2.20-0.el7.noarch.rpm` package is already installed.

```
# rpm -Uvh uptrack-offline-1.2.20-0.offline.4-0.el7.noarch.rpm
error: Failed dependencies:
        perl(Fatal) is needed by uptrack-offline-1.2.20-
0.offline.4-0.el7.noarch
```

- Note that the install fails due to a dependency.

c. Use the `rpm` command to install the `perl-autodie` package.

```
# rpm -Uvh perl-autodie-2.16-2.el7.noarch.rpm
Preparing...
Updating / installing...
   1: perl-autodie-2.16-2.el7  ########################### [100%]
```

d. Repeat step 4b: use the `rpm` command to install the Ksplice Offline Client package.

```
# rpm -Uvh uptrack-offline-1.2.20-0.offline.4-0.el7.noarch.rpm
Preparing...
Updating / installing...
   1: uptrack-offline-1.2.20-0.offline.4################ [100%]
There are no existing modules on disk that need basename
migration.
```

- The Ksplice Offline Client package is installed successfully.

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

```
# rpm -Uvh uptrack-updates-3.8.13-44.1.1.el7uek.x86_64-20140912-
0.noarch.rpm
Preparing...
Updating / installing...
   1:uptrack-updates-3.8.13-############################## [100%]
The following steps will be taken:
Install [cwy849] CVE-2014-5077: Remote denial-of-service in SCTP
on simultaneous connections.
Installing [cwy849] CVE-2014-5077: Remote denial-of-service in
SCTP on simultaneous connections.
Your kernel is fully up to date.
Effective kernel version is 3.8.13-44.1.1.el7uek
```

- All kernel updates are installed.

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

```
# uname -r
3.8.13-44.1.1.el7uek.x86_64
```

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

```
# uptrack-uname -r
3.8.13-44.1.1.el7uek.x86_64
```

- In this example, the effective kernel version is the same as the original kernel.

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

```
# uptrack-show
Installed updates:
[cwy849] CVE-2014-5077: Remote denial-of-service in SCTP on
simultaneous connections.

Effective kernel version is 3.8.13-44.1.1.el7uek
```

Practices for Lesson 8: Automating Tasks

Chapter 8

Practices for Lesson 8: Overview

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
```

   - Note 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 are 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.
   - Sample output is displayed.

```
# mail
...
>N 1 (Cron Daemon) <date_time>  33/1129  "Cron <root@host03> ls"
&
```

- To view the 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` command to remove `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
```

- Note 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. Right-click and select **Open Tab** from the pop-up menu to open another tab in your terminal window.

      - 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.

      - Sample output is displayed.

- You might need to wait a few seconds before you have any mail.

```
[oracle@host03 ~]$ mail
...
>N 1 (Cron Daemon)  <date_time>  25/864  "Cron <oracle@host03> "
&
```

g. To view the details of mailbox entries, press the associated number and then press **Enter**.

h. After viewing the details, press **q** and then press **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
```

- Note 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.

- Press **Ctrl + D** for <EOT>.

```
[oracle@host03 ~]$ at now + 1 minute
at> pwd
at> <EOT>
job 1 at <date_time>
```

c. Use the `atq` command to list the pending job.

```
[oracle@host03 ~]$ atq
1     <date_time> a oracle
```

d. After 1 minute has passed, run the `atq` command again.

```
[oracle@host03 ~]$ atq
```

- Note 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 <date_time> 14/518  "Output from your job"
&
```

f. To view the details of mailbox entries, press the associated number and then press **Enter**.

g. After viewing the details, press **q** and then press **Enter** to quit.

```
& 2
...
/home/oracle
& q
```

4. Restrict the 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 the `oracle` user 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: Overview

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 7-2.
      – These Ksplice modules are not included in the sample output.

```
# lsmod
Module                  Size  Used by
tcp_lp                 2415  0
nls_utf8               1421  1
bnep                  15888  2
bluetooth            350532  7 bnep
...
llc                    5472  2 stp,bridge
...
```

      – 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, `llc` depends on `stp` and `bridge` modules being loaded before `llc` 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-
44.1.1.el7uek.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
3.8.13-44.1.1.el7uek.x86_64
```

d. List the kernel modules for the kernel version.

```
# ls -R /lib/modules/`uname -r`/kernel
arch
crypto
drivers
fs
kernel
lib
mm
net
sound
...
/lib/modules/3.8.13-44.1.1.el7uek.x86_64/kernel/arch:
x86

/lib/modules/3.8.13-44.1.1.el7uek.x86_64/kernel/arch/x86:
crypto
kernel
kvm
oprofile

/lib/modules/3.8.13-44.1.1.el7uek.x86_64/kernel/arch/x86/crypto:
ablk_helper.ko
aesni-intel.ko
...
```

– The actual kernel modules have a `.ko` (kernel object) extension.

2. Load kernel modules.

a. Determine whether the `nfs` kernel module is currently loaded.

```
# lsmod | grep nfs
nfsd          268587  1
auth_rpcgss    38456  1  nfsd
nfs_acl         2653  1  nfsd
lockd          83776  1  nfsd
sunrpc        262929  5  nfsd,auth_rpcgss,lockd,nfs_acl
```

– In this example, `nfsd` is loaded but the `nfs` module 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-
44.1.1.el7uek.x86_64/kernel/fs/fscache/fscache.ko
insmod /lib/modules/3.8.13-
44.1.1.el7uek.x86_64/kernel/fs/nfs/nfs.ko
```

- The preceding example includes the `-v` (verbose) option.
- Note that a dependent module, `fscache`, is loaded before loading the `nfs` module.
- Also note 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                178150      0
fscache             51498      1   nfs
...
sunrpc             262929      6   nfs,...
```

- Note that the `nfs` module is loaded, as well as the dependent module, `fscache`.

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`.

- Include the `-w` option for `grep` to search for the whole word "`nfs`".

```
# grep -w nfs.ko /lib/modules/`uname -r`/modules.dep
kernel/fs/nfs/nfs.ko: kernel/fs/lockd/lockd.ko
kernel/net/sunrpc/sunrpc.ko kernel/fs/fscache/fscache.ko
...
```

- Note that the dependencies listed in the `modules.dep` file for `nfs` correspond with the modules loaded when running the `modprobe nfs` command.
- The `sunrpc` module did not load when loading `nfs` because it is a dependency for another module and was already loaded.

4. Unload kernel modules.

Use the `modprobe -r` command to unload the `nfs` kernel module.

- Do not unload or experiment with any other modules because you could potentially damage the operating system or leave the virtual machine in an unusable state.

```
# modprobe -rv nfs
rmmod nfs.ko
rmmod fscache.ko
```

- The preceding example includes the `-v` (verbose) option.
- Note that `modprobe -r` uses the `rmmod` command to remove the modules.
- Also note 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
```

   – Note that the `.modules` files are executable shell scripts.

Practices for Lesson 10: User and Group Administration

Chapter 10

Practices for Lesson 10: Overview

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: Administering User Accounts

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 the `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:1000:1000:Oracle Student:/home/oracle:/bin/bash
student1:x:1001:1001::/home/student1:/bin/bash
```

   - Note that the new user's UID and GID incremented by one.
   - Note that a home directory was created for the new user (`/home/student1`).
   - Note that the default shell for the new user is `/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
```

   - Note 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.  .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.  .mozilla
```

- Note 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:1000:oracle
student1:x:1001:
```

- Because Oracle Linux uses a user private group (UPG) scheme, a new private group (`student1`, `GID=1001`) was created when the user `student1` user was created.

i. View the new `student1` entry in the `/etc/shadow` file.

```
# cat /etc/shadow
...
oracle:$6$...:16331:0:99999:7:::
student1:!!:16337:0:99999:7:::
```

j. View the new `student1` entry in the `/etc/gshadow` file.

```
# cat /etc/gshadow
...
oracle:!!::oracle
student1:!::
```

k. Add a new user with the following characteristics:

- Username=`student2`
- UID=`1055`
- GECOS information="`Oracle Student2`"
- Default shell=`/bin/sh` (Bourne shell)

```
# useradd -u 1055 -c "Oracle Student2" -s /bin/sh student2
```

l. View the new `student2` entry in `/etc/passwd`.

```
# tail -2 /etc/passwd
student1:x:1001:1001::/home/student1:/bin/bash
student2:x:1055:1055:Oracle Student2:/home/student2:/bin/sh
```

- Note that the UID and GID are `1055`.

- Note the GECOS information.
- Note 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:1001:1001: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:$6$...:16331:0:99999:7:::
student1:!!:16337:0:99999:7:::
student2:!!:16337:0:99999:7:::
```

   - Note that the "`!!`" in the `student1` and `student2` records, indicate 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.
   - Ignore the "BAD PASSWORD" warning, continuing to use `password` as the password.

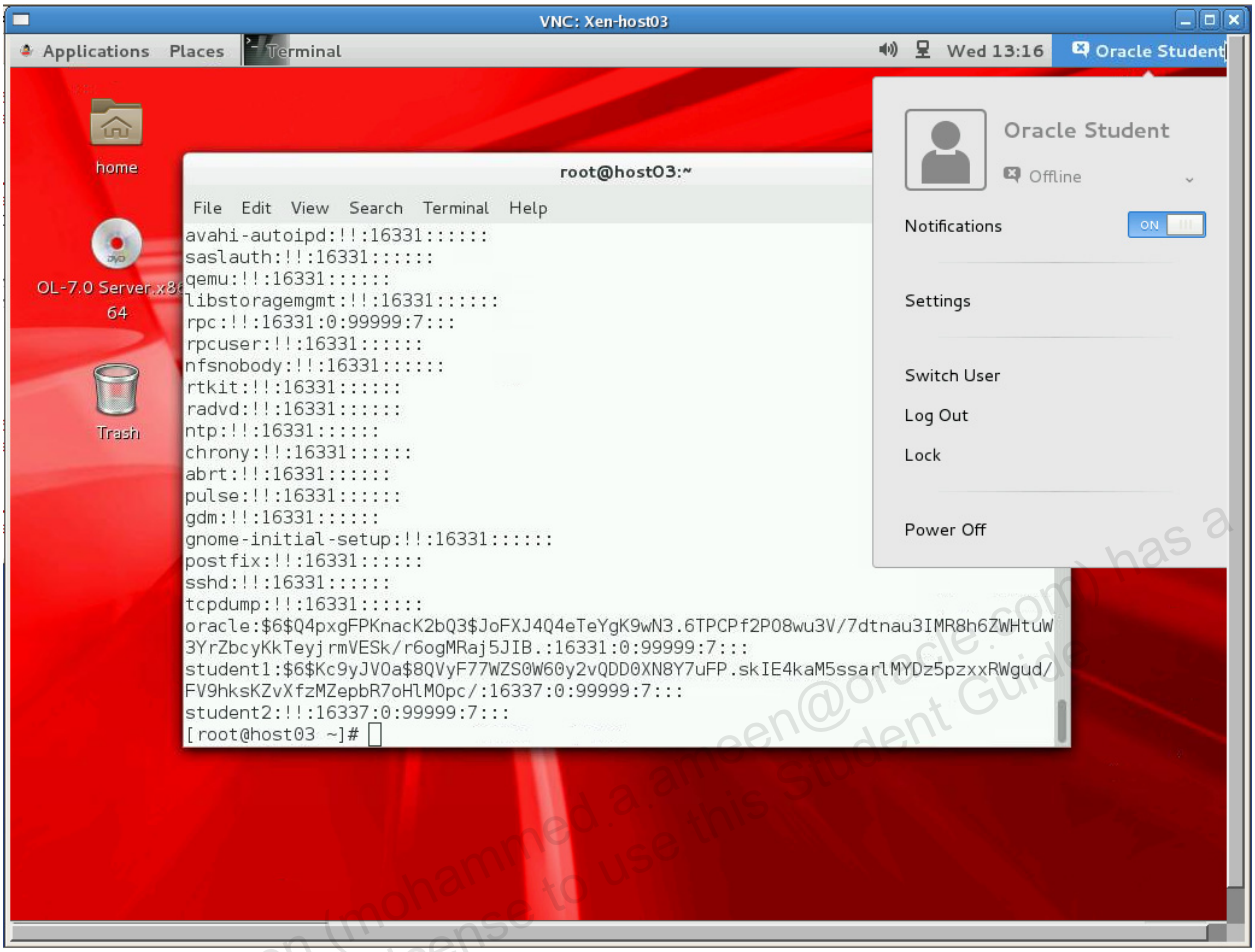
```
# passwd student1
Changing password for user student1.
New password: password
BAD PASSWORD: The password fails the dictionary check - it is
based on a dictionary word
Retype new password: password
passwd: all authentication tokens updated successfully.
```

   c. View the `/etc/shadow` file.

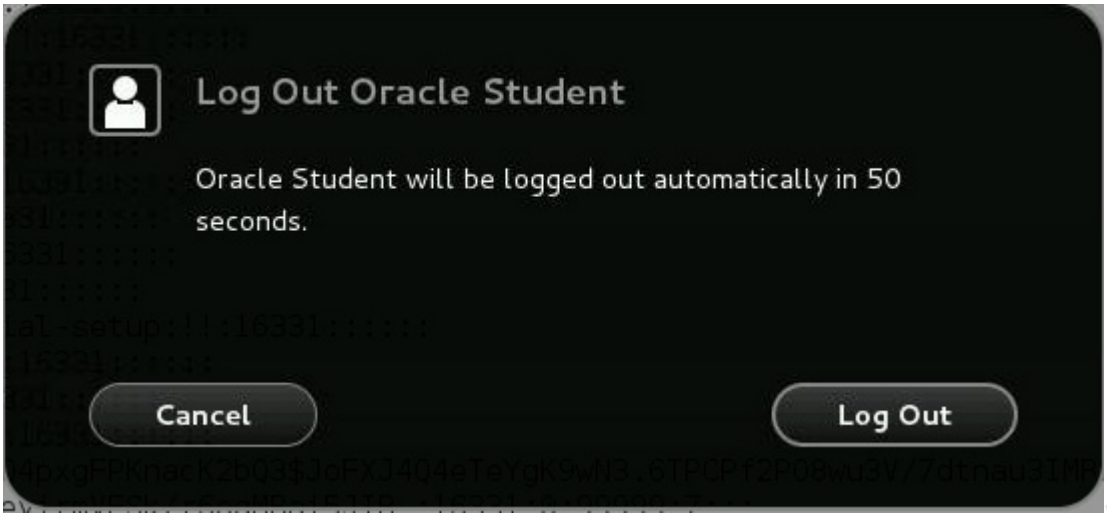
```
# cat /etc/shadow
...
oracle:$6$...:16331:0:99999:7:::
student1:$6$...: 16337:0:99999:7:::
student2:!!: 16337:0:99999:7:::
```

   - Now the "`!!`" for `student1` has been replaced with a hashed password value.
   - Note that the `student2` account is still locked because a password has not been assigned.

4. Log in as new user.
   a. Log out as the `oracle` user by clicking **Oracle Student** and selecting **Log Out** from the pop-up menu, as shown in the following.

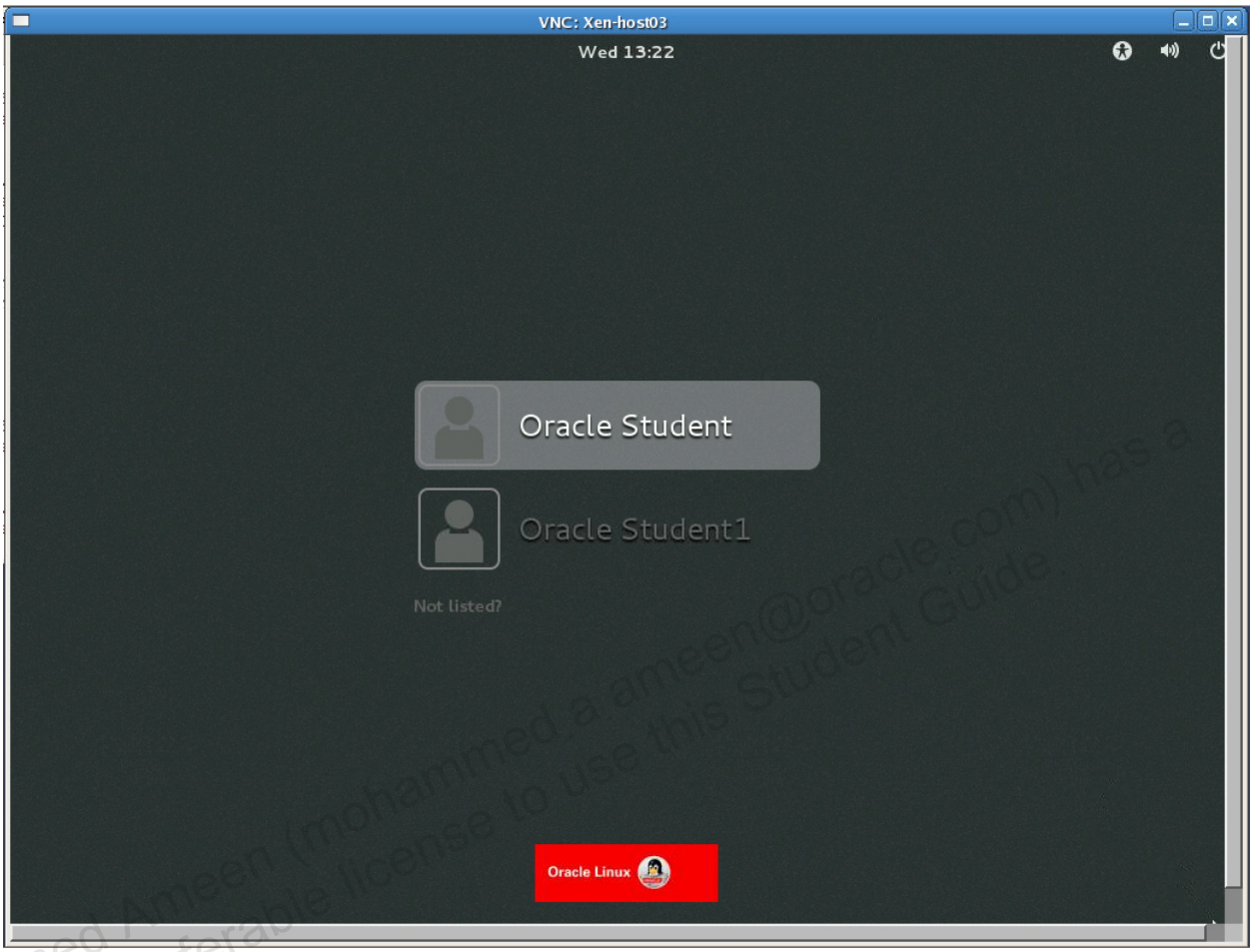


b. Click **Log Out** in the pop-up, as shown in the following.



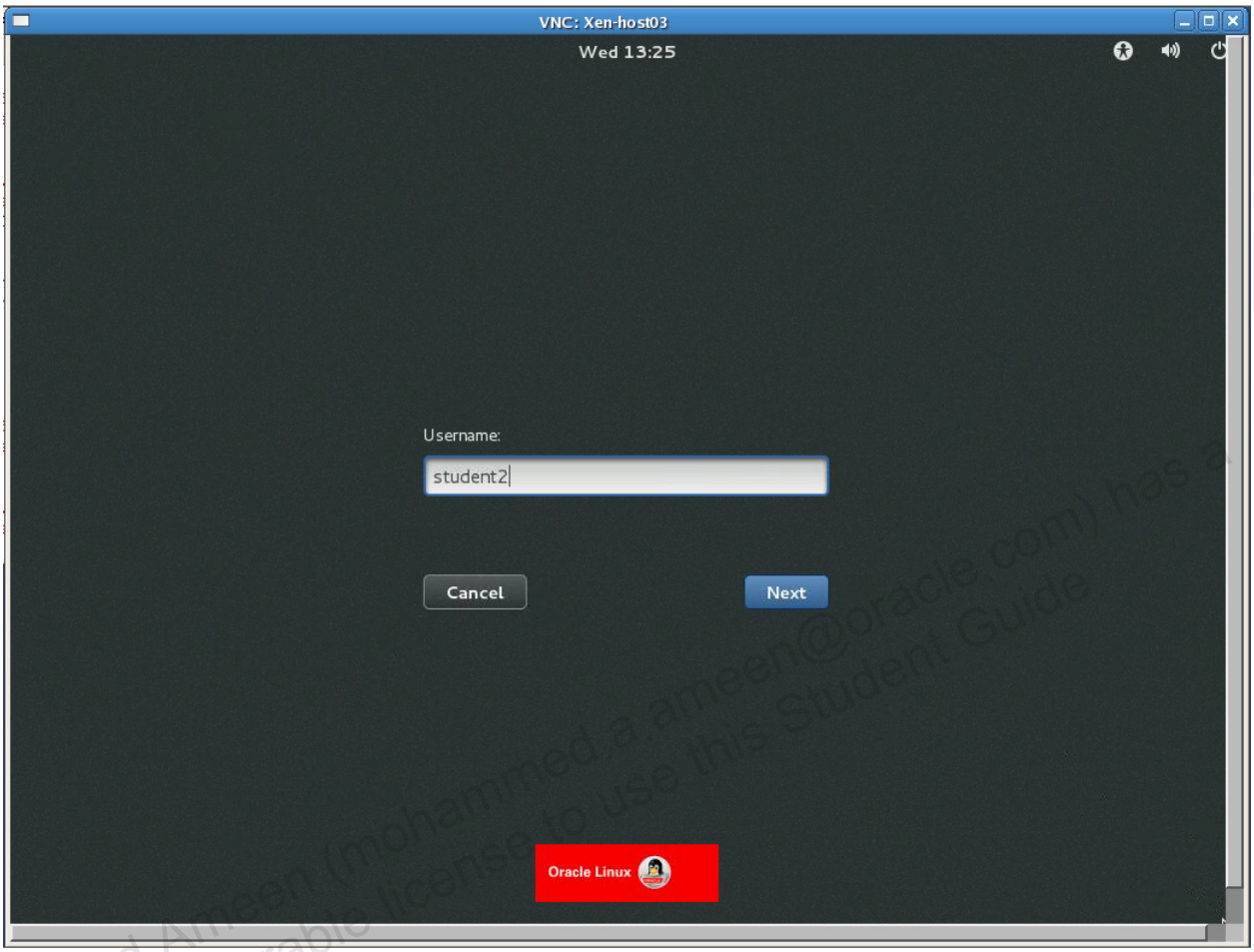
c. Attempt to log in as `student2` (**Oracle Student2**).

- Click **Not listed?** from the following window:



- You are presented with a **Username:** prompt.

d. Enter `student2` shown as follows:



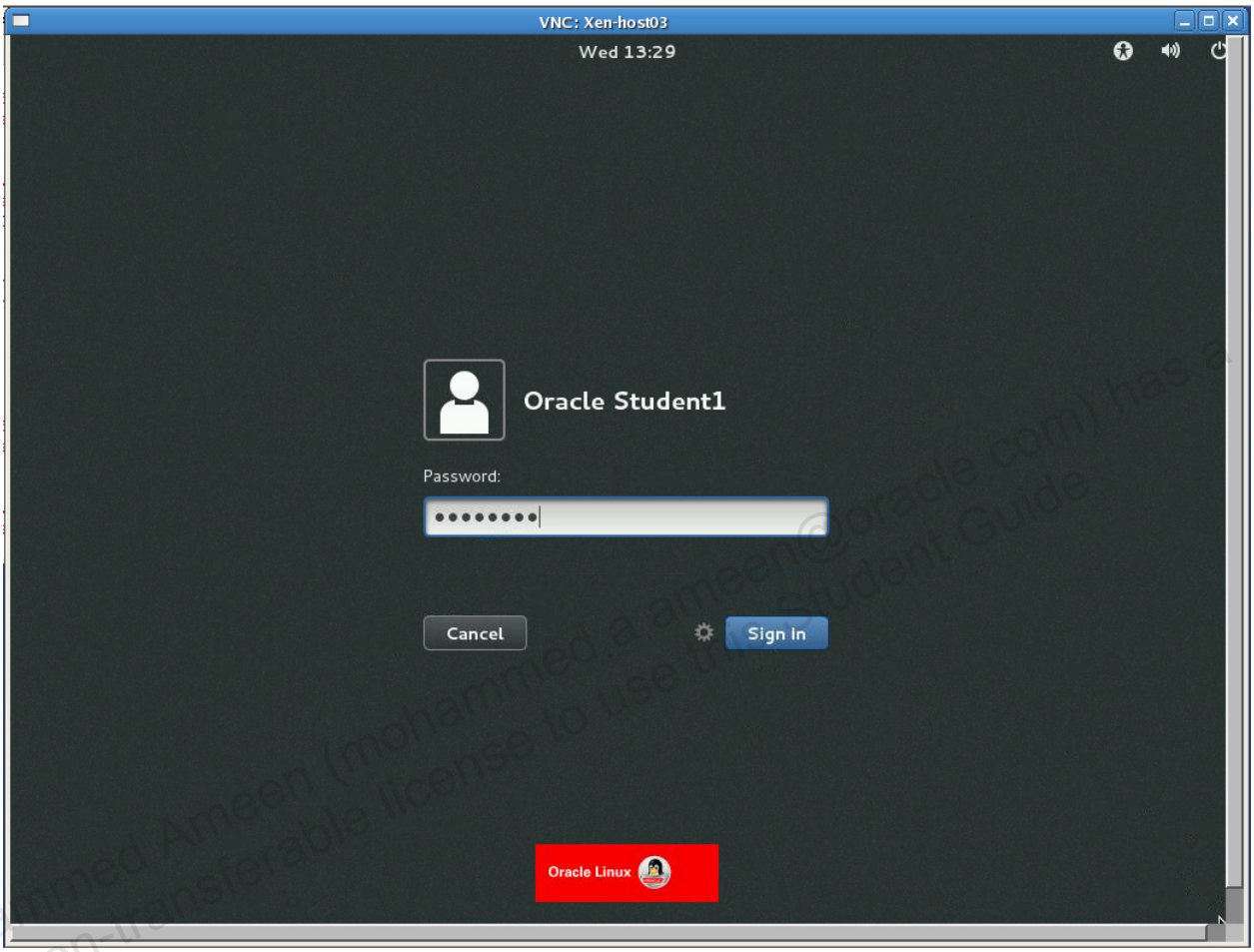
e. Click **Next**.

- You are presented with a **Password:** prompt.
- Because no password has been assigned to `student2`, the account is locked.
- Regardless of what you enter for a password, you cannot log in as `student2`.
- Click **Cancel** to continue.

f. Log in as `student1` by selecting **Oracle Student1** from the login menu.

g. Enter `password` when prompted for the password, as shown in the following.

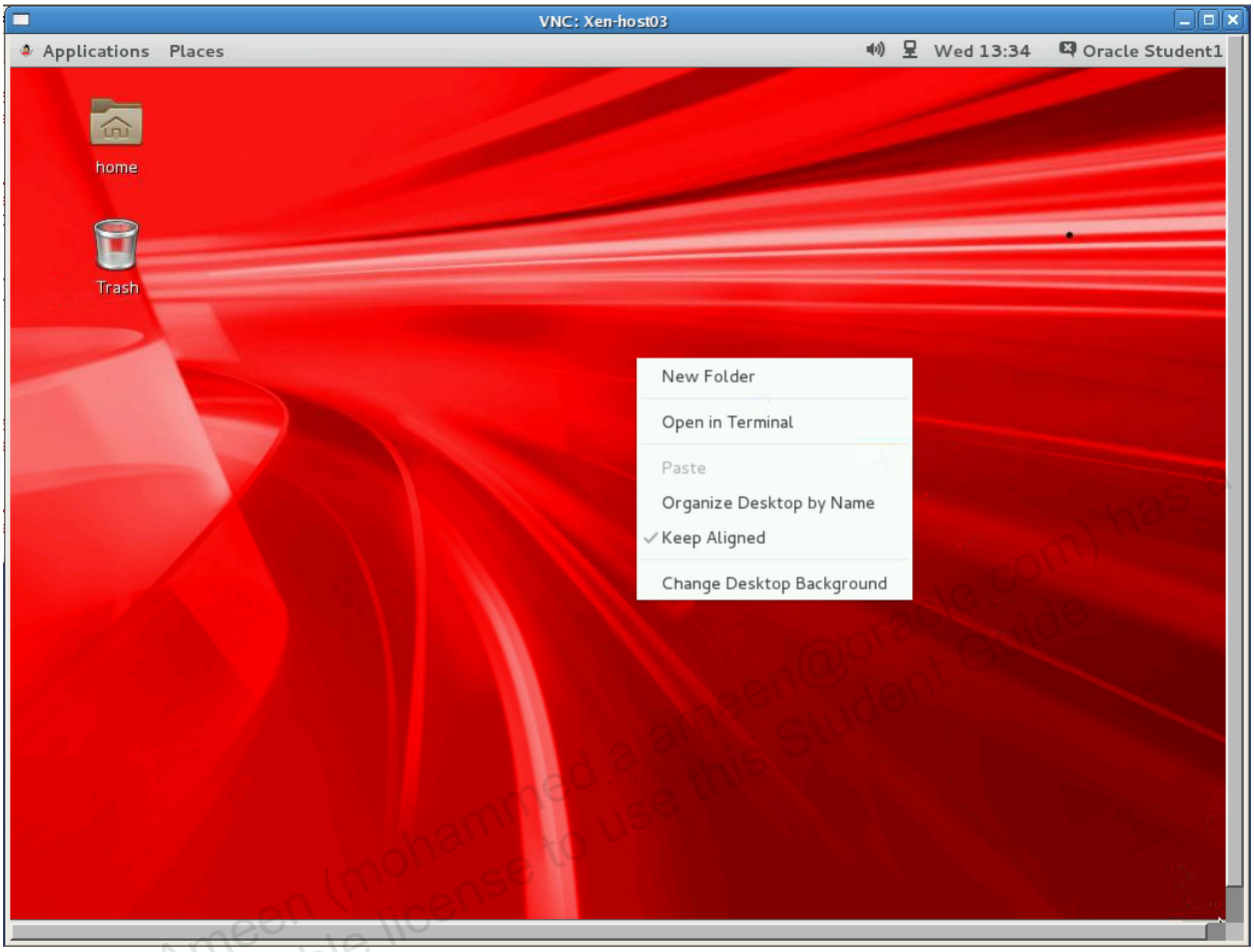
- The actual password is replaced by a string of asterisks (`********`) for security purposes.



h. Click **Sign In** to successfully log in as `student1`.

- Because this is the first time you have logged in as this user, a series of screens appears.
- Click **Next** in the Welcome window.
- Click **Next** in the Input Sources window.
- Click **Next** in the Online Accounts window.
- Click **Start using Oracle Linux Server**.
- Click **Page->Close** to close the GNOME Help window.

i. Right-click the desktop to display the pop-up menu as shown in the following.



j. Select **Open in Terminal** to open a terminal window.

k. From the terminal window, use the `whoami` command to verify that you are logged in as `student1`.

```
$ whoami
student1
```

l. Use the `su -` command to become the `root`. The password is `oracle`.

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

Practice 10-2: Administering Group Accounts

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:1000:
student1:x:1001:
student2:x:1055:
students:x:1056:
```

   - Note 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 the `student1` and `student2` users to the `students` group.

```
# usermod -aG 1056 student1
# tail -1 /etc/group
students:x:1056:student1
# usermod -aG 1056 student2
# tail -1 /etc/group
students:x:1056:student1,student2
```

   - Note 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:1001:1001:Oracle Student1:/home/student1:/bin/bash
student2:x:1055:1055:Oracle Student2:/home/student2:/bin/sh
```

   - Note that the `student1` primary group is still `1001`.
   - Note that the `student2` primary group is still `1055`.

Practice 10-3: Implementing 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
```

   - Note 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
```

   - Note 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
```

   - Note 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
```

- Note 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
```

- Note 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
```

- Note 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
```

- The `touch` command updates the time stamp, which 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
logout
$ whoami
oracle
$ exit
logout
# whoami
root
```

Practice 10-4: Configuring 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 the `student1` user.

```
# grep student1 /etc/shadow
student1:$6$...:16337:0:99999:7:::
```

   - This example gives `16337` days since the password changed (counted in days since Jan 1, 1970). This differs 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`.
   - Use the `-l` (list) option.

```
# chage -l student1
Last password change                                    : Sep 24, 2014
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) [2014-09-24]: ENTER
       Password Expiration Warning [7]:  ENTER
       Password Inactive [-1]:  ENTER
       Account Expiration Date (YYYY-MM-DD) [-1]:  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:$6$...:16337:14:30:7:::
```

c. Use the `chage` command to view password aging information for user `student1`.

```
# chage -l student1
Last password change                                    : Sep 24, 2014
Password expires                                        : Oct 24, 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.

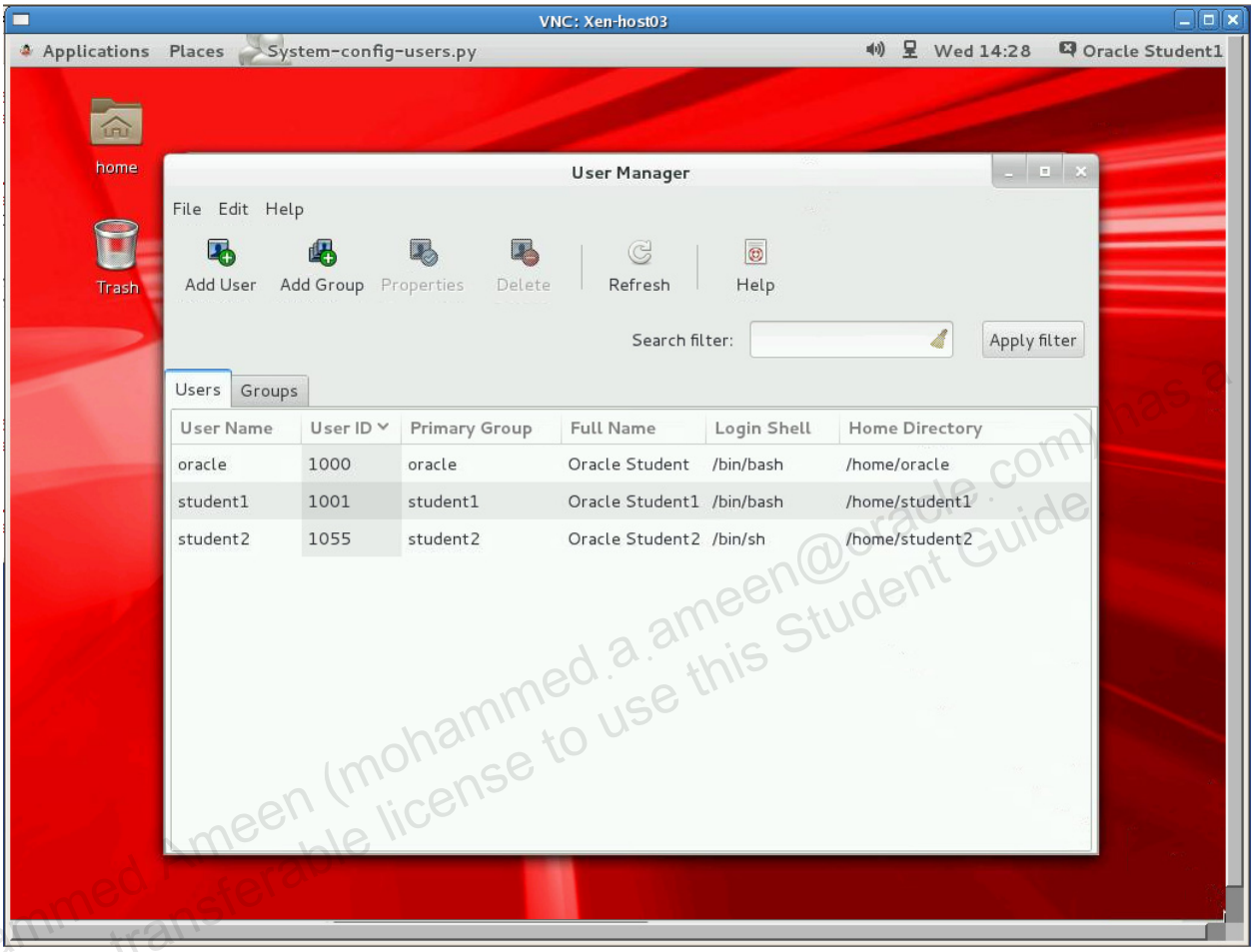
Tasks

1. Use the `yum install` command to install the `system-config-users` package.
   - Answer "`y`" when prompted.

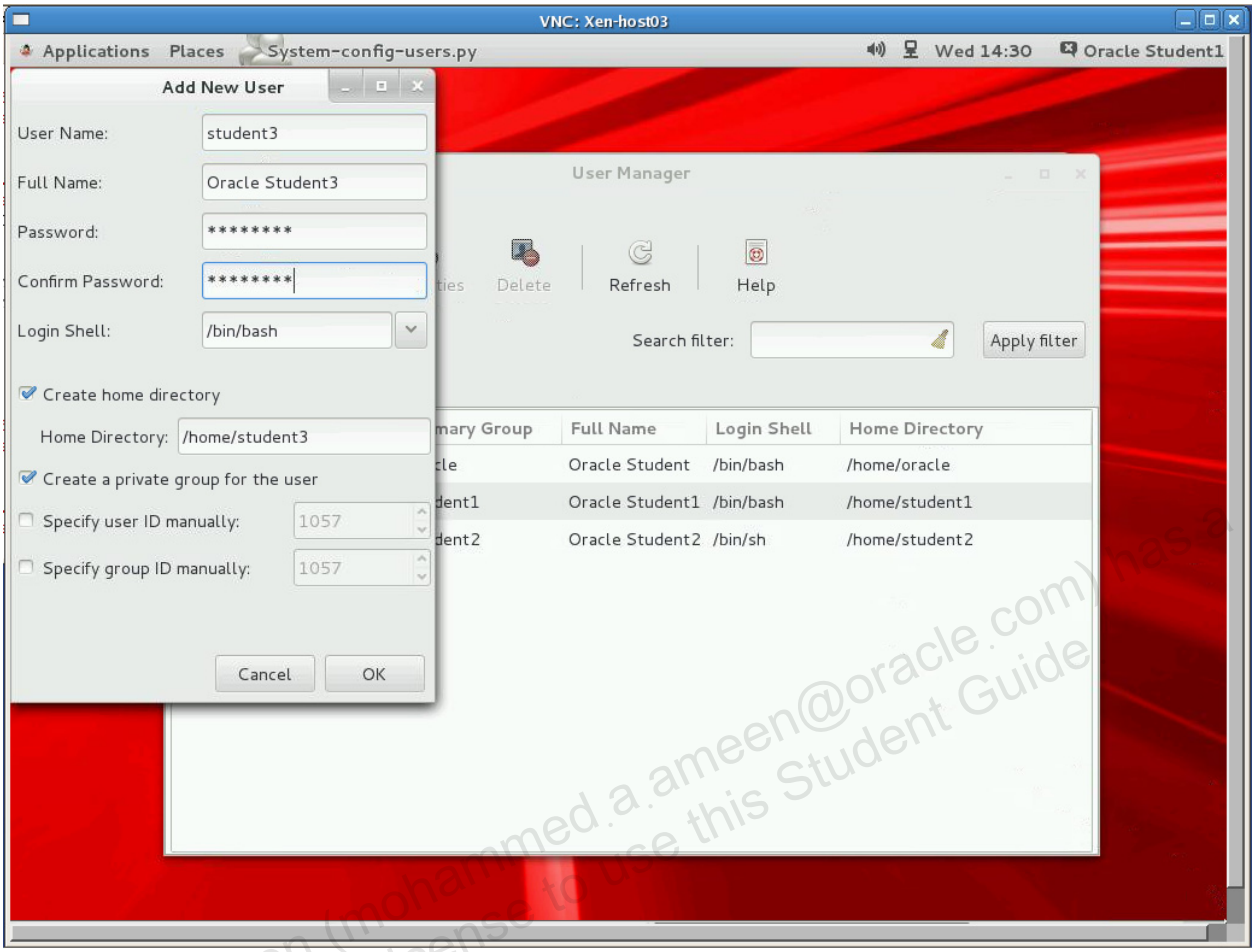
```
# yum install system-config-users
...
Transaction Summary
============================================================
Install  1 Package (+3 Dependent packages)

Total download size: 805 k
Installed size: 3.9 M
Is this ok [y/d/N]: y
...
Complete!
```

2. 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:
      - Use `password` for the **Password** and **Confirm Password** fields.

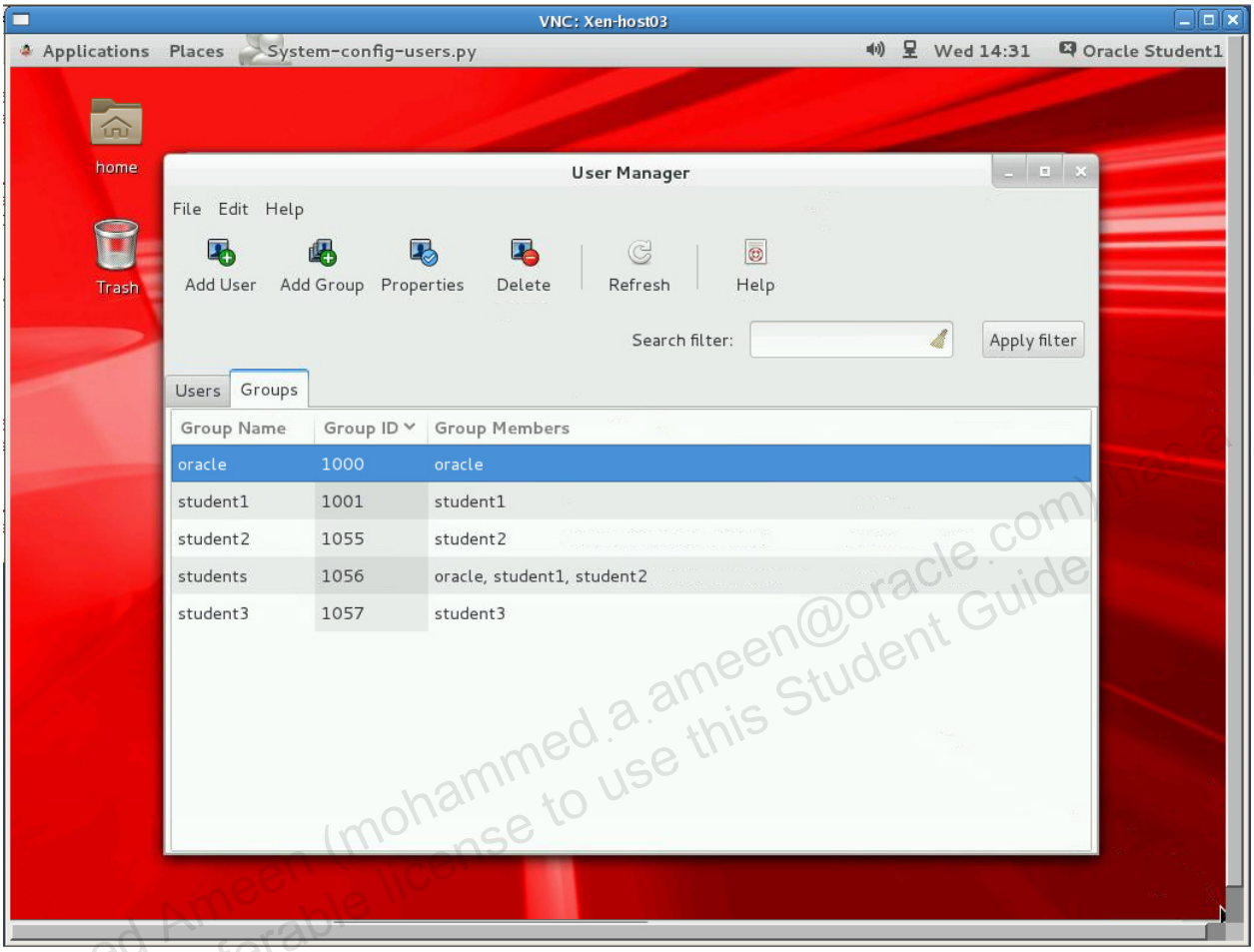


c. Click **OK**.

d. Click **Yes** to use the weak password.

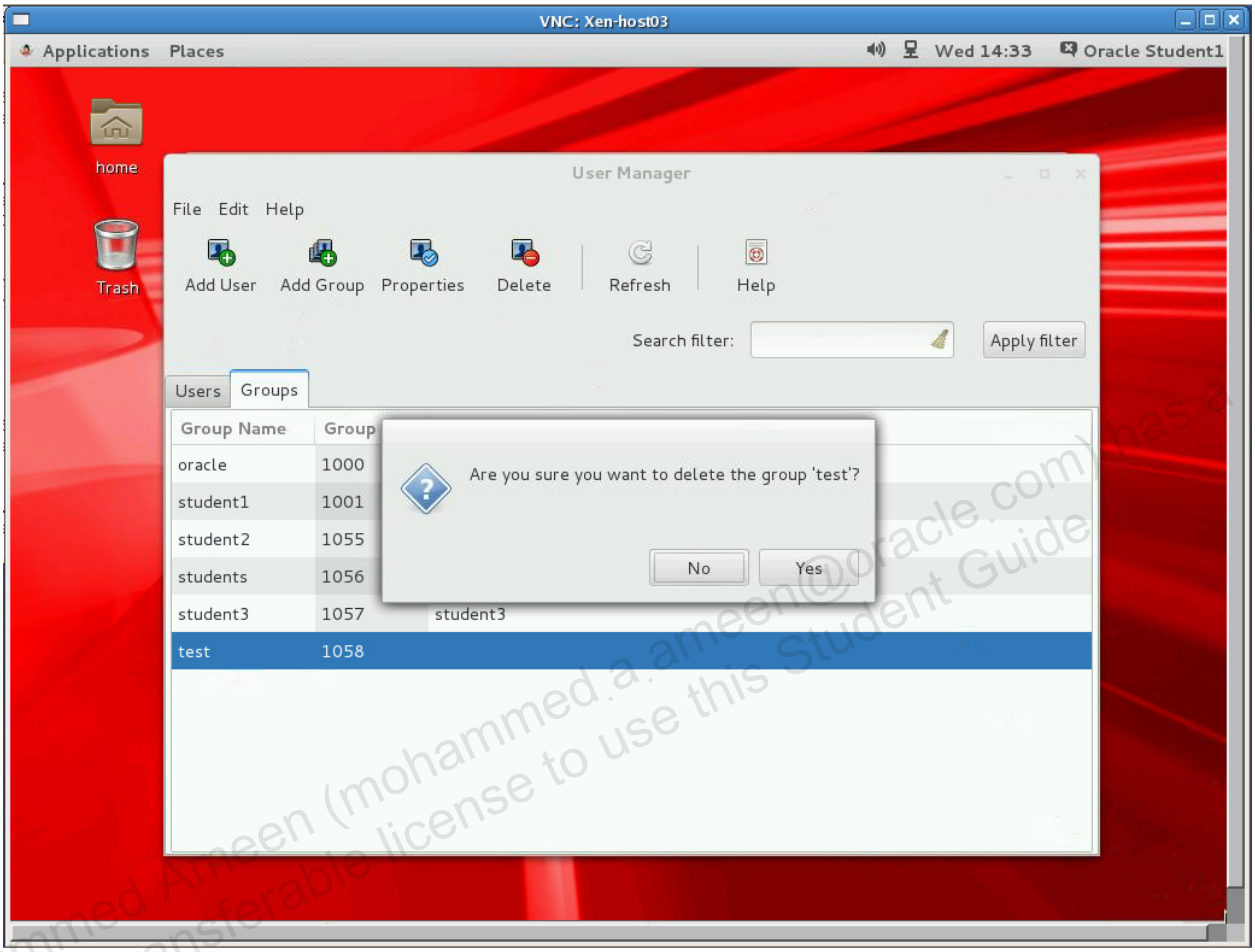
- The updated list of user accounts is displayed.

3. Add a new group.
   a. Click the **Groups** tab to display the list of group accounts, as shown in the following.



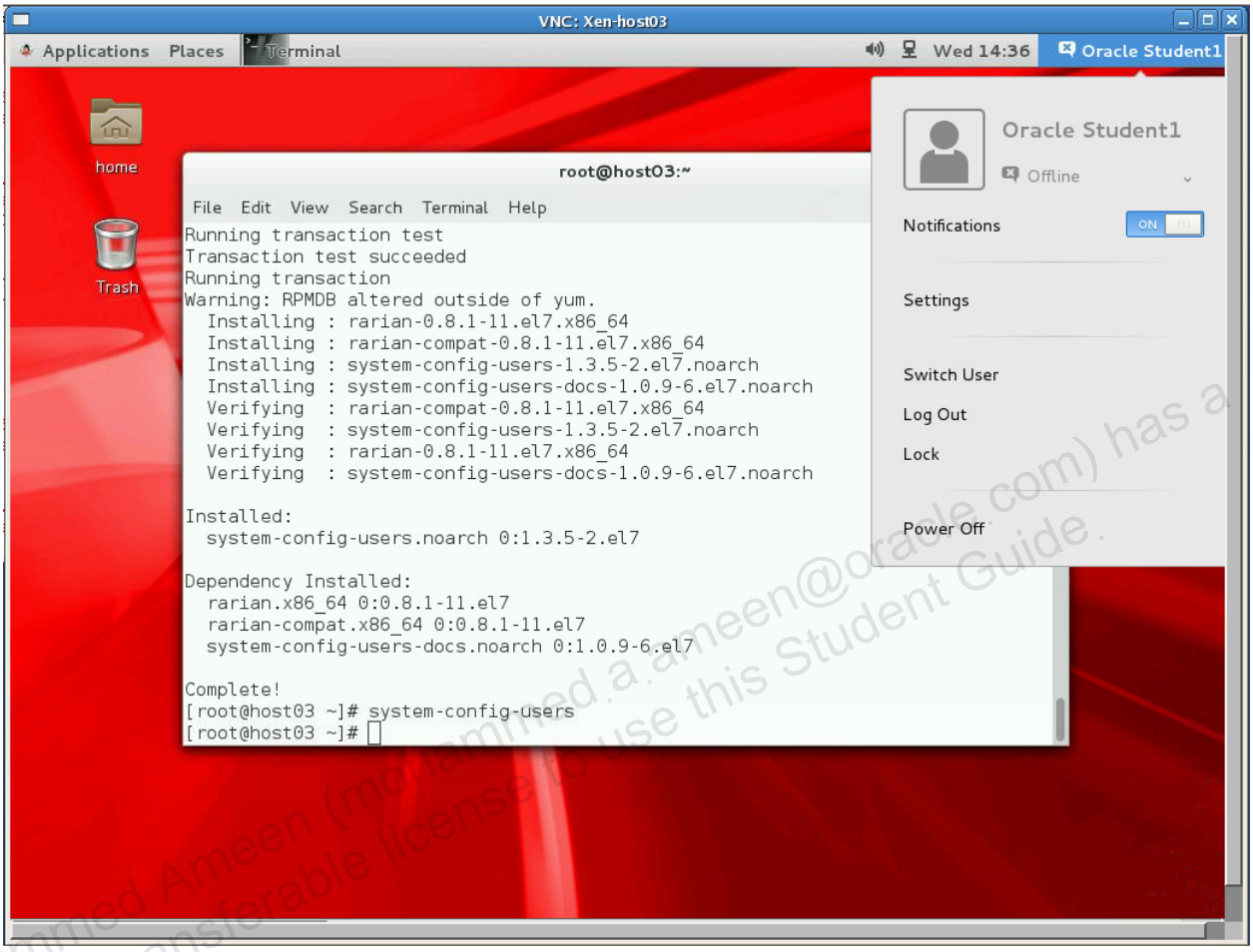
   b. Click **Add Group** and provide a **Group Name** of your choice when prompted.
   c. Click **OK** and notice the updated group list.

4. Delete a group.
   a. Select the newly added group and click the **Delete** button.
   b. Click **Yes** to confirm the delete, as shown in the following.

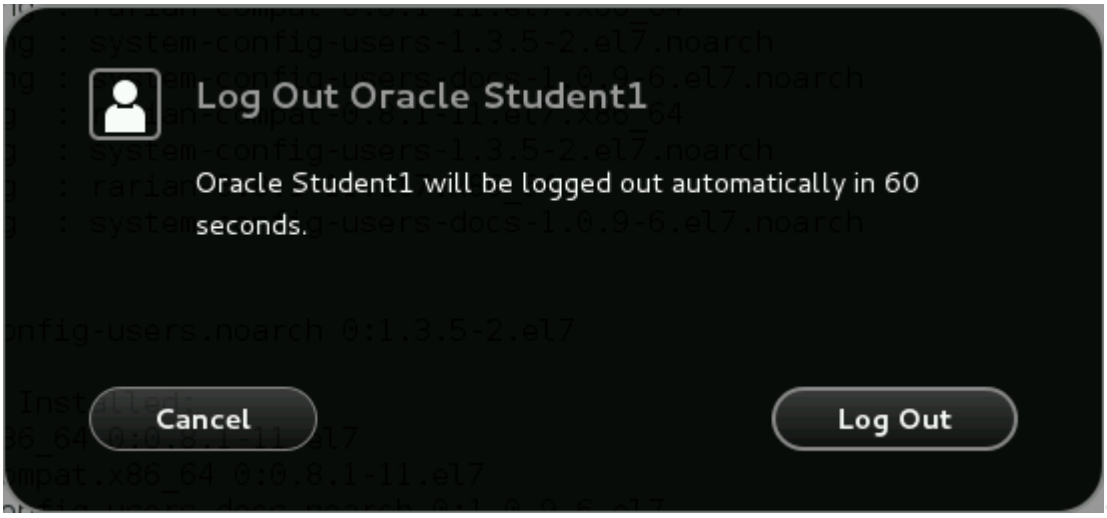


   c. Select the `student1` group and click the **Delete** button.
      - Note that you cannot delete this group, because the user `student1` user still exists.
5. Exit the User Manager tool.
   Select **Quit** from the **File** menu.

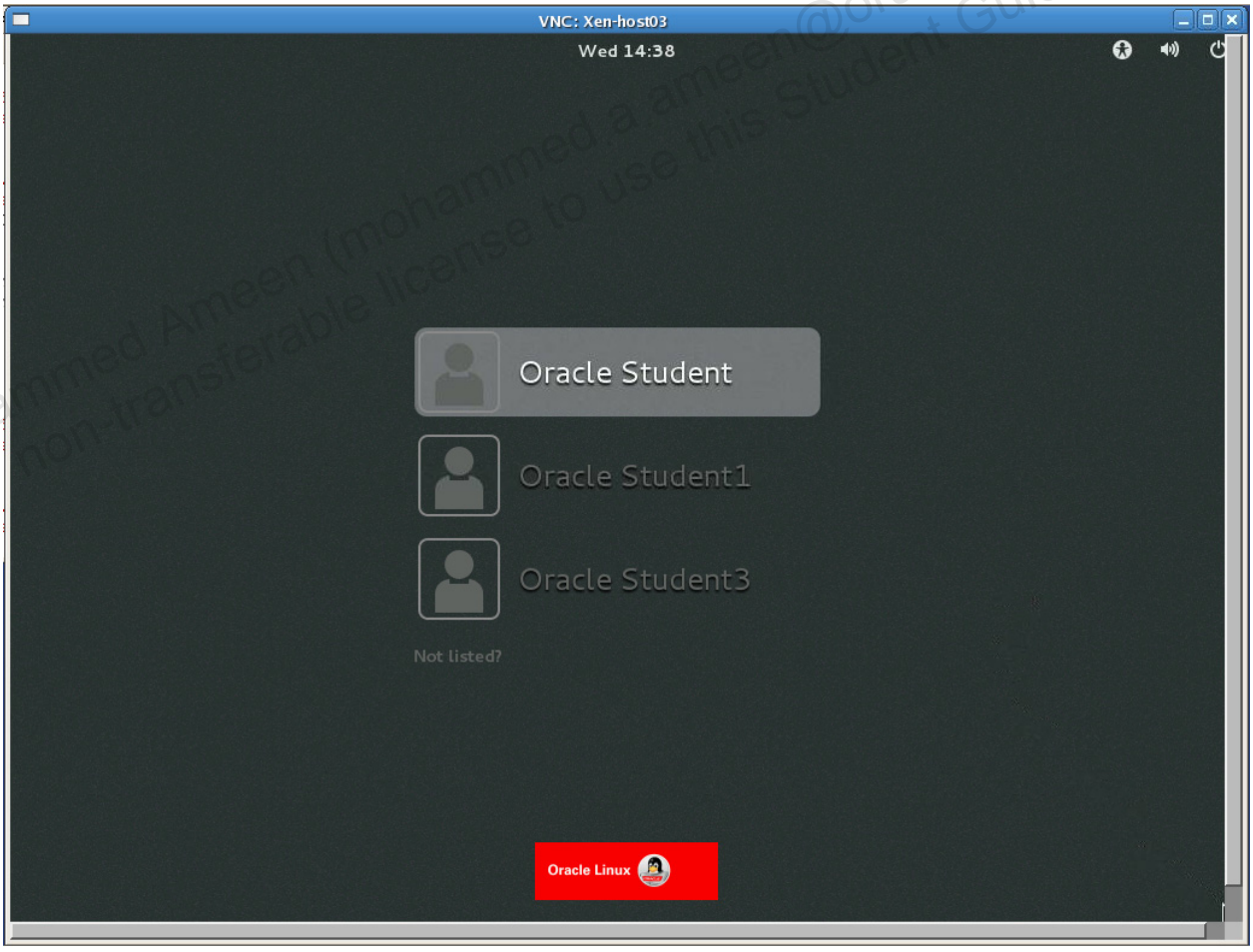
6. Log out as user `student1`.
   a. Log out as user `student1` by clicking **Oracle Student1** and selecting **Log Out** from the pop-up menu, as shown in the following:



b. Click **Log Out** in the pop-up as shown in the following.



7. Log in as the `oracle` user.
   a. Log in as the `oracle` user by selecting **Oracle Student** from the menu as shown in the following.



   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. Use `oracle` as the password.

```
$ 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
```

   - Note 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 in the following involves the modification of a Pluggable Authentication Module (PAM) configuration file.
   - PAM is covered in another course.

   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 password for the `root` user is `oracle`.

```
$ whoami
student1
$ su –
Password: oracle
su: Permission denied
```

- The `su` command fails 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 password for the `root` user 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 the `new_user` user.

```
$ useradd new_user
bash: /usr/sbin/useradd: Permission denied
```

   - Note that 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 the `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.
```

   - Note that the command fails because the `oracle` user is not in the `sudoers` file.
   - Also note that 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 ...
...
```

- Note 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 and then run the `whoami` command to verify you are the `oracle` user.

```
# exit
logout
$ whoami
oracle
```

   b. Use the `sudo useradd` command to add the `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:1058:1058::/home/new_user:/bin/bash
```

4. Run another command, which requires administrative privileges.
   a. Use the `systemctl` command to restart the `atd` service.

```
$ systemctl restart atd
Failed to issue method call: Access denied
```

      - Note that the command failed with "Access denied"
   b. Use `sudo` and run the `service` command again, to restart the `atd` service.

```
$ sudo systemctl restart atd
```

      - Note that there was no error message, implying the command succeeded this time.
      - Also note that 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: Partitions, File Systems, and Swap

Chapter 11

Practices for Lesson 11: Overview

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

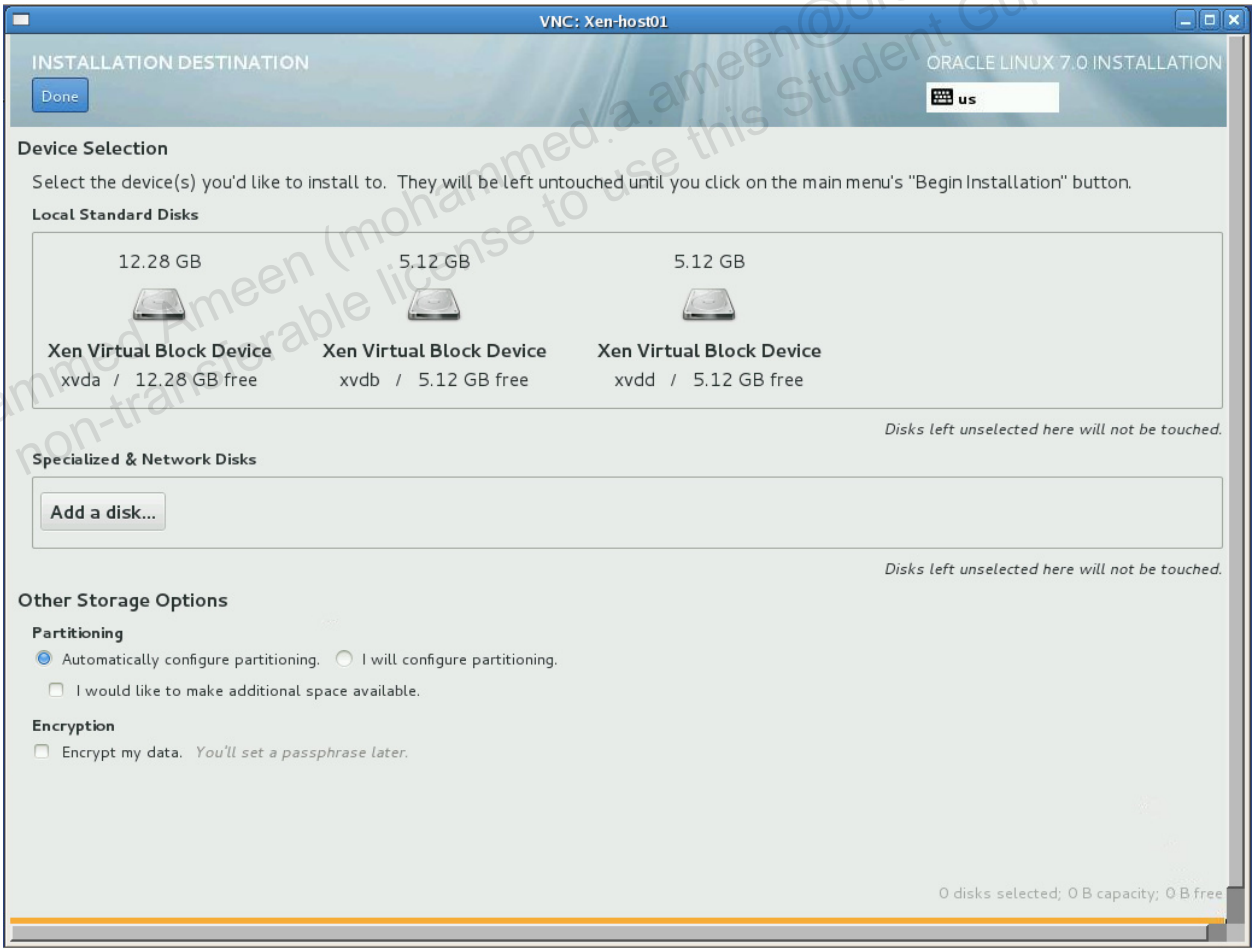
In this practice, you verify the selections made during installation regarding disk partitioning, mount points for file systems, and swap space.

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:

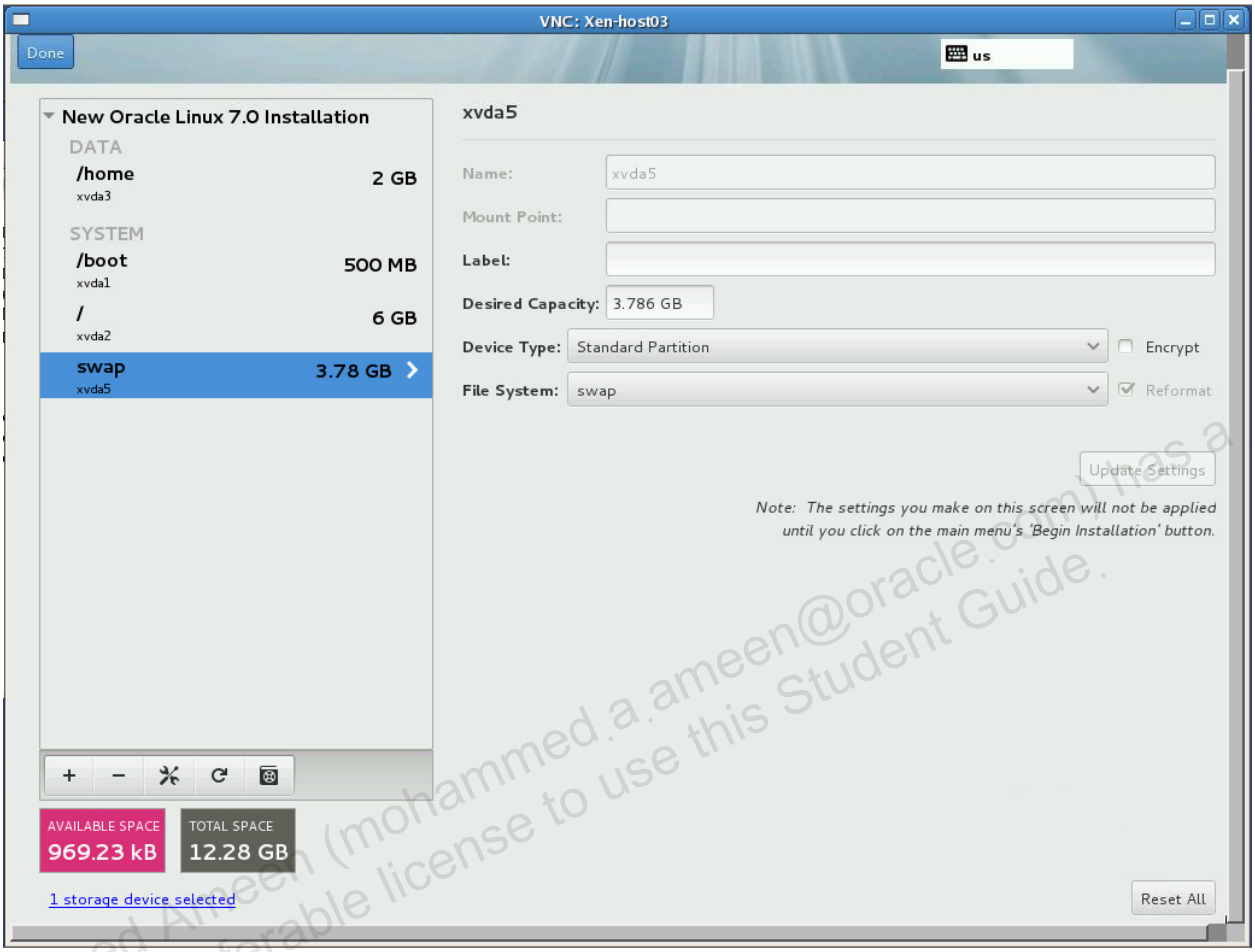


Use the `fdisk` command to display the partition table.

```
# fdisk -l | grep /dev
Disk /dev/xvda: 12.9 GB, 12884901888 bytes, 25165824 sectors
/dev/xvda1    *    2048   1026047     512000    83    Linux
/dev/xvda2      1026048  13314047    6144000    83    Linux
/dev/xvda3     13314048  17410047    2048000    83    Linux
/dev/xvda4     17410048  25165823    3877888    5  Extended
/dev/xvda5     17412096  25165823    3876864   82  Linux swap...
Disk /dev/xvdb: 5368 MB,  5368709120 bytes, 10485760 sectors
Disk /dev/xvdd: 5368 MB,  5368709120 bytes, 10485760 sectors
```

- 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 can be designated as an Extended partition.
  - This Extended partition can then be subdivided into multiple logical partitions.
  - The `/dev/xvda5` is a logical partition, which is designated as a Linux swap partition.

2. Relate the mounted partitions to selections made during installation.
   - Recall the final partition table created during the installation, as 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  3.9G   1.6G   72%  /
...
/dev/xvda3     1.9G   42M   1.8G    1%  /home
/dev/xvda1     477M  134M   314M   30%  /boot
...
```

   - 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
Welcome to dfisk (util-linux 2.23.2).
...
Command (m for help):
```

   b. Display the `fdisk` menu.

```
Command (m for help): m
Command action
   a    toggle a bootable flag
   b    edit bsd disklabel
   c    toggle the dos compatibility flag
   d    delete a partition
   g    create a new empty GPT partition table
   G    create an IRIX (SGI) partition table
   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)
```

   c. Add a new primary partition, giving the partition number `1`.

```
Command (m for help): n
Partition type:
    p    primary partition (0 primary, 0 extended, 4 free)
```

```
    e     extended
Select (default p): p
Partition number (1-4, default 1): 1
```

- A maximum of four primary partitions can be placed on any hard disk.
- One of the four partitions can be designated as an extended partition. This partition can then be subdivided into multiple logical partitions.

d. 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
Partition 1 of type Linux and of size 1 GiB is set
```

e. Display the new partition table.

```
Command (m for help): p
Disk /dev/xvdb: 5368 MB, 5368709120 bytes, 10485760 sectors
...
Disk label type: dos
...
    Device Boot   Start       End      Blocks    Id    System
/dev/xvdb1          2048   2100000    1048976+   83    Linux
```

f. 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.
```

g. 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          2048   2100000     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 3.1
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) ...
```

```
  help [COMMAND]      print general help, or help on COMMAND
  mklabel,mktable LABEL-TYPE ...
  mkpart PART-TYPE [FS-TYPE] START END ...
  name NUMBER NAME    name partition NUMBER as NAME
...
```

c. Enter `print` to print the partition table.

```
(parted) print
Error: /dev/xvdd: unrecognized disk label
Model: Xen Virtual Block Device (xvd)
Disk /dev/xvdd: 5369MB
Sector size (logical/physical): 512B/512B
Partition Table: unknown
Disk Flags:
```

- Note the "`unrecognized disk label`" error.
- Note that the partition table is "`unknown`".

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: btrfs, nilfs2, ext4, ext3, ext2, fat32...
  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.

```
(parted) mkpart
Error: /dev/xvdd: unrecognized disk label
```

- The `mkpart` command fails due to an unrecognized disk label.

f. Get `help` for the `mklabel` command.

```
(parted) help mklabel
  mklabel, mktable LABEL-TYPE     create a new disklabel
(partition label)
  LABEL-TYPE is one of: aix, amiga, bsd, dvh, gpt, mac, msdos,
pc98, sun, loop
```

g. Use the `mklabel` command to create a new disk label of type `gpt`.

```
(parted) mklabel gpt
```

h. 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: gpt
Disk Flags:

Number  Start  End  Size  File system  Name  Flags
```

- Note that the partition table is now "gpt" and there is no error message.

i. Use the mkpart command to create a partition, using the following parameters:

- The mkpart command works after using the mklabel command.

```
(parted) mkpart
Partition name? []? ENTER
Filesystem type? [ext2] ENTER
Start? 0
End? 20%
Warning: The resulting partition is not properly aligned for
best performance. Ignore/Cancel? i
```

j. Print the partition table.

```
(parted) print
Model: Xen Virtual Block Device (xvd)
Disk /dev/xvdd: 5369MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start    End      Size     File system  Name  Flags
1       17.4kB   1074MB   1074MB
```

k. Save the partition table and exit.

```
(parted) quit
Information: You may need to update /etc/fstab
```

l. Use the fdisk command to list the partition table on /dev/xvdd.

- Note the warning message.

```
# fdisk -l /dev/xvdd
WARNING: fdisk GPT support is currently new, and therefore in an
experimental phase. Use at your own discretion.

Disk /dev/xvdd: 5368 MB, 5368709120 bytes, 10485760 sectors
...
Disk label type: gpt

#   Start          End     Size    Type                Name
 1     34      2097151     1024M   Microsoft basic
```

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
  11      0   4074496 sr0
```

   - 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 <date_time>  /dev/xvda
brw-rw----. 1 root disk 202,  1 <date_time>  /dev/xvda1
brw-rw----. 1 root disk 202,  2 <date_time>  /dev/xvda2
brw-rw----. 1 root disk 202,  3 <date_time>  /dev/xvda3
brw-rw----. 1 root disk 202,  4 <date_time>  /dev/xvda4
brw-rw----. 1 root disk 202,  5 <date_time>  /dev/xvda5
brw-rw----. 1 root disk 202, 16 <date_time>  /dev/xvdb
brw-rw----. 1 root disk 202, 17 <date_time>  /dev/xvdb1
brw-rw----. 1 root disk 202, 48 <date_time>  /dev/xvdd
brw-rw----. 1 root disk 202, 49 <date_time>  /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 1.42.9 (28-Dec-2013)
warning: 100 blocks unused.
Filesystem label=
OS type: Linux
...
Writing inode tables: done
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: 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 1.42.9 (28-Dec-2013)
Filesystem label=Test
OS type: Linux
...
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: dome
```

   b. Display the attributes of the `/dev/xvdd1` block device.

```
# blkid /dev/xvdd1
/dev/xvdd1: LABEL="Test" UUID="..." TYPE="ext4" PARTUUID="..."
```

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: LABEL="Dev"  UUID="..." SEC_TYPE="ext2" TYPE="ext3"
/dev/xvdd1: LABEL="Test" UUID="..." TYPE="ext4" PARTUUID="..."
```

      - Note 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  3.9G   1.6G   72%  /
...
/dev/xvda3     1.9G   42M   1.8G    3%  /home
/dev/xvda1     477M  134M   314M   30%  /boot
...
/dev/xvdb1    1008M   34M   924M    4%  /Dev
/dev/xvdd1     992M  2.6M   923M    1%  /Test
```

   e. Use the `mount` command to display the mounted file systems.

```
# mount
...
/dev/xvda2 on / type ext4 (rw,relatime,seclabel,data=ordered)
...
/dev/xvda3 on /home type ext4 (rw,relatime,seclabel,data=...)
/dev/xvda1 on /boot type ext4 (rw,relatime,seclabel,data=...)
...
```

```
/dev/xvdb1 on /Dev type ext3 (rw,relatime,seclabel,errors=...)
/dev/xvdd1 on /Test type ext4 (rw,relatime,seclabel,data=...)
```

- Note 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,nosuid,nodev,noexec,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 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, ...
```

   b. Use the `mkswap` command to initialize the swap file.

```
# mkswap /swapfile
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
swapon: /swapfile: insecure permissions 0644, 0600 suggested.
```

   - Note the suggestion to change to permissions on the file.

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: XFS File System

Chapter 12

Practices for Lesson 12: Overview

Practices Overview

In these practices, you will perform the following:

- Create and mount an XFS file system.
- Set quotas on an XFS file system.
- Back up and restore an XFS file system.

Practice 12-1: Creating an XFS File System

Overview

In this practice, you will perform the following tasks:

- View the installed XFS software packages.
- Re-label and re-partition `/dev/xvdd`.
- Create an XFS file system.
- Mount the file system and copy files to the file system.
- Attempt to resize an XFS file system.
- Change and view the parameters of an XFS file system.

Assumptions

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

Tasks

1. Use the `rpm -qa | grep xfs` command to view the installed XFS software packages.

```
# rpm -qa | grep xfs
xfsprogs-3.2.0-0.10.10.alpha2.0.2.el7.x86_64
xfsdump-3.1.3-5.el7.x86_64
```

   - Note that both `xfsprog` and `xfsdump` packages are installed.

2. Create a new partition on `/dev/xvdd`.
   a. Use the `fdisk` command to display the available devices on your system.

```
# fdisk -l | grep /dev
WARNING: fdisk GPT support is currently new, and therefore in an
experimental phase. Use at your own discretion.
Disk /dev/xvda: 12.9 GB, 12884901888 bytes, 25165824 sectors
/dev/xvda1    *    2048   1026047     512000   83  Linux
/dev/xvda2      1026048  13314047    6144000   83  Linux
/dev/xvda3     13314048  17410047    2048000   83  Linux
/dev/xvda4     17410048  25165823    3877888    5  Extended
/dev/xvda5     17412096  25165823    3876864   82  Linux swap...
Disk /dev/xvdb: 5368 MB,  5368709120 bytes, 10485760 sectors
/dev/xvdb1         2048   2100000    1048976+  83  Linux
Disk /dev/xvdd: 5368 MB,  5368709120 bytes, 10485760 sectors
```

   - Notice the "`WARNING: fdisk GPT support is currently new...`" message.
   - The `/dev/xvdd` device has a disk label type of GPT.
   - You labeled the `/dev/xvdd` device GPT and also created the `/dev/xvdd1` partition in Practice 11-2 but the `fdisk` command does not show this partition.

   b. Use the `mount` command to view the `/dev/xvdd1` partition.

```
# mount | grep /dev/xvdd1
/dev/xvdd1 on /Test type ext4 (rw,relatime,seclabel,data=...)
```

   - Note that the `/dev/xvdd1` partition is mounted on `/Test`.

c. Use the `umount` command to unmount the file system on the `/dev/xvdd1` partition.

- Before creating a new partition on a device, you must unmount all mounted file systems on the device.
- The command shown uses the partition name as an argument but you could also use the mount point, `/Test`, as the argument.

```
# umount /dev/xvdd1
```

d. Use the `parted mklabel` command as follows to create a new disk label on `/dev/xvdd`.

- The new disk label type is `msdos`, which is supported by the `fdisk` utility.
- Note the warning message. Answer `Yes` to continue.

```
# parted /dev/xvdd
...
(parted) mklabel msdos
Warning: The existing disk label on /dev/xvdd will be destroyed
and all data on this disk will be lost. Do you want to continue?
Yes/No? Yes
(parted)
```

e. Use the `parted mkpart` command as follows to create a new primary partition on `/dev/xvdd`.

- The new partition uses all space on the device.

```
(parted) mkpart
Partition type? primary/extended? primary
Filesystem type? [ext2] ENTER
Start? 0
End? 100%
Warning: The resulting partition is not properly aligned for
best performance. Ignore/Cancel? i
(parted) quit
Information: You may need to update /etc/fstab
```

f. Use the `fdisk` command to display the partition table on all available devices.

```
# fdisk -l | grep /dev
Disk /dev/xvda: 12.9 GB, 12884901888 bytes, 25165824 sectors
/dev/xvda1    *    2048   1026047     512000   83  Linux
/dev/xvda2      1026048  13314047    6144000   83  Linux
/dev/xvda3     13314048  17410047    2048000   83  Linux
/dev/xvda4     17410048  25165823    3877888    5  Extended
/dev/xvda5     17412096  25165823    3876864   82  Linux swap…
Disk /dev/xvdb: 5368 MB,  5368709120 bytes, 10485760 sectors
/dev/xvdb1         2048   2100000    1048976+  83  Linux
Disk /dev/xvdd: 5368 MB,  5368709120 bytes, 10485760 sectors
/dev/xvdd1            1   10485759   5242879+  83  Linux
```

- Note that there is no WARNING message and that the `/dev/xvdd1` partition is shown.

3. Create an XFS file system.
   a. Use the `mkfs.xfs` command with any arguments or options to display the usage.

```
# mkfs.xfs
No device name given in argument list
Usage: mkfs.xfs
/* blocksize */           [-b log=n|size=num]
/* metadata */            [-m crc=[0|1]
/* data subvol */         [-d agcount=n,agsize=n,file,name=xxx...
/* force overwrite */     [-f]
/* inode size */          [-i log=n|perblock=n|size=num,maxpct...
/* no discard */          [-K]
/* log subvol */          [-l agnum=n,internal,size=num,logdev...
/* label */               [-L label (maximum 12 characters)]
...
```

   b. Use the `mkfs.xfs` command to create an XFS file system on `/dev/xvdd1`.
      - Use the `-L XFS` option to specify a label of XFS.
      - Use the `-b size=512` option to specify a block size of 512 bytes.

```
# mkfs.xfs -L XFS -b size=512 /dev/xvdd1
meta-data=/dev/xvdd1   isize=256    agcount=4, agzise=...
         =             sectsz=512   attr=2, projid32bit=1
         =             crc=0
data     =             bsize=512    blocks=10485758, imaxpct=25
         =             sunit=0      swidth=0 blks
naming   =version 2    bsize=4096   ascii-ci=0
log      =internal log bsize=512    blocks=20480, version=2
         =             sectsz=512   sunit=0 blks, lazy-count=1
realtime =none         extsz=4096   blocks=0, rtextents=0
```

      - Note that the block size of the `data` and the `log` sections is 512 bytes. The default is 4 KB.
      - Note that the block size of the `naming` (file system directory) section is 4 KB.

   c. Use the `blkid` command to display the block device attributes: UUID, label, and file system type.

```
# blkid | grep xfs
/dev/xvdd1: LABEL="XFS" UUID=... TYPE="xfs"
```

4. Mount the XFS file system on the `/Test` mount point and verify usability.
   a. Use the `vi` editor to edit the following entry in `/etc/fstab` to mount the file system.
      - Change `LABEL=` (from `Test` to `XFS`) and the file system type (from `ext4` to `xfs`).
      - Items that you need to change are in **bold** font.

```
# vi /etc/fstab
```

```
...
LABEL=Test     /Test     ext4     defaults     0  0   (old entry)
LABEL=XFS      /Test     xfs      defaults     0  0   (new entry)
```

b. Use the `mount -a` command to mount the file systems in `/etc/fstab`.

```
# mount -a
```

c. Use the `df -h` command to display the mounted file systems.

```
# df -h
Filesystem     Size  Used  Avail  Use%  Mounted on
...
/dev/xvdd1     5.0G  4.1M   5.0G    1%  /Test
```

d. Use the `mount` command to view the `/dev/xvdd1` partition.

```
# mount | grep /dev/xvdd1
/dev/xvdd1 on /Test type xfs (rw,relatime,seclabel,attr2,...)
```

e. Use the `cp` command to copy some files to `/Test`. Use the `ls` command to list the contents of the `/Test` directory.

- Copy `/boot/vmlinuz*` to `/Test`.

```
# cp /boot/vmlinuz* /Test
# ls /Test
vmlinuz-0-rescue-...
vmlinuz-3.10.0-123.el7.x86_64
vmlinuz-3.8.13-35.3.1.el7uek.x86_64
vmlinuz-3.8.13-44.1.1.el7uek.x86_64
```

- This confirms that the XFS file system is usable.

5. Attempt to increase the size of an XFS file system.

a. Use the `xfs_growfs` command to expand the data section to use all available space.

```
# xfs_growfs -d /Test
meta-data=/dev/xvdd1   isize=256   agcount=4, agzise=...
         =             sectsz=512  attr=2, projid32bit=1
         =             crc=0
data     =             bsize=512   blocks=10485758, imaxpct=25
         =             sunit=0     swidth=0 blks
naming   =version 2    bsize=4096  ascii-ci=0
log      =internal log bsize=512   blocks=20480, version=2
         =             sectsz=512  sunit=0 blks, lazy-count=1
realtime =none         extsz=4096  blocks=0, rtextents=0
data size unchanged, skipping
```

- Notice the "`data size unchanged, skipping`" message.
- The data section was not changed because there is no space available on the underlying device, `/dev/xvdd1`.
- The `xfs_growfs` command is most often used with logical volumes.

6. Change and view the parameters of an XFS file system.
   a. Use the `xfs_admin` command to display the file system label and the UUID.

```
# xfs_admin -l /dev/xvdd1
label = "XFS"
# xfs_admin -u /dev/xvdd1
UUID = ...
```

   b. Use the `xfs_admin` command to change the file system label to "`xfs`" (all lowercase).

```
# xfs_admin -L xfs /dev/xvdd1
xfs_admin: /dev/xvdd1 contains a mounted filesystem
fatal error - couldn't initialize XFS library
```

   - Note that the command failed because the file system is mounted.

   c. Use the `umount /Test` command to unmount the file system.

```
# umount /Test
```

   d. Repeat step 6b to change the file system label to "`xfs`".

```
# xfs_admin -L xfs /dev/xvdd1
Writing all SBs
new label = "xfs"
```

   - Note that the command is successful when the file system is not mounted.

   e. Use the `mount` command to mount the file systems in `/etc/fstab`.

```
# mount -a
mount: can't find LABEL=XFS
```

   - The `/etc/fstab` entry mounts the file system by label.
   - Because the label changed, the entry in `/etc/fstab` is no longer valid.

   f. Use the `vi` editor to edit the following entry in `/etc/fstab` to mount the file system.
   - Change `LABEL=XFS` to `/dev/xvdd1`.

```
# vi /etc/fstab
...
LABEL=XFS      /Test    xfs     defaults    0  0   (old entry)
/dev/xvdd1     /Test    xfs     defaults    0  0   (new value)
```

   g. Use the `mount -a` command to mount the file systems in `/etc/fstab`.

```
# mount -a
```

   h. Use the `blkid` command to display the new label on the XFS file system.

```
# blkid | grep xfs
/dev/xvdd1: LABEL="xfs" UUID=… TYPE="xfs"
```

   - Note that the label is now in lowercase.

Practice 12-2: Setting Quotas on an XFS File System

Overview

In this practice, you perform the following:

- Enable user quotas on an XFS file system.
- Use the `xfs_quota` command in interactive mode.
- Set limits for the `oracle` user and view quota information.
- Verify quota settings limit disk space to `oracle` user.
- Remove disk quotas.

Assumptions

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

Tasks

1. Enable quotas for users on an XFS file system.
   - This step directs you to edit the `/etc/fstab` file to enable quotas.
   - Alternatively, you could use the `mount` command and include XFS quota mount options.

   a. Use the `mount` command to view the new mount options for the XFS file system.

```
# mount | grep xfs
/dev/xvdd1 on /Test type xfs (rw,relatime,seclabel,...,noquota)
```

   - Notice the "`noquota`" mount option.

   b. Use the `umount` command to unmount `/Test`.

```
# umount /Test
```

   c. Use the `vi` editor to change the mount option for the XFS file system as shown.
   - Change "`defaults`" to "`rw,quota`" for the `/dev/xvdd1` entry.

```
# vi /etc/fstab
...
/dev/xvdd1    /Test    xfs    defaults     0  0  (old value)
/dev/xvdd1    /Test    xfs    rw,quota     0  0  (new value)
```

   d. Use the `mount` command to mount the file systems in `/etc/fstab`.

```
# mount -a
```

   e. Use the `mount` command to view the new mount options for the XFS file system.

```
# mount | grep xfs
/dev/xvdd1 on /Test type xfs (rw,relatime,seclabel,...,usrquota)
```

   - Notice the new "`usrquota`" mount option.

   f. Use the `xfs_quota` command to report overall state information.

```
# xfs_quota -x -c state
User quota state on /Test (/dev/xvdd1)
  Accounting: ON
  Enforcement: ON
```

```
  Inode: #39 (1 blocks, 1 extents)
Group quota state on /Test (/dev/xvdd1)
  Accounting: OFF
  Enforcement: OFF
  Inode: N/A
Project quota state on /Test (/dev/xvdd1)
  Accounting: OFF
  Enforcement: OFF
  Inode: N/A
Blocks grace time: [7 days 00:00:30]
Inodes grace time: [7 days 00:00:30]
Realtime Blocks grace time: [7 days 00:00:30]
```

- Note that accounting and enforcement are ON for user quota but OFF for group and project quota.

2. Use the `xfs_quota` command in interactive mode.
   a. Enter the `xfs_quota` command without any options or arguments.
      - The `xfs_quota>` prompt is displayed.

```
# xfs_quota
xfs_quota>
```

   b. Enter `help` at the `xfs_quota>` prompt to display online help for the commands.

```
xfs_quota> help
df [-bir] [-hn] [-f file] -- show free and used counts for ...
help [command] -- help for one or all commands
print -- list known mount points and projects
quit -- exit the program
quota [-bir] [-gpu] [-hnNv] [-f file] [id|name]... -- show ...
Use 'help commandname' for extended help.
```

   c. Enter `help quota` at the `xfs_quota>` prompt to display extended help for the `quota` command.

```
xfs_quota> help quota
quota [-bir] [-gpu] [-hnNv] [-f file] [id|name]... -- show ...
 display usage and quota information
 -g -- display group quota information
 -p -- display project quota information
 -u -- display user quota information
 -b -- display number of blocks used
 -b -- display number of inodes used
...
```

   d. Enter `quota -b` at the `xfs_quota>` prompt to display the number of blocks used.

```
xfs_quota> quota -b
Disk quotas for user root (0)
```

```
Filesystem      Blocks   Quota   Limit   Warn/Time      Mounted on
/dev/xvdd1       18553       0       0     00 [-------] /Test
```

e. Enter `quota -i` at the `xfs_quota>` prompt to display the number of inodes used.

```
xfs_quota> quota -i
Disk quotas for user root (0)
Filesystem      Files    Quota   Limit   Warn/Time      Mounted on
/dev/xvdd1          7        0       0     00 [-------] /Test
```

f. Enter `quit` at the `xfs_quota>` prompt to exit interactive mode.

```
xfs_quota> quit
```

3. Set limits for the `oracle` user and view quota information.

a. Use the `xfs_quota` command to set a soft limit of 5 MB and a hard limit of 6 MB on the XFS file system for user `oracle`.

```
# xfs_quota -x -c 'limit -u bsoft=5m bhard=6m oracle' /Test
```

b. Use the `xfs_quota` command to print all paths with devices and identifiers.

```
# xfs_quota -x -c print
Filesystem      Pathname
/Test            /dev/xvdd1 (uquota)
```

c. Use the `xfs_quota` command to report file system usage for blocks.

- The `-h` option provides human-readable output.
- The `-b` option provides block information.

```
# xfs_quota -x -c 'free -hb'
Filesystem  Size   Used    Avail Use%  Pathname
/dev/xvdd1  5.0G  22.2M     5.0G   0%  /Test
```

d. Use the `xfs_quota` command to report file system usage for inodes.

- The `-i` option provides inode information.

```
# xfs_quota -x -c 'free -hi'
Filesystem  Inodes  Used   Free  Use%  Pathname
/dev/xvdd1    5.2m     8   5.2m    0%  /Test
```

e. Use the `xfs_quota` command to report file system quota information in human-readable form.

```
# xfs_quota -x -c 'report -h'
User quota on /Test (/dev/xvdd1)
                        Blocks
User ID      Used    Soft    Hard    Warn/Grace
----------  ---------------------------------------
root       18.1M       0       0     00 [--------]
oracle          0      5M      6M     00 [--------]
```

4. Test quota settings.
   a. Use the `chmod` command to give read, write, and execute permissions on the `/Test` directory. Use the `ls -d /Test` command to view the permissions on the `/Test` directory.

```
# chmod 777 /Test
# ls -ld /Test
drwxrwxrwx ... /Test
```

   b. Use the `su - oracle` command to become the `oracle` user.

```
# su - oracle
[oracle@host03 ~]$
```

   c. As the `oracle` user, use the `cd` command to change to the `/Test` directory.

```
[oracle@host03 ~]$ cd /Test
[oracle@host03 Test]$
```

   d. As the `oracle` user, use the `dd if=/dev/zero of=bigfile bs=1K count=7168` command to attempt to create a 7 MB file.
      - The command fails because the `oracle` user has a hard limit of 6 MB on the XFS file system.

```
[oracle@host03 Test]$ dd if=/dev/zero of=bigfile bs=1K
count=7168
dd: writing 'bigfile': Disk quota exceeded
6145+0 records in
6144+0 records out
6291456 bytes (6.3 MB) copied, ...
```

      - Notice the "Disk quota exceeded" error message.

   e. Use the `exit` command to log off as the `oracle` user. Use the `whoami` command to confirm that you are the `root` user.

```
[oracle@host03 Test]$ exit
logout
# whoami
root
```

   f. Use the `xfs_quota` command to report file system quota information in human-readable form.

```
# xfs_quota -x -c 'report -h'
User quota on /Test (/dev/xvdd1)
                        Blocks
User ID      Used    Soft    Hard    Warn/Grace
----------  ---------------------------------
root       18.1M       0       0     00 [--------]
oracle        6M      5M      6M     00 [--------]
```

      - Note that the `Used` column for the `oracle` user is `6M`.

5. Disable XFS quotas and re-test.
   a. Use the `umount` command to unmount `/Test`.

```
# umount /Test
```

   b. Use the `vi` editor to change the mount option for the XFS file system as shown.
      - Change "`rw,quota`" to "`defaults`" for the `/dev/xvdd1` entry.

```
# vi /etc/fstab
...
/dev/xvdd1     /Test     xfs     rw,quota     0  0  (old value)
/dev/xvdd1     /Test     xfs     defaults     0  0  (new value)
```

   c. Use the `mount` command to mount the file systems in `/etc/fstab`.

```
# mount -a
```

   d. Use the `mount` command to view the new mount options for the XFS file system.

```
# mount | grep xfs
/dev/xvdd1 on /Test type xfs (rw,relatime,seclabel,...,noquota)
```

      - Note that the mount option shows "`noquota`".

   e. Use the `su - oracle` command to become the `oracle` user.

```
# su - oracle
[oracle@host03 ~]$
```

   f. As the `oracle` user, use the `cd` command to change to the `/Test` directory.

```
[oracle@host03 ~]$ cd /Test
[oracle@host03 Test]$
```

   g. As the `oracle` user, use the `dd if=/dev/zero of=bigfile bs=1K count=7168` command to attempt to create a 7 MB file.

```
[oracle@host03 Test]$ dd if=/dev/zero of=bigfile bs=1K
count=7168
7168+0 records in
7168+0 records out
7340032 bytes (7.3 MB) copied, ...
```

      - The file is successfully created with no disk quotas.

   h. Use the `exit` command to log off as the `oracle` user. Use the `whoami` command to confirm that you are the `root` user.

```
[oracle@host03 xfs]$ exit
logout
# whoami
root
```

Practice 12-3: Backing Up and Restoring XFS File Systems

Overview

In this practice, you perform the following:

- Use `xfsdump` to back up an XFS file system.
- Use `xfsdump` to back up a specific file in an XFS file system.
- View the `xfsdump` inventory.
- Use `xfsrestore` to restore from a backup.

Assumptions

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

Tasks

1. Use the `xfsdump` utility.
   a. Run the `xfsdump -h` command to display the usage.

```
# xfsdump -h
xfsdump: version 3.1.3 (dump format 3.0)
xfsdump: usage: xfsdump [ -a (dump DMF dualstate files as ...
                        [ -b <blocksize> ]
                        [ -c <media change alert program> ]
                        [ -d <dump media file size> ]
                        [ -e (allow files to be excluded) ]
                        [ -f <destination> ... ]
                        [ -h (help) ]
                        [ -l <level> ]
                        [ -m (force usage of minimal rmt) ]
                        [ -o (overwrite tape) ]
                        [ -p <seconds between progress ...
                        [ -q <use QIC tape settings> ]
...
```

   b. Use the `xfsdump` command to back up the entire XFS file system on `/Test` to a local file, `/usr/tmp/backup`.
      - Omitting the `-l <level>` option performs a full backup (level 0).
      - Enter a session label of your choice when prompted.
      - Enter a media label of your choice when prompted.
      - These labels could be provided as options to the `xfsdump` command:
        - (`-L <session_label>`)
        - (`-M <media_label>`)

```
# xfsdump -f /usr/tmp/backup /Test
xfsdump: using file dump (drive_simple) strategy
xfsdump: version 3.1.3 (dump format 3.0) - type ^C for status...
```

```
================ dump label dialog ==================
please enter label for this dump session (timeout in 300 sec)
 -> Full backup of /Test on 9/29/2014
session label entered: "Full backup of /Test on 9/29/2014"
-------------------- end dialog ---------------------

xfsdump: level 0 dump of host03.example.com:/Test
xfsdump: dump date: ...
...
xfsdump: /var/lib/xfsdump/inventory created

================ media label dialog ==================
please enter label for media in drive 0 (timeout in 300 sec)
 -> Full backup to /usr/tmp/backup on 9/29/14
media label entered: "Full backup to /usr/tmp/backup on 9/29/14"
-------------------- end dialog ---------------------

xfsdump: creating dump session media file 0 (media 0, file 0)
xfsdump: dumping ino map
xfsdump: dumping directories
...
xfsdump: Dump Status: SUCCESS
```

c. Use the `xfsdump` command to back up only `/Test/bigfile` to `/usr/tmp/bigfile_backup`.

- Provide a session label of your choice as a command-line argument.
- Provide a media label of your choice as a command-line argument.

```
# xfsdump -f /usr/tmp/bigfile_backup -L "session label" -M
"media label" -s bigfile /Test
xfsdump: using file dump (drive_simple) strategy
xfsdump: version 3.1.3 (dump format 3.0) - type ^C for status...
xfsdump: level 0 dump of host03.example.comj:/Test
xfsdump: dump date: ...
...
xfsdump: creating dump session media file 0 (media 0, file 0)
xfsdump: dumping ino map
xfsdump: dumping directories
...
xfsdump: Dump Status: SUCCESS
```

- Note that you were not prompted for session label or media label.

d. Use the `xfsdump -I` command to display the inventory.

- Only selected lines are shown from the output.

```
# xfsdump -I
file system 0:
        fs id:    ...
        session 0:
                ...
                session label: "Full back of /Test on 9/29/2014"
                ...
                level:         0
                ...
                stream 0:
                        pathname: /usr/tmp/backup
                        ...
                        media file 0:
                                ...
                                media label: "Full backup ...
        session 1:
                ...
                session label: "session label"
                ...
                level:         0
                ...
                stream 0:
                        pathname: /usr/tmp/bigfile_backup
                        ...
                        media file 0:
                                ...
                                media label: "media label"
                                ...
xfsdump: Dump Status: SUCCESS
```

- Note that there are two sessions:
  - One session for the full backup of the `/Test` directory
  - One session for the backup of the single file, `bigfile`, in the `/Test` directory

e. Make a change to the `bigfile` file in the `/Test` directory. Use the `ls -l` command before and after making the change to note the difference in file size.

- The change to `bigfile` is made by copying `/etc/fstab` and overwriting the original `bigfile`.

```
# cd /Test
# ls -l bigfile
-rw-rw-r-- ... 7340032 ... bigfile
# cp /etc/fstab bigfile
```

```
cp: overwrite 'bigfile'? y
# ls -l bigfile
-rw-rw-r-- ... 662 ... bigfile
```

- In this example, the original size of `bigfile` was `7340032` bytes. The new size is `662` bytes.

f. Use the `xfsdump` command to perform a level-1 backup of the entire XFS file system on `/Test` to a local file, `/usr/tmp/level1_backup`.

- Provide a session label of your choice as a command-line argument.
- Provide a media label of your choice as a command-line argument.

```
# xfsdump -l 1 -f /usr/tmp/level1_backup -L "level 1 session" -M
"level 1 media" /Test
xfsdump: using file dump (drive_simple) strategy
xfsdump: version 3.1.3 (dump format 3.0) - type ^C for status...
xfsdump: level 1 incremental dump of host03.example.comj:/Test
based on level 0 dump begun ...
xfsdump: dump date: ...
...
xfsdump: Dump Status: SUCCESS
```

g. Use the `xfsdump -I` command to display the inventory.

- Only selected lines are shown from the output.

```
# xfsdump -I
...
        session 2:
                ...
                session label: "level 1 session"
                ...
                level:         1
                ...
                stream 0:
                        pathname: /usr/tmp/level1_backup
                        ...
                        media file 0:
                                ...
                                media label: "level 1 media"
                                ...
xfsdump: Dump Status: SUCCESS
```

- Note that the last backup is a level 1 backup.

2. Use the `xfsrestore` utility.

a. Run the `xfsrestore -h` command to display the usage.

```
# xfsrestore -h
xfsrestore: version 3.1.3 (dump format 3.0)
xfsrestore: usage: xfsrestore [ -a <alt. workspace dir> ... ]
```

```
                                [ -b <blocksize> ]
                                [ -c <media change alert ... ]
                                [ -e (don't overwrite existing ...
                                [ -f <source> ... ]
                                [ -h (help) ]
                                [ -i (interactive) ]
                                [ -m (force usage of minimal ...
                                [ -n <file> (restore only if ...
                                [ -o (restore owner/group even ...
                                [ -p <seconds between progress ...
                                [ -q <use QIC tape settings> ]
...
```

b. Use the `ls -l /Test` command to view the contents and file sizes of the XFS file system before initiating an `xfsrestore`.

```
# ls -l /Test
-rw-rw-r-- ...     662 ... bigfile
-rwxr-xr-x ... 4902184 ... vmlinuz-0-rescue-...
-rwxr-xr-x ... 4902184 ... vmlinuz-3.10.0-123.el7.x86_64
-rwxr-xr-x ... 4587024 ... vmlinuz-3.8.13-35.3.1.el7uek.x86_64
-rwxr-xr-x ... 4602072 ... vmlinuz-3.8.13-44.1.1.el7uek.x86_64
```

- Note that the size of `bigfile` is `662` bytes.

c. Use the `xfsrestore` command to restore from `/usr/tmp/backup` to the `/Test` directory.

- `/usr/tmp/backup` is a level 0 backup.

```
# xfsrestore -f /usr/tmp/backup /Test
xfsrestore: using file dump (drive_simple) strategy
xfsrestore: version 3.1.3 (dump format 3.0) - type ^C for...
...
xfsrestore: level: 0
xfsrestore: session label: "Full backup of /Test on 9/29/2014"
xfsrestore: media label: "Full backup to /usr/tmp/backup on ...
...
xfsrestore: Restore Status: SUCCESS
```

d. Use the `ls -l /Test` command to view the contents and file sizes of the XFS file system.

```
# ls -l /Test
-rw-rw-r-- ... 7340032 ... bigfile
-rwxr-xr-x ... 4902184 ... vmlinuz-0-rescue-...
-rwxr-xr-x ... 4902184 ... vmlinuz-3.10.0-123.el7.x86_64
-rwxr-xr-x ... 4587024 ... vmlinuz-3.8.13-35.3.1.el7uek.x86_64
-rwxr-xr-x ... 4602072 ... vmlinuz-3.8.13-44.1.1.el7uek.x86_64
```

- Note that the size of `bigfile` is now `7340032` bytes.

- Recall that the level 0 backup was done before changing the size of `bigfile` in step 1e.

e. Use the `rm` command to remove the `vmlinuz-3.10.0-123.el7.x86_64` file in the `/Test` directory.

```
# rm /Test/vmlinuz-3.10.0-123.el7.x86_64
rm: remove regular file '/Test/vmlinuz-3.10.0-123.el7.x86_64'? y
```

f. Use the `xfsrestore` command to restore from the `/usr/tmp/level1_backup` to the `/Test` directory.

- `/usr/tmp/level1_backup` is a level 1 backup.

```
# xfsrestore -f /usr/tmp/level1_backup /Test
xfsrestore: using file dump (drive_simple) strategy
xfsrestore: version 3.1.3 (dump format 3.0) - type ^C for...
...
xfsrestore: level: 1
xfsrestore: session label: "level 1 session"
xfsrestore: media label: "level 1 media"
...
xfsrestore: Restore Status: SUCCESS
```

g. Use the `ls -l /Test` command to view the contents and file sizes of the XFS file system.

```
# ls -l /Test
-rw-rw-r-- ...    662 ... bigfile
-rwxr-xr-x ... 4902184 ... vmlinuz-0-rescue-...
-rwxr-xr-x ... 4587024 ... vmlinuz-3.8.13-35.3.1.el7uek.x86_64
-rwxr-xr-x ... 4602072 ... vmlinuz-3.8.13-44.1.1.el7uek.x86_64
```

- Recall that the level 1 `xfsdump` to `/usr/tmp/level1_backup` (performed in step 1f) backed up only the file that changed (`bigfile`) since the previous backup.
- Therefore, only the `bigfile` (`662` bytes) was restored.
- The `vmlinuz-3.10.0-123.el7.x86_64` file was not restored, because it was not backed up in the level 1 `xfsdump`.

h. Use the `xfsrestore` utility to restore from `/usr/tmp/backup`, and then restore from `/usr/tmp/level1_backup`.

- Use the `ls -l` command after each restore to display the contents of `/Test`.

```
# xfsrestore -f /usr/tmp/backup /Test
...
xfsrestore: Restore Status: SUCCESS
# ls -l /Test
-rw-rw-r-- ... 7340032 ... bigfile
-rwxr-xr-x ... 4902184 ... vmlinuz-0-rescue-...
-rwxr-xr-x ... 4902184 ... vmlinuz-3.10.0-123.el7.x86_64
-rwxr-xr-x ... 4587024 ... vmlinuz-3.8.13-35.3.1.el7uek.x86_64
-rwxr-xr-x ... 4602072 ... vmlinuz-3.8.13-44.1.1.el7uek.x86_64
```

```
# xfsrestore -f /usr/tmp/level1_backup /Test
...
xfsrestore: Restore Status: SUCCESS
# ls -l /Test
-rw-rw-r-- ...     662 ... bigfile
-rwxr-xr-x ... 4902184 ... vmlinuz-0-rescue-...
-rwxr-xr-x ... 4902184 ... vmlinuz-3.10.0-123.el7.x86_64
-rwxr-xr-x ... 4587024 ... vmlinuz-3.8.13-35.3.1.el7uek.x86_64
-rwxr-xr-x ... 4602072 ... vmlinuz-3.8.13-44.1.1.el7uek.x86_64
```

- All files are restored, including the latest copy of `bigfile` (`662` bytes).

3. Clean up before proceeding to the lesson titled "Btrfs File System".
   a. Use the `umount` command to unmount `/Test`.
      - Use the `cd` command to ensure that you are not in the `/Test` directory.

```
# cd
# umount /Test
```

   b. Use the `umount` command to unmount `/Dev`.

```
# umount /Dev
```

   c. Use the `vi` editor to delete the two entries.

```
# vi /etc/fstab
LABEL=Dev   /Dev    ext3    defaults    0  0  (delete this line)
/dev/xvdd1  /Test   xfs     defaults    0  0  (delete this line)
```

   d. Use the `rm` command to remove the `xfsdump` files in `/usr/tmp`.

```
# rm /usr/tmp/*
rm: remove regular file '/usr/tmp/backup'? y
rm: remove regular file '/usr/tmp/bigfile_backup'? y
rm: remove regular file '/usr/tmp/level1_backup'? y
```

Practices for Lesson 13: Btrfs File System

Chapter 13

Practices for Lesson 13: Overview

Practices Overview

In these practices, you will perform the following:

- Verify that the `btrfs-progs` package is installed.
- Create Btrfs file systems.
- Resize a Btrfs file system.
- Create and mount Btrfs subvolumes and snapshots.
- Take a snapshot of a file in a Btrfs subvolume.
- Recover a corrupted Btrfs file system.

Practice 13-1: Creating a Btrfs File System

Overview

In this practice, you will perform the following tasks:

- Create a single-disk Btrfs file system with different specifications.
- Mount the file system, copy a file to it, and display the file system information.
- Resize a Btrfs file system.
- Create a two-disk Btrfs file system with different specifications.
- Mount the file system, copy a file to it, and display the file system information.

Assumptions

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

Tasks

1. View the files provided by the `btrfs-progs` package.
   a. Use the `rpm -ql` command to install the `btrfs-progs` package.

```
# rpm -ql btrfs-progs
/usr/lib64/libbtrfs.so.0
/usr/lib64/libbtrfs.so.0.1
/usr/sbin/btrfs
...
```

2. Create a single-disk Btrfs file system.
   a. Use the `fdisk` command to display the available devices on your system.

```
# fdisk -l | grep /dev
Disk /dev/xvda: 12.9 GB, 12884901888 bytes, 25165824 sectors
/dev/xvda1    *    2048   1026047     512000   83    Linux
/dev/xvda2      1026048  13314047    6144000   83    Linux
/dev/xvda3     13314048  17410047    2048000   83    Linux
/dev/xvda4     17410048  25165823    3877888    5  Extended
/dev/xvda5     17412096  25165823    3876864   82  Linux swap...
Disk /dev/xvdb: 5368 MB,  5368709120 bytes, 10485760 sectors
/dev/xvdb1         2048   2100000    1048976+  83    Linux
Disk /dev/xvdd: 5368 MB,  5368709120 bytes, 10485760 sectors
/dev/xvdd1            1  10485759    5242879+  83    Linux
```

   - The `/dev/xvdb` and `/dev/xvdd` devices each have one partition.
   - Delete the partitions and use the entire devices in this practice.

   b. Use the `fdisk` command to delete the `/dev/xvdb1` partition.
   - This step assumes that the `/dev/xvdb1` partition exists.
   - If the `/dev/xvdb1` partition does not exist, skip this step and go to step 2c.

```
# fdisk /dev/xvdb
...
Command (m for help): d
```

```
Selected partition 1
Partition 1 is deleted
Command (m for help): w
Calling ioctl() to re-read partition table.
Syncing disks.
```

c. Use the `fdisk` command to delete the `/dev/xvdd1` partition.

- This step assumes that the `/dev/xvdd1` partition exists.
- If the `/dev/xvdd1` partition does not exist, skip this step and go to step 2d.

```
# fdisk /dev/xvdd
...
Command (m for help): d
Selected partition 1
Partition 1 is deleted
Command (m for help): w
Calling ioctl() to re-read partition table.
Syncing disks.
```

d. Use the `mkfs.btrfs` command to make a Btrfs file system on `/dev/xvdb`.

```
# mkfs.btrfs -L Btrfs /dev/xvdb
/dev/xvdb appears to contain a partition table (dos).
Error: Use the -f option to force overwrite.
```

e. Repeat the previous command but include the `-f` option.

```
# mkfs.btrfs -f -L Btrfs /dev/xvdb
WARNING! - Btrfs Btrfs v3.12 IS EXPERIMENTAL
WARNING! - see http://btrfs.wiki.kernel.org before using
Turning ON incompat feature 'extref': increased hardlink limit
per file to 65536
fs created label Btrfs on /dev/xvdb
     nodesize 16384 leafsize 16384 sectorsize 4096 size 5.00GiB
Btrfs v3.12
```

f. Run the `blkid` command to display the block device attributes: UUID, label, and file system type.

```
# blkid | grep btrfs
/dev/xvdb:  LABEL="Btrfs" UUID=... UUID_SUB=... TYPE="btrfs"
```

g. Use the `mkdir` command to make the `/btrfs` directory. This directory is used as the mount point.

```
# mkdir /btrfs
```

h. Use the `vi` editor to add the following entry in `/etc/fstab` to mount the file system:

```
# vi /etc/fstab
...
/dev/xvdb  /btrfs  btrfs  defaults  0  0
```

- You can use LABEL or UUID when mounting as well.

i. Use the `mount` command to mount the file systems in `/etc/fstab`.

```
# mount -a
```

j. Run the `df -h` command to display the space information for `/btrfs`.

```
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb       5.0G  320K   4.5G    1%  /btrfs
```

k. Use the `cp` command to copy `/boot/vmlinuz-3.10*` to `/btrfs`.

```
# cp /boot/vmlinuz-3.10* /btrfs
```

l. Run the `df -h` command again to display the space information for `/btrfs`.

```
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb       5.0G  320K   4.5G    1%  /btrfs
```

- Note that there is no difference in the amount of space used and available.

m. Run the `sync` command to flush the file system buffers. Run the `df -h /btrfs` command again.

```
# sync
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb       5.0G  5.0M   4.5G    1%  /btrfs
```

- Note that the `sync` command resulted in a difference in the output of `df -h`.
- Running the `sync` command allows the `df -h` command to be somewhat truthful.
- However, `df` often shows incorrect free space. This inaccuracy issue is briefly discussed on the `btrfs filesystem df` page in the lesson titled “Btrfs File System.”
- It is not a bug, the main point is that `df` cannot be trusted on Btrfs volumes regardless of whether `sync` runs or not.
- Essentially, `df` knows nothing about how Btrfs duplicates and COWs data, so it is only accurate on a very small number of Btrfs volumes and rapidly becomes invalid.
- These practices use `df`, and it works, but these are small disks.
- For more on why you should not use `df` on a Btrfs volume, see: https://btrfs.wiki.kernel.org/index.php/FAQ#Help.21_Btrfs_claims_I.27m_out_of_space.2C_but_it_looks_like_I_should_have_lots_left.21.

n. Run the `btrfs file` command to display the available `btrfs filesystem` commands.

```
# btrfs file
usage: btrfs filesystem [<group>] <command> [<args>]
    btrfs filesystem df <path>
          Show space usage information for a mount point
    btrfs filesystem show [options|<path>|<uuid>]
          Show the structure of a filesystem
    ...
```

o. Use the `btrfs filesystem show` command to view the structure of the file system.

```
# btrfs file show
Label: Btrfs  uuid: ...
        Total devices 1 FS bytes used 4.86MiB
        devid    1 size 5.00GiB used 548.00MiB path /dev/xvdb
Btrfs v3.12
```

p. Use the `btrfs filesystem df` command to show the space used on the mount point.

```
# btrfs file df /btrfs
Data, single: total=8.00MiB, used=4.74MiB
System, DUP: total=8.00MiB, used=16.00KiB
System, single: total=4.00MiB, used=0.00
Metadata, DUP: total=256.00MiB, used=112.00KiB
Metadata, single: total=8.00MiB, used=0.00
```

- Note that, by default, the metadata is duplicated even on a single-disk file system.

3. Create a single-disk Btrfs file system without duplicating the metadata.

a. Use the `umount` command to unmount `/btrfs`.

```
# umount /btrfs
```

b. Use the `mkfs.btrfs -m single` command to make a Btrfs file system on `/dev/xvdb`.

```
# mkfs.btrfs -f -m single -L Btrfs /dev/xvdb
...
```

c. Use the `mount` command to mount the file system.

```
# mount -a
```

d. Use the `cp` command to copy `/boot/vmlinuz-3.10*` to `/btrfs`.

```
# cp /boot/vmlinuz-3.10* /btrfs
```

e. Run the `sync` command, and then use the `df -h` command to display the space information for `/btrfs`.

```
# sync
# df -h /btrfs
Filesystem    Size  Used  Avail  Use%  Mounted on
/dev/xvdb     5.0G  4.9M   4.8G    1%  /btrfs
```

f. Use the `btrfs filesystem df` command to show the space used on the mount point.

```
# btrfs file df /btrfs
Data, single: total=8.00MiB, used=4.74MiB
System, single: total=4.00MiB, used=16.00KiB
Metadata, single: total=264.00MiB, used=112.00KiB
```

- Note that the metadata is not duplicated.

4. Increase and decrease the size of a Btrfs file system.
   a. Use the `btrfs file resize` command to increase the size of the file system by 2 GB.

```
# btrfs file resize +2G /btrfs
Resize '/btrfs/' of '+2G'
ERROR: unable to resize '/btrfs/' - File too large
```

   - The error indicates that you are unable to increase the size of the file system.
   - The resize command does not change the size of the underlying partition.
   - If you want to enlarge a file system, you must expand the partition first.

   b. Use the `btrfs file resize` command to reduce the size of the file system by 2 GB.

```
# btrfs file resize -2G /btrfs
Resize '/btrfs/' of '-2G'
```

   - Note that you are able to reduce the size of the file system.

   c. Use the `df -h` command to display disk space usage on the `/btrfs` file system.

```
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb       3.0G  4.9M   2.8G    1%  /btrfs
```

   - Note that the size is reduced by 2 GB, from 5 GB down to 3 GB.

   d. Use the `btrfs file resize` command to increase the size of the file system by 1 GB.

```
# btrfs file resize +1G /btrfs
Resize '/btrfs/' of '+1G'
```

   - Note that the resize was successful this time because space is available on the underlying partition.

   e. Use the `df -h` command to display disk space usage on the `/btrfs` file system.

```
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb       4.0G  4.9M   3.8G    1%  /btrfs
```

   - Note that the size is increased by 1 GB, from 3 GB up to 4 GB.

5. Add a disk to and remove a disk from a Btrfs file system.
   a. Use the `btrfs device add` command to add a 5 GB disk, `/dev/xvdd`, to the existing `/btrfs` file system.

```
# btrfs device add /dev/xvdd /btrfs
/dev/xvdd appears to contain a partition table (dos).
Error: Use the -f option to force overwrite.
```

   b. Repeat the previous command but include the `-f` option.

```
# btrfs device add -f /dev/xvdd /btrfs
```

   c. Use the `df -h` command to display disk space usage on the `/btrfs` file system.

```
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb       9.0G  4.9M   8.8G    1%  /btrfs
```

- Note that the size is increased by 5 GB, from 4 GB up to 9 GB.

d. Use the `btrfs file show` command to verify that the file system now contains two disks.

```
# btrfs file show
Label: Btrfs  uuid: ...
        Total devices 2 FS bytes used 4.86MiB
        devid    1 size 4.00GiB used 276.00MiB path /dev/xvdb
        devid    2 size 5.00GiB used 0.00 path /dev/xvdd
Btrfs v3.12
```

- Note that all of the data is on `/dev/xvdb` (276.00MiB) and that `/dev/xvdd` has 0.00.

e. Run the `blkid` command to display the block device attributes: UUID, label, and file system type.

```
# blkid | grep btrfs
/dev/xvdb: LABEL="Btrfs" UUID=... UUID_SUB=... TYPE="btrfs"
/dev/xvdd: LABEL="Btrfs" UUID=... UUID_SUB=... TYPE="btrfs"
```

- Note that `/dev/xvdd` has the same label and UUID as `/dev/xvdb`.
- The two devices do have different UUID_SUB values.

f. Run the `btrfs filesystem balance` command on the file system.

```
# btrfs file balance /btrfs
Done, had to relocate 4 out of 4 chunks
```

- After adding a device, it is recommended that you run this command on the file system to redistribute the chunks of the file system across all the devices.

g. Use the `btrfs file show` command to show the effect of the balance command.

```
# btrfs file show
Label: Btrfs  uuid: ...
        Total devices 2 FS bytes used 5.05MiB
        devid    1 size 4.00GiB used 32.00MiB path /dev/xvdb
        devid    2 size 5.00GiB used 1.15GiB path /dev/xvdd
Btrfs v3.12
```

- Note that both devices now have some data on them.

h. Use the `btrfs device delete` command to remove `/dev/xvdd` from the `/btrfs` file system.

```
# btrfs device delete /dev/xvdd /btrfs
```

i. Use the `df -h` command to display disk space usage on the `/btrfs` file system.

```
# df -h /btrfs
Filesystem    Size  Used  Avail  Use%  Mounted on
/dev/xvdb     4.0G  5.0M   3.8G    1%  /btrfs
```

- Note that the size is decreased by 5 GB, from 9 GB down to 4 GB.

6. Create a two-disk Btrfs file system with the default RAID level.
   a. Use the `umount` command to unmount `/btrfs`.

```
# umount /btrfs
```

   b. Use the `mkfs.btrfs` command to make a Btrfs file system with two disks, `/dev/xvdb` and `/dev/xvdd`.

```
# mkfs.btrfs -f -L Btrfs /dev/xvd[bd]
WARNING! - Btrfs Btrfs v3.12 IS EXPERIMENTAL
WARNING! - see http://btrfs.wiki.kernel.org before using
Turning ON incompat feature 'extref': increased hardlink limit
per file to 65536
Adding device /dev/xvdd id 2
fs created label Btrfs on /dev/xvdb
     nodesize 16384 leafsize 16384 sectorsize 4096 size 10.00GiB
Btrfs v3.12
```

   - Note that `/dev/xvdd` was added and that the file system size is now 10 GiB.

   c. Run the `blkid` command to display the block device attributes: UUID, label, and file system type.

```
# blkid | grep btrfs
/dev/xvdb:  LABEL="Btrfs" UUID=... UUID_SUB=... TYPE="btrfs"
/dev/xvdd:  LABEL="Btrfs" UUID=... UUID_SUB=... TYPE="btrfs"
```

   - Note that both devices are listed, each having the same `UUID` but different `UUID_SUB` values.

   d. Use the `mount` command to mount the file system.

```
# mount -a
```

   - Note that the original `/etc/fstab` entry, which references only one device, `/dev/xvdb`, still mounts the two-disk file system.
   - You need to reference only one device even when your file system contains multiple devices.
   - In this example, you can reference either device, `/dev/xvdb` or `/dev/xvdd`, when mounting the file system.

   e. Use the `cp` command to copy `/boot/vmlinuz-3.10*` to `/btrfs`.

```
# cp /boot/vmlinuz-3.10* /btrfs
```

   f. Run the `sync` command, and then use the `df -h` command to display the space information for `/btrfs`.

```
# sync
# df -h /btrfs
Filesystem    Size  Used  Avail  Use%  Mounted on
/dev/xvdb      10G  5.5M   8.0G    1%  /btrfs
```

   g. Use the `btrfs filesystem show` command to view the structure of the file system.

```
# btrfs file show
Label: Btrfs  uuid: ...
```

```
        Total devices 2 FS bytes used 5.30MiB
        devid    1 size 5.00GiB used 1.53GiB path /dev/xvdb
        devid    2 size 5.00GiB used 1.51GiB path /dev/xvdd
Btrfs v3.12
```

- Note that the data is distributed evenly across the two devices.

h. Use the `btrfs filesystem df` command to show the space used on the mount point.

```
# btrfs file df /btrfs
Data, RAID0: total=1.00GiB, used=5.18MiB
Data, single: total=8.00MiB, used=0.00
System, RAID1: total=8.00MiB, used=16.00KiB
System, single: total=4.00MiB, used=0.00
Metadata, RAID1: total=1.00GiB, used=112.00KiB
Metadata, single: total=8.00MiB, used=0.00
```

- Note that, by default, the data is `RAID0` and the metadata is `RAID1`.

7. Create a two-disk Btrfs file system with RAID-1 for both the data and metadata.

a. Use the `umount` command to unmount `/btrfs`.

```
# umount /btrfs
```

b. Use the `mkfs.btrfs` command to make a Btrfs file system with two disks, `/dev/xvdb` and `/dev/xvdd`, and specify RAID-1 for data.

- Metadata is RAID1 by default.

```
# mkfs.btrfs -f -L Btrfs -d raid1 /dev/xvd[bd]
WARNING! - Btrfs Btrfs v3.12 IS EXPERIMENTAL
WARNING! - see http://btrfs.wiki.kernel.org before using
Turning ON incompat feature 'extref': increased hardlink limit
per file to 65536
Adding device /dev/xvdd id 2
fs created label Btrfs on /dev/xvdb
     nodesize 16384 leafsize 16384 sectorsize 4096 size 10.00GiB
Btrfs v3.12
```

c. Use the `mount` command to mount the file system.

```
# mount -a
```

d. Use the `cp` command to copy `/boot/vmlinuz-3.10*` to `/btrfs`.

```
# cp /boot/vmlinuz-3.10* /btrfs
```

e. Run the `sync` command, and then use the `df -h` command to display the space information for `/btrfs`.

```
# sync
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb        10G   11M    8.0G    1%  /btrfs
```

- Note that the output shows that the file system has a size of 10 GB.

- This is inaccurate because this is a RAID-1 array.

f. Use the `btrfs filesystem df` command to show the space used on the mount point.

```
# btrfs file df /btrfs
Data, RAID1: total=1.00GiB, used=5.18MiB
Data, single: total=8.00MiB, used=0.00
System, RAID1: total=8.00MiB, used=16.00KiB
System, single: total=4.00MiB, used=0.00
Metadata, RAID1: total=1.00GiB, used=112.00KiB
Metadata, single: total=8.00MiB, used=0.00
```

- Note that both the data and metadata are `RAID1`.
- You can also see that data is allocated in 1 GiB chunks.

Practice 13-2: Working with Subvolumes and Snapshots

Overview

In this practice, you perform the following:

- Create a Btrfs subvolume.
- Create a snapshot of the subvolume.
- List the subvolume and snapshot.
- Mount the subvolume and snapshot.
- Delete the snapshot.
- Take a snapshot of a file by using the `cp --reflink` command.

Assumptions

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

Tasks

1. Create a Btrfs subvolume.
   a. Run the `btrfs sub` command to display the available `btrfs subvolume` commands.

```
# btrfs sub
usage: btrfs subvolume <command> <args>
    btrfs subvolume create [-i <qgroupid>] [<dest>/]<name>
         Create a subvolume
    btrfs subvolume delete <subvolume> [<subvolume>...]
         Delete a subvolume(s)
    btrfs subvolume list [options] [-G [+|-]value] [-C [+|-
]value] [--sort=gen,ogen,rootid,path] <path>
         List subvolumes (and snapshots)
    btrfs subvolume snapshot [-r] <source> <dest>|
[<dest>/]<name>
         btrfs subvolume snapshot [-r] [-i <qgroupid>] <source>
<dest>|[<dest>/]<name>
    btrfs subvolume get-default <path>
         Get the default subvolume of a filesystem
    btrfs subvolume set-default <subvolid> <path>
         Set the default subvolume of a filesystem
    btrfs subvolume find-new <path> <lastgen>
         List the recently modified files in a filesystem
    btrfs subvolume show <subvol-path>
         Show more information of the subvolume
```

   b. Use the `btrfs sub create` command to create a subvolume named `SV1`.

```
# btrfs sub create /btrfs/SV1
Create subvolume '/btrfs/SV1'
```

c. Use the `ls -l` command to display the contents of `/btrfs`.

```
# ls -l /btrfs
drwxr-xr-x ... SV1
-rwxr-xr-x ... vmlinuz-3.10...
```

- Note that the `SV1` subvolume is created and appears as a regular directory.

d. Use the `mv` command to move the `vmlinuz*` file to the `/btrfs/SV1` subvolume.

```
# mv /btrfs/vmlinuz* /btrfs/SV1
```

e. Use the `ls -lR` command to display the contents of `/btrfs`.

```
# ls -lR /btrfs
/btrfs/:
drwxr-xr-x ... SV1

/btrfs/SV1:
-rwxr-xr-x ... vmlinuz-3.10...
```

- Note that the `vmlinuz*` file now resides in the `SV1` subvolume.

f. Run the `sync` command, and then use the `df -h` command to display the disk space usage on the `/btrfs` file system.

```
# sync
# df -h /btrfs
Filesystem    Size  Used  Avail  Use%  Mounted on
/dev/xvdb      10G   11M   8.0G    1%  /btrfs
```

2. Create a Btrfs snapshot.

a. Use the `btrfs sub snapshot` command to create a snapshot of the `SV1` subvolume. Name the snapshot `SV1-snap`.

```
# btrfs sub snapshot /btrfs/SV1 /btrfs/SV1-snap
Create a snapshot of '/btrfs/SV1' in '/btrfs/SV1-snap'
```

b. Use the `ls -lR` command to display the contents of `/btrfs`.

```
# ls -lR /btrfs
/btrfs/:
drwxr-xr-x ... SV1
drwxr-xr-x ... SV1-snap

/btrfs/SV1:
-rwxr-xr-x ... vmlinuz-3.10...

/btrfs/SV1-snap:
-rwxr-xr-x ... vmlinuz-3.10...
```

- Note that the new `SV1-snap` directory in `/btrfs` is created.
- The `SV1-snap` snapshot is a point-in-time copy of the `SV1` subvolume.

c. Run the `sync` command, and then use the `df -h` command to display disk space usage on the `/btrfs` file system.

```
# sync
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb        10G   11M   8.0G    1%  /btrfs
```

- Note that the creation of the snapshot did not reduce the available disk space.

d. Use the `cp` command to copy `/boot/vmlinuz-3.8.13-35*` to the `/btrfs/SV1` subvolume.

```
# cp /boot/vmlinuz-3.8.13-35* /btrfs/SV1
```

e. Use the `ls -lR` command to display the contents of `/btrfs`.

```
# ls -lR /btrfs
/btrfs/:
drwxr-xr-x ... SV1
drwxr-xr-x ... SV1-snap

/btrfs/SV1:
-rwxr-xr-x ... vmlinuz-3.10...
-rwxr-xr-x ... vmlinuz-3.8.13...

/btrfs/SV1-snap:
-rwxr-xr-x ... vmlinuz-3.10...
```

- Note that the `SV1` subvolume now contains two `vmlinuz*` files.

3. Mount the subvolume and the snapshot.

a. Use the `btrfs sub list` command to list the subvolumes and snapshots.

- Output is sample only. The `ID` and `gen` numbers may be different.

```
# btrfs sub list /btrfs
ID 258 gen 10 top level 5 path SV1
ID 259 gen 10 top level 5 path SV1-snap
```

- In this example, the ID of the `SV1` subvolume is `258`.
- The ID of the `SV1-snap` subvolume (snapshot) is `259`.
- The IDs of your subvolumes might be different. Substitute your IDs as necessary in the following steps.

b. Use the `btrfs sub get-default` command to get the default subvolume ID of the file system.

```
# btrfs sub get-default /btrfs
ID 5 (FS_TREE)
```

- Note that the ID of `5` indicates that the `root` subvolume is the default.

c. Use the `btrfs sub set-default` command to set the subvolume ID to `258` (ID of the `SV1` subvolume).

```
# btrfs sub set-default 258 /btrfs
```

d. Use the `btrfs sub get-default` command to get the default subvolume ID of the file system.

```
# btrfs sub get-default /btrfs
ID 258 gen 12 top level 5 path SV1
```

- Note that the default subvolume ID is `258` for the `SV1` subvolume.

e. Use the `umount` command to unmount the file system mounted on `/btrfs`.

```
# umount /btrfs
```

f. Use the `mount` command to mount the file system.

```
# mount -a
```

g. Use the `ls -lR` command to display the contents of `/btrfs`.

```
# ls -lR /btrfs
/btrfs/:
-rwxr-xr-x ... vmlinuz-3.10...
-rwxr-xr-x ... vmlinuz-3.8.13...
```

- Note that the contents of the `SV1` subvolume are now mounted on `/btrfs`.

h. Use the `btrfs sub set-default` command to set the subvolume ID to `259` (ID of the `SV1-snap` snapshot).

```
# btrfs sub set-default 259 /btrfs
```

i. Use the `btrfs sub get-default` command to get the default subvolume ID of the file system.

```
# btrfs sub get-default /btrfs
ID 259 gen 10 top level 5 path SV1-snap
```

- Note that the default subvolume ID is `259` for the `SV1-snap` subvolume.

j. Use the `umount` command to unmount the file system mounted on `/btrfs`.

```
# umount /btrfs
```

k. Use the `mount` command to mount the file system.

```
# mount -a
```

l. Use the `ls -lR` command to display the contents of `/btrfs`.

```
# ls -lR /btrfs
/btrfs/:
-rwxr-xr-x ... vmlinuz-3.10...
```

- Note that the contents of the `SV1-snap` snapshot are now mounted on `/btrfs`.

4. Mount the `root` subvolume.

a. Use the `btrfs sub set-default` command to set the subvolume ID to `5` (ID of the root subvolume).

```
# btrfs sub set-default 5 /btrfs
```

b. Use the `btrfs sub get-default` command to get the default subvolume ID of the file system.

```
# btrfs sub get-default /btrfs
ID 5 (FS_TREE)
```

- Note that the default subvolume ID is `5` for the root subvolume.

c. Use the `umount` command to unmount the file system mounted on `/btrfs`.

```
# umount /btrfs
```

d. Use the `mount` command to mount the file system.

```
# mount -a
```

e. Use the `ls -lR` command to display the contents of `/btrfs`.

```
# ls -lR /btrfs
/btrfs/:
drwxr-xr-x ... SV1
drwxr-xr-x ... SV1-snap

/btrfs/SV1:
-rwxr-xr-x ... vmlinuz-3.10...
-rwxr-xr-x ... vmlinuz-3.8.13...

/btrfs/SV1-snap:
-rwxr-xr-x ... vmlinuz-3.10...
```

- Note that the root subvolume is now mounted on `/btrfs`.
- You could also mount the root file system as follows, but this does not change the default subvolume ID (do not perform this command, this is information only):
  `# mount -o subvolid=5 /dev/xvdb /btrfs`
- ID of `0` can also be used to mount the root subvolume.

5. Delete the snapshot.

a. Use the `btrfs sub delete` command to delete the snapshot.

```
# btrfs sub delete /btrfs/SV1-snap
Delete subvolume '/btrfs/SV1-snap'
```

b. Use the `btrfs sub list` command to list the subvolumes and snapshots.

- Rerun this command if an error is encountered.

```
# btrfs sub list /btrfs
ID 258 gen 12 top level 5 path SV1
ID 259 gen 10 top level 0 path DELETED
```

- Note that the `SV1-snap` snapshot is DELETED.

c. Use the `ls -lR` command to display the contents of `/btrfs`.

```
# ls -lR /btrfs
/btrfs/:
drwxr-xr-x ... SV1

/btrfs/SV1:
-rwxr-xr-x ... vmlinuz-3.10...
-rwxr-xr-x ... vmlinuz-3.8.13...
```

- Note that the `SV1-snap` directory is removed.

6. Take a snapshot of a file.
   a. Run the `sync` command, and then use the `df -h` command to display disk space usage on the `/btrfs` file system.

```
# sync
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb        10G   20M   8.0G    1%  /btrfs
```

   b. Use the `cd` command to change to the `/btrfs/SV1` directory.

```
# cd /btrfs/SV1
```

   c. Use the `cp` command to copy the `vmlinuz-3.10*` file to `copy_of_10`.

```
# cp vmlinuz-3.10* copy_of_10
```

   d. Run the `sync` command, and then use the `df -h` command to display disk space usage on the `/btrfs` file system.

```
# sync
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb        10G   29M   8.0G    1%  /btrfs
```

   - Note that the "Used" amount increased from 20M to 29M.

   e. Use the `rm` command to remove the `copy_of_10` file.

```
# rm copy_of_10
rm: remove regular file 'copy_of_10'? y
```

   f. Run the `sync` command, and then use the `df -h` command to display disk space usage on the `/btrfs` file system.

```
# sync
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb        10G   20M   8.0G    1%  /btrfs
```

   - Note that the "Used" amount decreased from 29M to 20M.

   g. Use the `cp --reflink` command to copy the `vmlinuz-3.10*` file to `copy_of_10`.

```
# cp --reflink vmlinuz-3.10* copy_of_10
```

   h. Use the `ls -l` command to display the sizes of the original file and the copy.

```
# ls -l
-rwxr-xr-x ... 4902184  ...  copy_of_10
-rwxr-xr-x ... 4902184  ...  vmlinuz-3.10...
-rwxr-xr-x ... 4587024  ...  vmlinuz-3.8.13...
```

   - Note that the sizes of the `vmlinuz-3.10*` and `copy_of_10` files are the same.

   i. Use the `diff` command to compare the original file and the copy.

```
# diff vmlinuz-3.10* copy_of_10
```

   - No output indicates that there are no differences in the two files.

j. Run the `sync` command, and then use the `df –h` command to display disk space usage on the `/btrfs` file system.

```
# sync
# df –h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb        10G   20M   8.0G    1%  /btrfs
```

- Note that the “Used” amount remained at 20M even after making a copy.
- The data blocks are not duplicated when the `--reflink` option is used.
- This allows the copy to be almost instantaneous and also saves disk space.
- One restriction is that this operation works only within the boundaries of the same file system and within the same subvolume.

Practice 13-3: Recovering from Data Corruption

Overview

In this practice, you perform the following:

- Prepare the environment by re-creating a RAID-1 file system.
- Use the `btrfs-corrupt-block` utility to induce data corruption.
- Use the `btrfs scrub` command to recover from data corruption.

Assumptions

- You are the `root` user on the **dom0**.
- You are the `root` user on the **host03** VM.

Tasks

1. Prepare the environment.
   a. From **host03**, use the `cd` command to change to the `root` user home directory, and then use the `umount` command to unmount `/btrfs`.

```
# cd
# umount /btrfs
```

   b. Use the `mkfs.btrfs` command to make a Btrfs file system with two disks, `/dev/xvdb` and `/dev/xvdd`, and specify RAID-1 for data.

```
# mkfs.btrfs -f -L Btrfs -d raid1 /dev/xvd[bd]
...
```

   c. Use the `mount` command to mount the file system.

```
# mount -a
```

   d. Use the `cp` command to copy `/boot/vmlinuz-3.10*` to `/btrfs`.

```
# cp /boot/vmlinuz-3.10* /btrfs
```

   e. Run the `sync` command, and then use the `df -h` command to display space information for `/btrfs`.

```
# sync
# df -h /btrfs
Filesystem      Size  Used  Avail  Use%  Mounted on
/dev/xvdb        10G   11M    8.0G    1%  /btrfs
```

   f. Use the `btrfs filesystem show` command to view the structure of the file system.

```
# btrfs file show
Label: Btrfs  uuid: ...
        Total devices 2 FS bytes used 5.30MiB
        devid    1 size 5.00GiB used 2.03GiB path /dev/xvdb
        devid    2 size 5.00GiB used 2.01GiB path /dev/xvdd
Btrfs v3.12
```

2. Induce data corruption on the Btrfs file system.

Perform these steps from **dom0**.

a. From **dom0**, open a new terminal window and use the `su -` command to become the `root` user (password is `oracle`) on **dom0**.

```
[dom0]$ su -
Password: oracle
[dom0]#
```

b. As the `root` user on **dom0**, use the `cd` command to change to the `/OVS/seed_pool/btrfs` directory. Use the `ls` command to list the directory contents.

```
[dom0]# cd /OVS/seed_pool/btrfs
[dom0]# ls
btrfs-corrupt-block
```

c. From **dom0**, use the `sftp` command to connect to **host03** as `root`. Password is `oracle`.

```
# sftp host03
Connecting to host03...
root@host03's password: oracle
sftp>
```

d. From the `sftp>` prompt on **dom0**, use the `put` command to copy `btrfs*` to **host03**. Use the `quit` command to close the connection after copying the file.

```
sftp> put *
Uploading btrfs-corrupt-block to /root/btrfs-corrupt-block
btrfs-corrupt-block ...
sftp> quit
```

Perform remaining steps from **host03**.

e. From **host03**, use the `cd` command to change to the `/btrfs` directory, and then run the `filefrag` command to report on file fragmentation on the `vmlinuz-3.10*` file.

```
# cd /btrfs
# filefrag -v vmlinuz-3.10*
Filesystem type is 9123683e
File size of vmlinuz-3.10... is 4902184 (1197 blocks of 4096
bytes)
 ext: logical_offset:  physical_offset: length: expected: flags:
   0:    0..    1196: 269440..  270636:   1197:          eof
vmlinuz-3.10...: 1 extent found
```

- In this example, the file is on physical block `269440` and has a 4K (`4096`) block size.

f. Use a calculator to multiply the physical block by the block size.

1) Multiply `4096 * 269440 = 1103626240`.
2) In this example, `1103626240` is the amount that you want to corrupt.

g. Use the `cd` command to change to your home directory, and then use the `umount` command to unmount `/btrfs`.

```
# cd
# umount /btrfs
```

h. Use the `btrfs-corrupt-block` tool to corrupt the first copy of that block.

- This is a very dangerous tool, and is not part of the `btrfs-progs` RPM.
- Do not use this tool outside the parameters of this lesson.
- By default, this tool corrupts all the copies of the block.
- You want to corrupt only the first copy of the block.

```
# ./btrfs-corrupt-block -c 1 -l 1103626240 /dev/xvdb
mirror 1 logical 1103626240 physical 1083703296 device /dev/xvdd
corrupting 1103626240 copy 1
mirror 2 logical 1103626240 physical 1103626240 device /dev/xvdb
```

3. Mount the file system and scrub.

a. Use the `mount` command to mount the file system.

```
# mount -a
```

b. Use the `btrfs scrub` command to start a file system scrub.

```
# btrfs scrub start /btrfs
scrub started on /btrfs/, fsid ... (pid=...)
```

c. Use the `btrfs scrub` command to get the status of the file system scrub.

```
# btrfs scrub status /btrfs
Scrub status for ...
        scrub started at ... and finished after 0 seconds
        total bytes scrubbed: 10.60MiB with 1 errors
        error details: csum=1
        corrected errors: 1, uncorrectable errors: 0,
        unverified errors: 0
```

- Note that one error was found, a checksum error, and one error was corrected.

d. Use the `dmesg` command to determine what happened.

```
# dmesg
...
[...] btrfs: checksum error at logical 1103626240 on dev
/dev/xvdd, sector 2116608, root 5, inode 257, offset 0, length
4096, links 1 (path: vmlinuz-3.10.0-123.el7.x86_64)
[...] btrfs: bdev /dev/xvdd errs: wr 0, rd 0, flush 0, corrupt
1, gen 0
[...] btrfs: fixed up error at logical 1103626240 on dev
/dev/xvdd
```

- Note that the error induced by the `btrfs-corrupt-block` command at logical `1103626240` was corrected by the scrub.

e. Use the `btrfs scrub` command to start a file system scrub.

```
# btrfs scrub start /btrfs
scrub started on /btrfs/, fsid ... (pid=...)
```

f. Use the `btrfs scrub` command to get the status of the file system scrub.

```
# btrfs scrub status /btrfs
Scrub status for ...
        scrub started at ... and finished after 0 seconds
        total bytes scrubbed: 10.60MiB with 0 errors
```

- Note that 0 errors were found.

4. Prepare for future practices.

a. Use the `umount` command to unmount `/btrfs`.

```
# umount /btrfs
```

b. Use the `vi` editor to remove the following entry in `/etc/fstab`:

```
# vi /etc/fstab
...
/dev/xvdb  /btrfs  btrfs  defaults  0  0    (delete this entry)
```